A DEEP LEARNING APPROACH TO PROTON BACKGROUND REJECTION FOR POSITRON ANALYSIS WITH THE AMS ELECTROMAGNETIC CALORIMETER

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

RAHEEM KARIM HASHMANI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

JANUARY 2023

Approval of the thesis:

A DEEP LEARNING APPROACH TO PROTON BACKGROUND REJECTION FOR POSITRON ANALYSIS WITH THE AMS ELECTROMAGNETIC CALORIMETER

submitted by **RAHEEM KARIM HASHMANI** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** ________________

Prof. Dr. Mehmet Halit S. Oğuztüzün
Head of Department, **Computer Engineering** ________________

Assist. Prof. Dr. Emre Akbaş
Supervisor, **Computer Engineering, METU** ________________

Prof. Dr. Melahat Bilge Demirköz
Co-supervisor, **Physics, METU** ________________

Examining Committee Members:

Assoc. Prof. Dr. Sinan Kalkan
Computer Engineering, METU ________________

Assist. Prof. Dr. Emre Akbaş
Computer Engineering, METU ________________

Prof. Dr. Melahat Bilge Demirköz
Physics, METU ________________

Prof. Dr. Muammer Altan Çakır
Physics Engineering, İstanbul Teknik Üniversitesi ________________

Assoc. Prof. Dr. Baosong Shan
Institute of Artificial Intelligence, Beihang University ________________

Date: 26.01.2023

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: RAHEEM KARIM HASHMANI

Signature :

ABSTRACT

A DEEP LEARNING APPROACH TO PROTON BACKGROUND REJECTION FOR POSITRON ANALYSIS WITH THE AMS ELECTROMAGNETIC CALORIMETER

HASHMANI, RAHEEM KARIM
M.S., Department of Computer Engineering
Supervisor: Assist. Prof. Dr. Emre Akbaş
Co-Supervisor: Prof. Dr. Melahat Bilge Demirköz

January 2023, 116 pages

The Alpha Magnetic Spectrometer (AMS) is a high-precision particle detector onboard the International Space Station containing six different subdetectors. The Transition Radiation Detector and Electromagnetic Calorimeter (ECAL) are used to separate electrons/positrons from the abundant cosmic-ray proton background.

The positron flux measured in space by AMS falls with a power law which unexpectedly softens above 25 GeV and then hardens above 280 GeV. Several theoretical models try to explain these phenomena, and a purer measurement of positrons at higher energies is needed to help test them. The current methods to reject the proton background at high energies involve extrapolating shower features from the ECAL to use as inputs for boosted decision tree and likelihood classifiers.

We present a new approach for particle identification with the AMS ECAL using deep learning (DL). By taking the energy deposition within all the ECAL cells as an input and treating them as pixels in an image-like format, we train an MLP, a CNN, and multiple ResNets and Convolutional vision Transformers (CvTs) as shower classifiers.

Proton rejection performance is evaluated using Monte Carlo (MC) events and AMS data separately. For MC, using events with a reconstructed energy between 0.2 – 2 TeV, at 90% electron accuracy, the proton rejection power of our CvT model is more than 5 times that of both the other DL models and the AMS models. Similarly, for AMS data with a reconstructed energy between 50 – 70 GeV, the proton rejection power of our CvT model is more than 2.5 times that of the AMS models.

Keywords: Particle Classification, Cosmic Rays, Electromagnetic Calorimeter, Deep Learning, Vision Transformers

ÖZ

AMS ELEKTROMANYETİK KALORİMETRESİ İLE POZİTRON ANALİZİ İÇİN PROTON ARKAPLAN REDDİNE BİR DERİN ÖĞRENME YAKLAŞIMI

HASHMANI, RAHEEM KARIM
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Dr. Öğr. Üyesi Emre Akbaş
Ortak Tez Yöneticisi: Prof. Dr. Melahat Bilge Demirköz

Ocak 2023, 116 sayfa

Alfa Manyetik Spektrometresi (AMS), Uluslararası Uzay İstasyonunda altı farklı alt dedektör içeren yüksek hassasiyetli bir parçacık detektörüdür. Geçiş Radyasyonu Detektörü ve Elektromanyetik Kalorimetresi (ECAL), kozmik ışınlarda bol olan proton arka planından elektronları/pozitronları ayırmak için kullanılır.

AMS tarafından uzayda ölçülen pozitron akısı, beklenmedik bir şekilde 25 GeV'nin üzerinde yumuşayan ve ardından 280 GeV'nin üzerinde sertleşen bir güç yasasıyla düşer. Birkaç teorik model bu fenomeni açıklamaya çalışır ve onları test etmeye yardımcı olmak için daha yüksek enerjilerde pozitronların daha saf bir ölçümüne ihtiyaç bulunmaktadır. Yüksek enerjilerde proton arka planını reddetmek için şu anda kullanılan yöntemler, güçlendirilmiş karar ağacı ve olasılık sınıflandırıcılarına girdi olarak kullanmak üzere ECAL'den parçacık yağmurunun özelliklerinin tahmin edilmesini içerir.

Derin öğrenmeyi (DÖ) kullanarak AMS ECAL ile parçacık tanımlama için yeni bir yaklaşım sunuyoruz. Tüm ECAL hücrelerindeki enerji birikimini bir girdi olarak alarak ve bunlara görüntü benzeri bir formatta pikseller olarak davranarak, parçacık yağmur sınıflandırıcıları olarak bir MLP, bir CNN ve birden çok ResNet'ler ve Evrişimsel Görüntü Dönüştürücü'ler (CvT) eğitiyoruz. Proton reddetme performansı, Monte Carlo (MC) olayları ve AMS verileri kullanılarak ayrı ayrı değerlendirilir. MC için, %90 elektron doğruluğunda 0,2 – 2 TeV arasında yeniden yapılandırılmış enerjiye sahip olaylar kullanıldığında, CvT modelimizin proton reddetme gücü, hem diğer DÖ modellerinden hem de AMS modellerinden 5 kat daha fazladır. Benzer şekilde, yeniden yapılandırılmış enerjisi 50 – 70 GeV arasında olan AMS verileri için, CvT modelimizin proton reddetme gücü, AMS modellerinden 2,5 kat daha fazladır.

Anahtar Kelimeler: Parçacık Sınıflandırması, Kozmik Işınlar, Elektromanyetik Kalorimetre, Derin Öğrenme, Görüş Dönüştürücüler

Patri carissimo. Requiescat in pace.

ACKNOWLEDGMENTS

I would first and foremost like to thank my advisors. When it comes to Prof. M. Bilge Demirköz, it is a challenge itself to fit more than 5 years' worth of acknowledgement in a few sentences. From taking me on as a novice when I joined her group in 2017, to preparing me for the life of a scientist, she has been my strongest advocator.

Since focusing on deep learning, Prof. Emre Akbaş has been a fountain of knowledge who helped bring me up to speed on a variety of machine learning topics. Without his warm personality and guidance, attaining this degree would be almost impossible.

I would like to thank Prof. Samuel C. C. Ting and the AMS Collaboration for their constant support of my endeavors at AMS. In particular, I would like to thank Dr. Zhili Weng, who acted like a third advisor, and Prof. Shan, Dr. Capell, Dr. Cai, and Dr. Konyushikhin, for their invaluable advice.

There are many who impacted my life and research at METU and IVMER. Of particular note, I would like to thank Ugur Kılıç, Erinç Ünsür, and Cenk Tüysüz for their support, Gülce Karagöz, Egecan Karadöller, and Janset Turan for their discussions on AMS, and Berk Türk for his contributions. I would also like to thank Prof. Altuğ Özpineci, who has been a strong supporter since my freshman year.

Making it this far would be difficult without the support of my friends Brunilda Muçogllava and Gubad Islamov. Around the globe, I had the virtual companionship of Saad, Anum, Murtaza, Hamza, Kohmeer, Daniyal, Hassan, Salman, and Farzan.

Last but not least, I owe everything to my family. Especially Karim and Rozmeen Hashmani, the best parents anyone could ask for, Assadali and Anahita, my cherished siblings, and Habib, Shireen, and Shahsultan, who are constantly supporting me.

I would like to acknowledge the financial support of the Scientific and Technological Research Council of Turkey (TÜBİTAK) and the Turkish Energy, Nuclear and Mineral Research Agency (TENMAK) under Grant No. 2020TAEK(CERN)A5.H1.F5-26.

TABLE OF CONTENTS

CHAPTERS

APPENDICES

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

ACC	Anti-Coincidence Counters
AMS	Alpha Magnetic Spectrometer
BDT	boosted decision tree
CERN	European Organization for Nuclear Research
CNN	convolutional neural network
CvT	Convolutional vision Transformer
DL	deep learning
ECAL	Electromagnetic Calorimeter
GeV	giga-electronvolts
HistBDT	histogram-based gradient boosting decision tree
ISS	International Space Station
IVMER	The Research and Application Center for Space and Accelerator Technologies
LHD	likelihood function
LogReg	logistic regression
MC	Monte Carlo
ML	machine learning
MLP	multilayer perceptron
PMT	photomultiplier tube
POCC	Payload Operations Control Center
ReLU	Rectified Linear Unit
ResNet	residual neural network
RICH	Ring Imaging Cherenkov
ROC	receiver operating characteristic
SVM	support vector machine
TeV	tera-electronvolts
ToF	Time of Flight

TR	transition radiation
Tracker	Silicon Tracker
TRD	Transition Radiation Detector
val	validation

CHAPTER 1

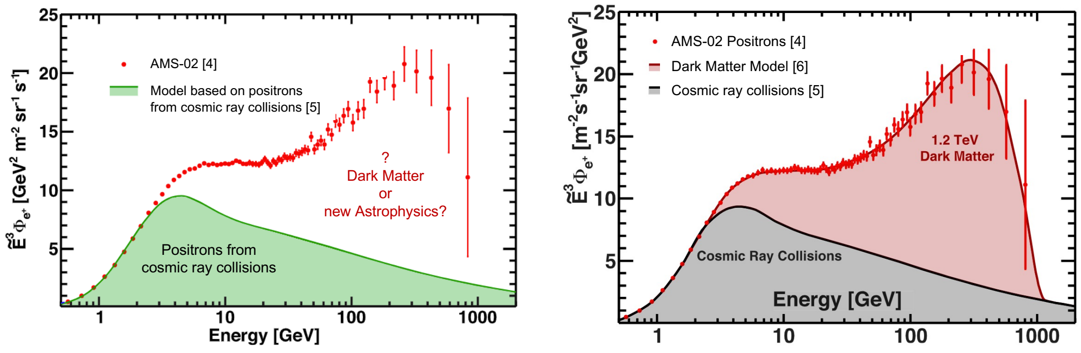
INTRODUCTION

There are many aspects of the Universe that are yet to be fully understood. Why is there more matter than antimatter when they were both created equally during the Big Bang? What really is dark matter, the undetectable substance responsible for the motion of the galaxies at large? Where do cosmic rays come from? This thesis is a modest attempt at helping to understand the Universe at large.

1.1 Motivation

The Alpha Magnetic Spectrometer (AMS) is a general purpose, high-precision particle detector onboard the International Space Station (ISS). It was installed on 19 May 2011 to conduct a long-duration mission of fundamental physics research in space. It has three primary scientific objectives: searching for antimatter, investigating dark matter and analyzing cosmic rays. To aid in its objectives, AMS is outfitted with 6 subdetectors: the gaseous Xe/CO_2 Transition Radiation Detector (TRD), the nine layer Silicon Tracker (Tracker), two Time of Flight (ToF) Counters, the aerogel/NaF Ring Imaging Cherenkov (RICH), and the lead/scintillator fiber Electromagnetic Calorimeter (ECAL). At the center of AMS is a permanent Nd-Fe-B magnet, which generates a magnetic field of 0.15 T [1]. AMS has been continuously collecting cosmic ray particles since its installation onboard the ISS and has collected over 215 billion cosmic ray events [2].

One important goal of AMS is to accurately measure the flux of cosmic positrons in space [3]. Figure 1.1 shows the measurement of the cosmic positron flux detected by AMS [4] and how it defers from the theoretical calculations of the flux from cosmic ray collisions [5], which is the main source of cosmic positrons. At high energies the results agree more closely with other theoretical models, for example positrons being produced from dark matter interaction [6]. As such, these results provide significant evidence for the existence of dark matter [4].



(a)

(b)

Figure 1.1: The cosmic positron flux measured by AMS [4] compared to (a) the theoretical results of the cosmic positron source being cosmic ray collisions [5] and (b) a dark matter model [6].

The currently used analysis methods to separate cosmic positrons from the abundant proton background have a large uncertainty at energies close to 1 tera-electronvolts (TeV) and above. Of the two subdetectors capable of differentiating between positrons and protons, the TRD's proton rejection falls rapidly after 100 giga-electronvolts (GeV) [7] per charge of the particle (1 in the case of protons and positrons) and the ECAL can only partially contain electromagnetic showers at TeV energies [8].

This thesis proposes a deep learning (DL) approach to help improve positron classification for the ECAL. Using four architectures, we investigate the ability of deep learning to extract more information out of electromagnetic showers, and the classification efficiency compared to the currently used boosted decision tree (BDT) and likelihood function (LHD) methods. To do so, we propose the treatment of the ECAL cells as pixels in a 3D array, in order to utilize the entire raw data output instead of handcrafted features, therefore treating cosmic ray shower as images. This allows the use of state-of-the-art models to classify positrons against protons.

1.2 Proposed Methodology

The ECAL output consists of energy depositions in 1296 cells organized into 18 layers with 72 cells in each layer. Every 2 layers, denoted as a superlayer, is dedicated to either the X or the Y direction. Four superlayers are dedicated to the X direction while five are dedicated to the Y direction [8]. This allows for 3D imaging of the ECAL shower and has a data format that resembles images.

We mainly use Monte Carlo (MC) simulations of cosmic ray protons and electrons/-positrons to build datasets consisting of showers whose ECAL reconstructed energy ranges from 200 GeV to 2 TeV. As a final experiment, we train and test on actual data collected by AMS, denoted as ISS data, that have a reconstructed energy between 50 and 70 GeV.

The main deep learning architectures trained and tested are the multilayer perceptron (MLP), convolutional neural network (CNN), residual neural network (ResNet) [9], and Convolutional vision Transformer (CvT) [10]. For most of the experiments, we first divide our datasets by separating the particles with a reconstructed energy below 1 TeV from the ones above 1 TeV. Then, we further divide each into train and test sets for below 1 TeV particles and validation (val) and test sets for above 1 TeV particles. We train the models on the train sets and validate after each epoch. This continues until 20 training epochs with no improvement of the model performance on the validation sets. If so, early stopping is employed and the trained model from the epoch with the best validation performance is saved. Following this, the models are evaluated on the test set using three performance metrics, the most important of which is the proton rejection versus electron efficiency.

By training on below 1 TeV particles and validating on above 1 TeV particles, our goal was to develop a model with a reduced dependence on the energy of the incoming particle itself. This would allow for a higher confidence when the model classifies real positrons at high energies (TeV range) even without a large sample of labeled real data to test on.

Additionally, to better understand what the model focuses on, we built a custom class activation heatmap generator to see which parts of the shower contribute the most to a model's classification score. Finally, we developed a physics-based feature engineering preprocessing method to improve model learning efficiency, trained on ISS data with reconstructed energies between 50 GeV and 70 GeV, and tested both our MC-trained and ISS-trained models on ISS data.

1.3 Contributions

Our contributions are as follows:

- We explore the potential of deep learning to classify cosmic positrons against the large proton background for the AMS ECAL.
- We build an AMS ROOT-to-ML pipeline that extracts the needed raw data and associated variables from AMS files using the AMS ROOT framework and prepares them for use with Python ML frameworks.
- We compare the performance of our deep learning models against the currently used AMS models on MC and show that at 90% electron accuracy, the proton rejection power of the CvT outperforms the best AMS model by factors of 153 and 7 for below 1 TeV and above 1 TeV particles, respectively.
- Comparing the performance of our deep learning models against themselves on MC, we show that at 90% electron accuracy, the proton rejection power of the CvT outperforms the ResNet18 model by factors of 5 and 14 for below 1 TeV and above 1 TeV particles, respectively.
- We show a consistent decrease in performance when training the CvT on smaller and smaller amounts of training data, prompting the need for an increased learning efficiency.
- We develop a physics-based, feature engineering preprocessing method that uses the cosmic ray's incident position, angles of incidence, and depth-wise energy deposition to extract the necessary, relevant pixels from an ECAL shower image to improve the efficiency of learning. Using this, the Phys+CvT model is able to perform equally on the above 1 TeV MC test set after having trained on only 5% of the training data.

- At 90% electron efficiency on 50-70 GeV ISS data, we show the Phys+CvT model outperforms the CvT model and best performing AMS model by a factor of 1.8 and 2.5, respectively.

1.4 The Outline of the Thesis

The remaining thesis is organized as follows:

- Chapter 2 provides a review on the AMS experiment in space, its physics goals, and its various subdetectors, with a particular focus on the ECAL.
- Chapter 3 describes the ML models we use, provides a literature review of ML and their use in physics, and gives details on our particular implementation, hyperparameters, and hardware.
- Chapter 4 outlines our experimental setups and details our implementation with AMS ECAL data.
- Chapter 5 talks about our experiments and showcases the results.
- Chapter 6 presents our conclusion and proposes future directions.

CHAPTER 2

THE AMS EXPERIMENT

2.1 Introduction and Goals

Cosmic rays provide an excellent insight into the Universe, its composition, and other high energy astronomical phenomena taking place in space. First discovered in 1917 [11], they have since helped in understanding processes taking place in the interstellar medium [12], learning about properties of astronomical objects such as blazars [13], and discovering a myriad of particles including the positron [14].

The Alpha Magnetic Spectrometer (AMS), officially denoted as the AMS-02, is a large acceptance magnetic spectrometer experiment onboard the International Space Station (ISS) designed to collect and analyze cosmic rays at an unprecedented precision. Installed on 19 May, 2011 [15], it was designed with three main science objectives [1]: searching for antimatter [16], finding evidence of dark matter [17], and studying the amount and origin of cosmic rays [18]. The AMS collaboration consists of 44 institutes spanning America, Asia, and Europe, including Turkey, and has the Payload Operations Control Center (POCC) located at CERN, Switzerland. The METU-AMS team has a long history of contributing to both the analysis side [19–22] and the operations side by taking shifts to monitor the detector's health and building the currently used monitoring interface and system [23].

Prior to the AMS-02, the AMS-01 was a precursor detector that flew on board the Space Shuttle Discovery in June 1998 for a total of 10 days [24]. It did not contain an ECAL or a TRD, and measured cosmic rays between the energy interval of 0.1 GeV to 200 GeV. It successfully showcased the potential of having a high-precision, multipurpose, magnetic spectrometer in space and helped in understanding cosmic rays even within this short interval of time [25, 26].

The AMS-02 contained an ECAL and a TRD, raised the accepted energy interval from 200 GeV to TeV energies and has been influential in our understanding of cosmic rays, dark matter, and antimatter [4].

2.2 Subdetectors

The AMS has 5 subdetectors and a Nd-Fe-B magnet shown in figure 2.1. It uses a Cartesian coordinate system with the center of the AMS being the origin. The z-axis is along the vertical length of the AMS, the x-axis is parallel to the main component of the magnetic field, and the y-axis is the bending direction of the magnet.

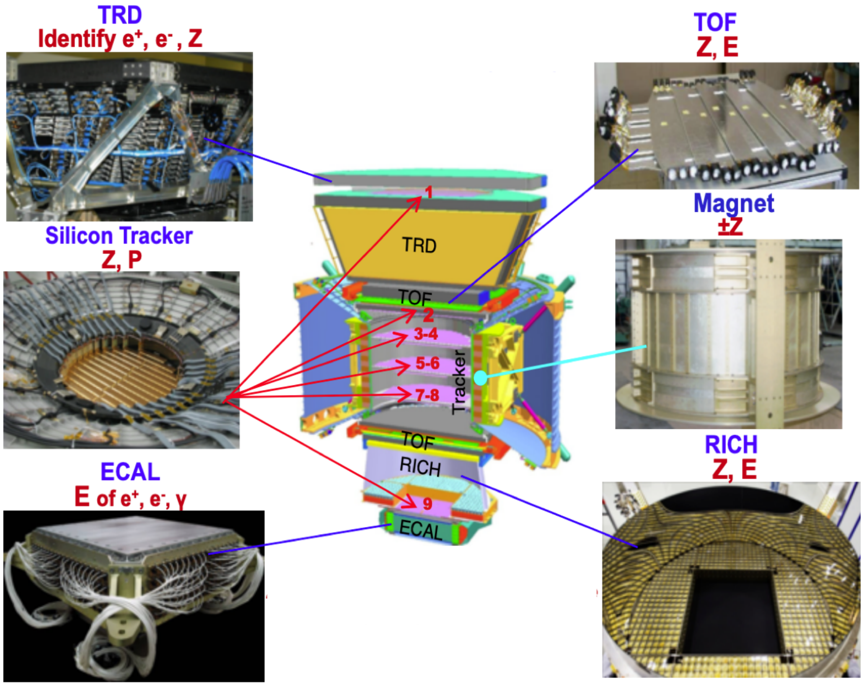


Figure 2.1: The AMS subdetectors and magnet [27]. As a cosmic ray passes through, it leaves energy depositions in each of the subdetectors. Each subdetector is then able to reconstruct a property of the cosmic ray, allowing us to identify the type and energy of a particle.

2.2.1 Transition Radiation Detector

The Transition Radiation Detector (TRD) contains 20 layers with each layer consisting of 20 mm thick fleece and proportional straw tubes filled with a Xe-CO_2 gaseous mixture, totalling 5248 tubes in total [28]. When cosmic rays pass through the fleece, they emit transition radiation (TR) in proportion to the Lorentz factor,

$$\gamma = \frac{E}{m_0 c^2} \tag{2.1}$$

where E is the energy of the incident particle, m_0 is the rest mass, and c is the speed of light. Since protons have a substantially larger mass than electrons and positrons, the latter have more intense TR at low energies. This TR, which is in the x-ray regime, results in extra ionization of the gaseous mixture on top of the energy deposition from the passage of the primary particle. Figure 2.3 showcases this schematically. The TRD thus provides an independent tracking ability and measurement of charge [29], as well as a separation of positron/electrons and protons.

A log-likelihood method is used to analyze the readout signals and create a TRD Likelihood (Λ_{TRD}) to determine whether the signals from each layer belongs to a proton or a positron/electron [30]. Figure 2.4a shows the Λ_{TRD} values for positrons/electrons and protons between the reconstructed energy ranges of 10–100 GeV. Figure 2.4b shows the proton rejection for the same energy range, where we can see that if we moved the dotted line from Figure 2.4a to the right, we would get a larger electron efficiency, but more misclassified protons, resulting in a lower proton rejection.

At higher energies, the Lorentz factor for protons (Eq. 2.1) starts to increase and they begin to emit TR, making them difficult to distinguish from positrons/electrons. As such, the TRD is generally considered most reliable for particles below 100 GeV [7] and cannot accurately separate protons from positrons/electrons at energies close to the TeV range.

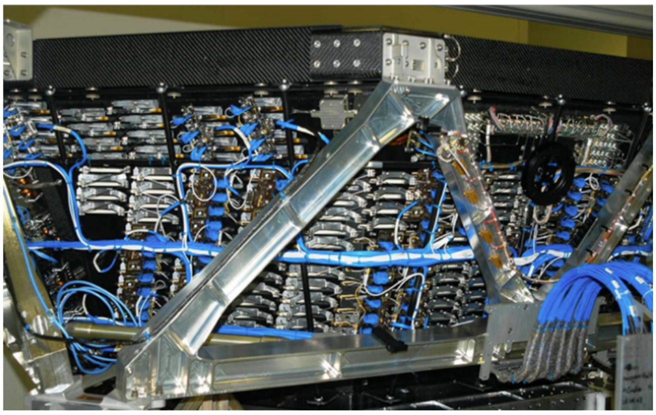


Figure 2.2: The TRD including the readout system and support structure [31].

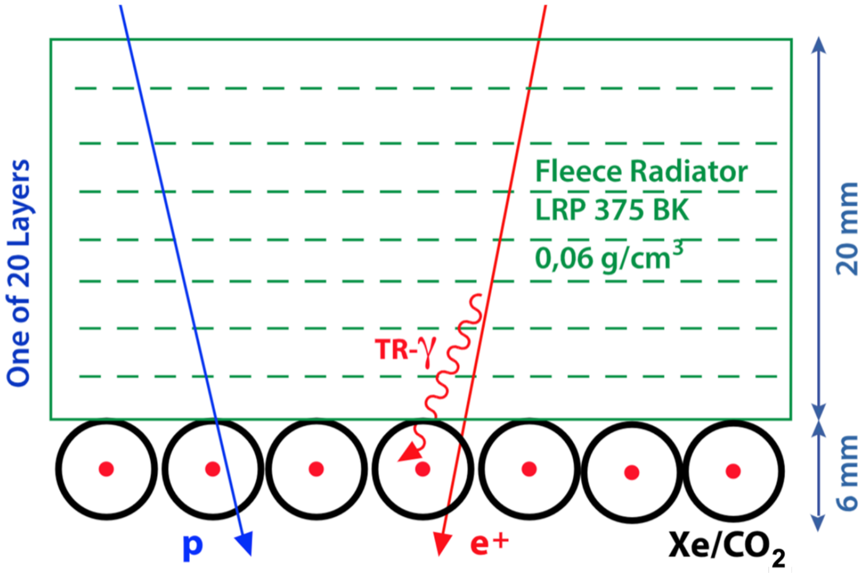
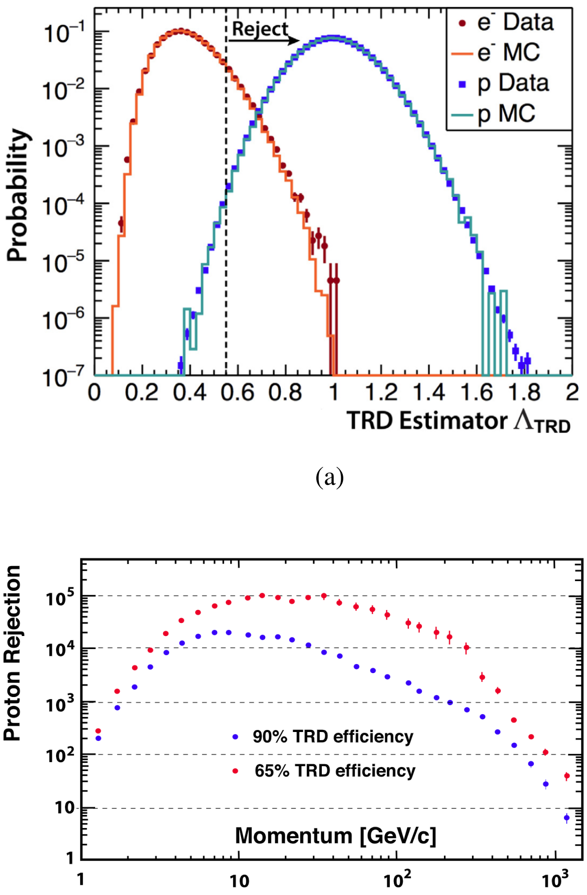


Figure 2.3: The working principle of the TRD for one of the TRD layers [4]. As a proton (P) passes through, only the proton ionizes the gas mixture as it generally does not emit transition radiation at low energies. As positrons (e^+) (and similarly electrons, e^-) pass through, they emit transition radiation (γ) and both the e^+ and γ ionize the gas mixture.



(a)

(b)

Figure 2.4: (a) The TRD Estimator (i.e. Likelihood) (Λ_{TRD}) probability distribution for Monte Carlo (MC) and ISS data for the reconstructed energy range of 10-100 GeV [4]. Keeping the value of Λ_{TRD} equal to the dotted vertical line would result in an electron efficiency of 90% (see (b)). If we moved the line to the right, we would get a larger electron efficiency, but more misclassified protons, resulting in a lower proton rejection. (b) Proton rejection versus momentum (which is equivalent to energy since $c = \pm 1$ for protons, positrons, and electrons) on ISS data for 90% and 65% TRD efficiency, which is equivalent to electron accuracy (see Section 4.4).

2.2.2 Silicon Tracker and Magnet

The Silicon Tracker (Tracker), depicted in Figure 2.5, makes up the core of the AMS and, along with the magnet, is a key component in differentiating between matter and antimatter.

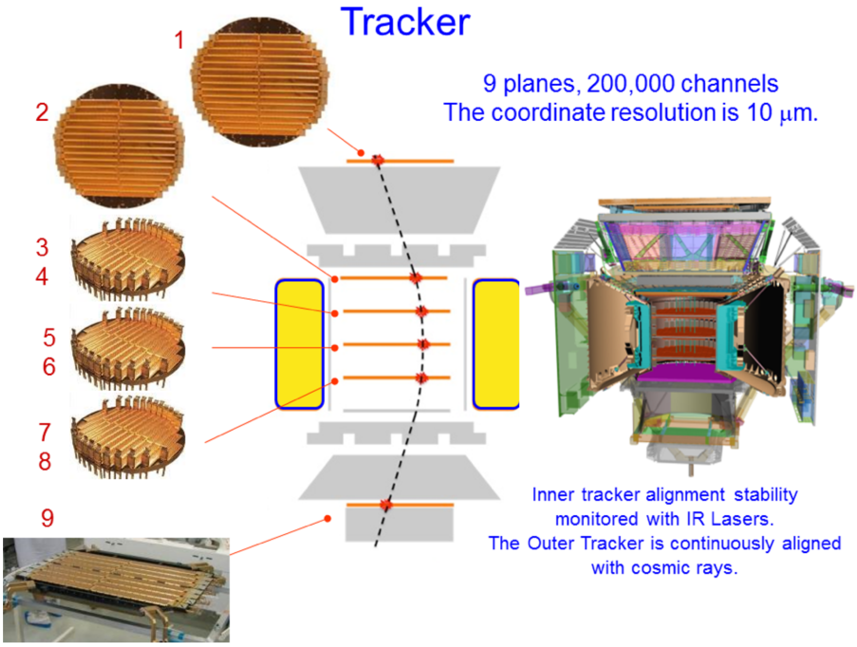

Tracker
1
2
3
4
5
6
7
8
9
9 planes, 200,000 channels
The coordinate resolution is 10 μm.
Inner tracker alignment stability
monitored with IR Lasers.
The Outer Tracker is continuously aligned
with cosmic rays.

Figure 2.5: A representation of the Silicon Tracker and its alignment system [2].

The Silicon Tracker consists of 9 layers, 7 of which are within the barrel of the magnet, and a layer on the top of both the TRD and ECAL. Each layer consists of a set of read-out units known as ladders, with a total of 192 ladders. Each ladder contains mechanic support, readout electronics, and silicon sensors [32]. As a particle passes through the layer material, it loses energy according to the Bethe-Bloch formula [33] where the deposited energy $\left\langle \frac{dE}{dx} \right\rangle$ is proportional to Z^2.

The permanent magnet at the center of the AMS [34] can be seen in Figure 2.6. Weighing 2.2 tons, it consists of 64 Neodymium-Iron-Boron sectors in the shape of a cylinder and produces a magnetic field of 1.4 kG which is well contained within the cylinder. The magnetic field runs along the y-axis of the AMS and has a maximum bending power of $BL^2 = 0.15$ Tm^2 [26]. Since matter and antimatter are identical in all properties except the sign of their charge, they will bend in opposite directions of the magnetic field, allowing them to be separable.

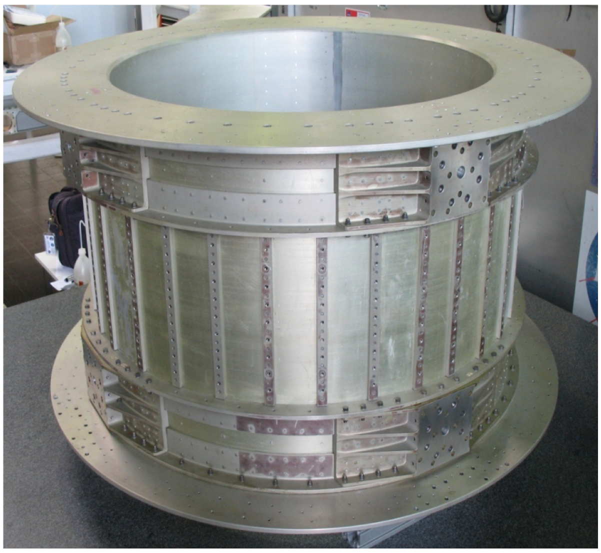


Figure 2.6: A photo of the AMS magnet [2].

Due to the magnetic field causing the cosmic ray to bend along the y direction, the particle's trajectory allows us to calculate the rigidity, R, defined as the ratio between a cosmic ray's energy and charge, using $R = \vec{B} \cdot \vec{r}$, where r is the radius of the curvature and B is the magnetic field.

Thus, the Tracker provides an independent measurement of the cosmic ray's charge, charge sign, rigidity, energy, and trajectory. In particular, we will later use the Tracker's trajectory coordinates to develop the physics-based feature engineering (see Section 4.3).

2.2.3 Time of Flight Counters and Anti-Coincidence Counters

The Time of Flight (ToF) counters, shown in Figure 2.7 consist of 4 layers of scintillation counters, 2 above and 2 below the magnet. When a cosmic ray passes through, it deposits energy in the counters which then produces photons that get absorbed by 2 or 3 photomultiplier tubes (PMTs) on each end of the counters [35]. This provides a high-efficiency trigger system that can measure the time between the particles entering and leaving the Tracker and allows for the ToF to independently measure the velocity and the energy of the particle. The triggering system is used to accept particles that come from the top of AMS while rejecting particles that come from below.

The Anti-Coincidence Counters (ACC) provide another triggering system. They consist of 16 scintillation panels surrounding the Tracker and reject particles coming or leaving the Tracker from any transverse direction [36].

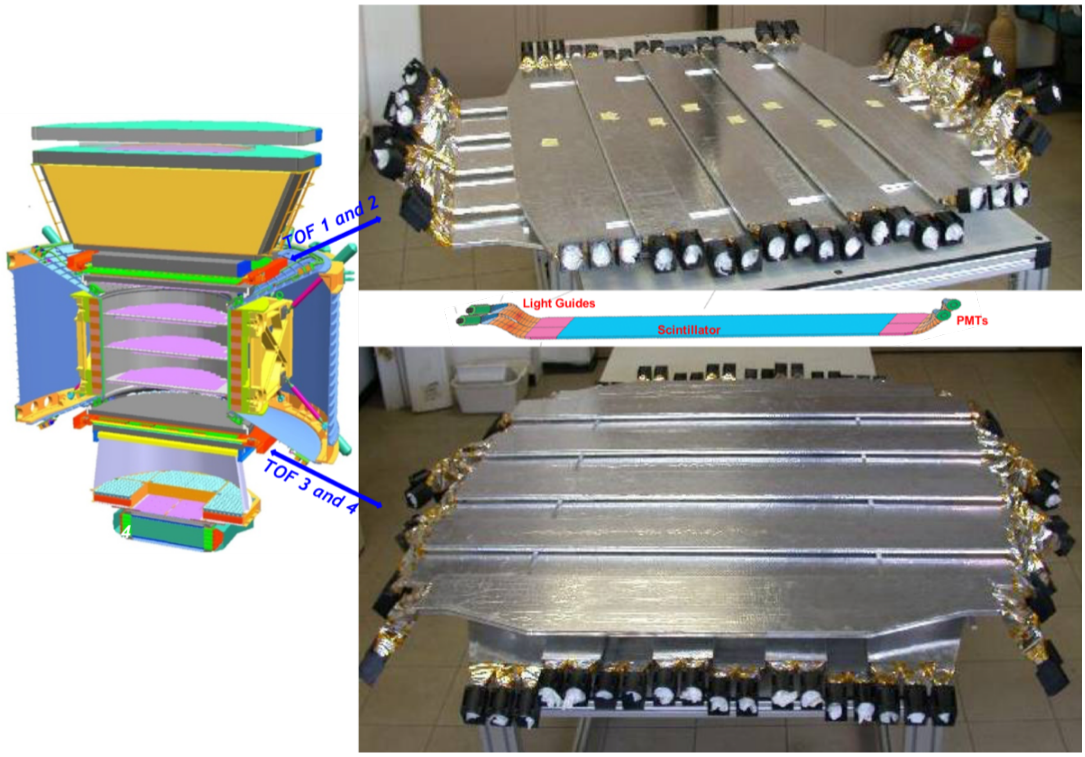


Figure 2.7: A photo of the ToF planes in an assembly test before they were integrated into AMS [2].

2.2.4 Rich Imaging Cherenkov Detector

The Ring Imaging Cherenkov (RICH) detector, shown in Figure 2.8, is placed above the ECAL and consists of a radiator plane, an expansion volume, and a photo-detection plane. The radiator plane is formed by 16 sodium fluoride tiles with a refractive index of $n = 1.33$ surrounded by 92 silica aerogel tiles with a refractive index of $n = 1.05$. As particles pass through these media, their speed is faster than the speed of light in these media, causing the release of Cherenkov radiation photons similar to the mechanism of a sonic boom. These photons comprise of a cone of light that expands in the expansion volume before being detected by the photo-detection volume which is composed of 680 PMTs [37]. The angular aperture of the cone is directly related to the velocity and charge of the particle, and thus the RICH detector provides an independent measurement of the energy and charge of a cosmic ray.

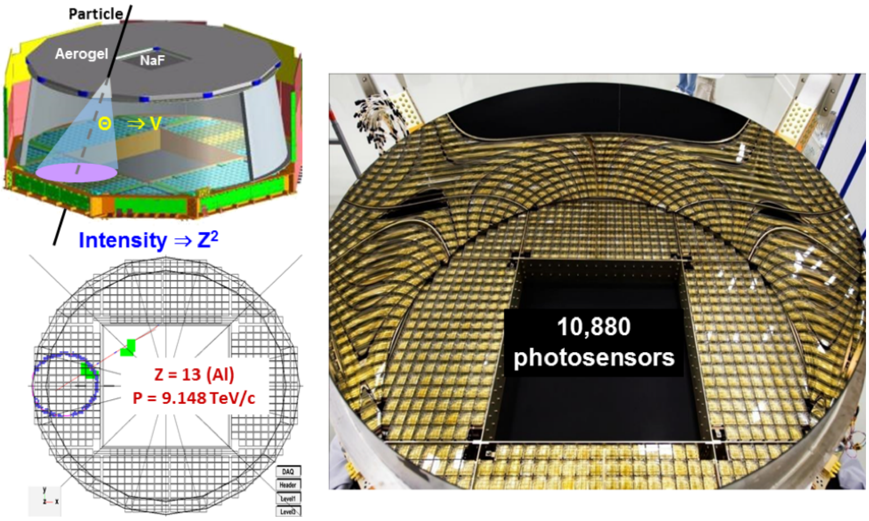


Figure 2.8: The RICH detector along with an event view showing a cone of Cherenkov radiation from a particle passing through the RICH [2].

2.2.5 Electromagnetic Calorimeter

The ECAL, depicted in Figure 2.9, is the main detector we work with. Located at the bottom of AMS, it is a sampling calorimeter consisting of lead-scintillating fibres arranged into an alternating direction every two layers. It measures $64.8 \times 64.8 \times 16.6$ cm^3 with each fibre having a diameter of 1 mm [8]. The depth corresponds to 17 radiation lengths, X_0, which is a unique characteristic of each detector material and is the distance a particle travels before it loses 63% of its energy [38].

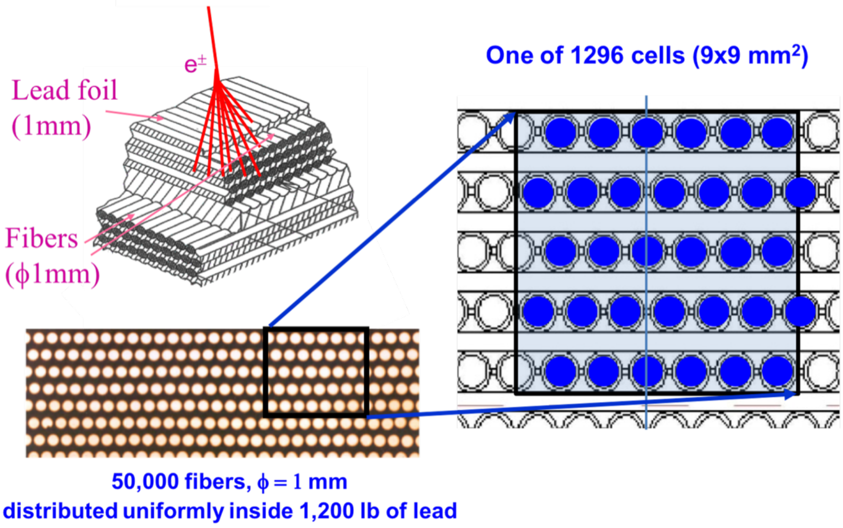

e±
Lead foil
(1mm)
Fibers
(ϕ1mm)
One of 1296 cells (9x9 mm²)
50,000 fibers, ϕ = 1 mm
distributed uniformly inside 1,200 lb of lead

Figure 2.9: Schematic of the ECAL, showing an electromagnetic shower passing through the ECAL fibres, as well as a view of the scintillating fibres themselves [2].

When a cosmic ray passes through the lead material, depending on the type and energy of the particle, it may produce an electromagnetic shower. At GeV energy ranges, this process is dominated by bremsstrahlung and pair production. Bremsstrahlung, depicted in Figure 2.10, occurs when a charged particle, such as an electron, is deflected by another charged particle, such as an atomic nucleus, and losses energy in the process. This energy is released in the form of a photon. Pair production occurs when a photon, in the presence of a nucleus, is converted into a matter-antimatter pair such as an electron and positron.

Most of the electromagnetic shower energy is absorbed by the lead while the scintillating fibres only sample the shower. When the ionizing shower particles interact with the scintillating material, it produces excited states in the atoms that subsequently decay back to their original states, releasing energy in the form of photons. The number of scintillation photons emitted is proportional to the amount of energy deposited by the shower in the scintillating fibres. These photons are then guided by Plexiglas light guides to the PMTs at the end of the fibres, which convert them into an electric signal proportional to the collected photons. These PMTs are organized together into cells, with each layer having 72 cells.

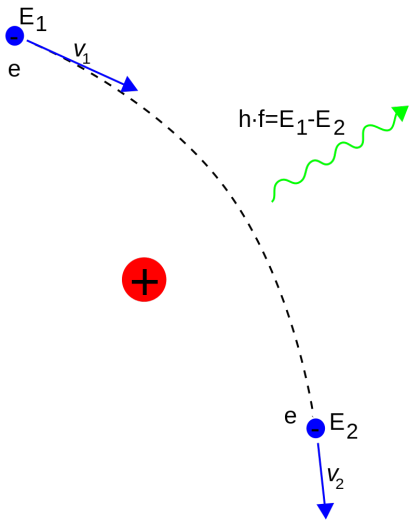


Figure 2.10: The working principle of Bremsstrahlung. At high enough energies, generally above 25 MeV, when a charged particle, such as an electron, is deflected by another charged particle, such as an atomic nucleus, the charged particle loses energy during the slowdown. This energy, known as braking radiation or bremsstrahlung, is released in the form of a photon.

There are 18 layers in total, with each layer having a thickness of approximately 9.2 mm. The 72 cells in each layer measure 9×9 mm^2, giving us a total of 18×72 layers. Every two layers, called a superlayer, has fibres running along either the x or the y direction, with the latter being parallel to the bending direction of the magnet. In total, there are 4 superlayers in the x-direction and 5 in the y-direction, which can be seen in Figure 2.11. This allows for a 3D imaging of the showers, as the depth provides the z direction and every superlayer provides information on either the x or y direction. As such, the ECAL provides an independent measurement of the energy and trajectory of a cosmic ray, as well as a separation between electrons/positrons and protons.

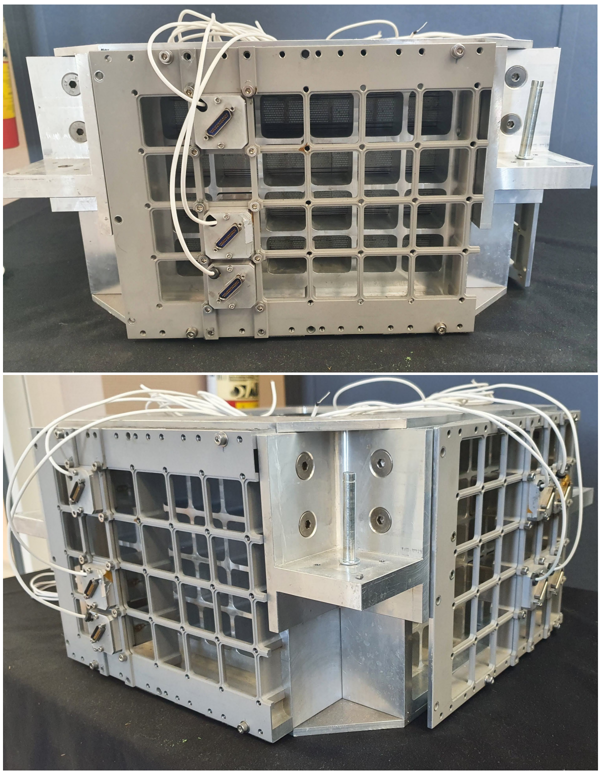


Figure 2.11: A replica of the ECAL body with the PMT housing. One side has 4 layers while the other has 5, showcasing the alternating direction of light collection.

2.2.5.1 Positron/Electron Shower Classification and Proton Rejection

The ECAL provides an efficient way to differentiate between positrons and protons. The electromagnetic shower occurring within the cells of the ECAL, making up a 3D view of the shower, depends on both the energy and type of cosmic ray. When an electron or positron with high enough energy (above 25 MeV) passes through the material, the effects of bremsstrahlung and pair production cause a cascade of particles that make up the electromagnetic shower. The physics of the electromagnetic shower is well understood [33]. For example, the distance at which the shower deposits the maximum amount of energy, known as shower maximum, t_{max}, is governed by,

$$t_{max} \approx \ln \frac{E}{E_c} \tag{2.2}$$

where E is the energy of the particle and E_c is a property of the material, the critical energy in lead as in the AMS ECAL. Similarly, 95% of the shower's width remains confined within two Moliére radii, R_M, which is defined as,

$$R_M = \frac{21MeV}{E_c} \times X_0 \tag{2.3}$$

Where, for the AMS ECAL, R_M is equivalent to 2 ECAL cells. Since the only differentiating feature between positrons and electrons is the sign of their charge, their showers look the same in the ECAL.

In contrast, when a proton passes through the ECAL, it behaves as a minimum ionizing particle for the most part, occasionally undergoing a hadronic shower for which the ECAL lacks the depth necessary to fully capture [38]. Thus, it deposits considerably less energy compared to an equally energetic positron/electron and has a different shower profile. Nonetheless, hadronic showers occasionally have an electromagnetic component [33] and given the large amount of cosmic protons, some of the hadronic showers within the ECAL might have properties of an electromagnetic shower [39].

The AMS currently uses two different models to differentiate between proton and positrons/electrons: the boosted decision tree (BDT) [39] and the likelihood function

(LHD) [40]. Both models were trained on test beam data and used datasets that are no longer available for our use. While their performance on MC electrons are similar to their performance on ISS data electrons, their performance on both MC and ISS protons differ by a small, but difficult to quantify amount (see Section 4.2 for an explanation for MC and ISS data discrepancies).

The BDT is a machine learning method that uses a regression based decision tree with the weights of misclassified events boosted [41]. It is widely used and was often considered the de facto method for particle classification [42] before the push for deep learning. AMS's BDT is trained on 19 correlated variables describing the ECAL shower such as shower maximum, width, energy in each layer and average location, and outputs a model score between -1 (protons) and +1 (positrons/electrons). Figure 2.12 shows that the BDT achieves a proton rejection power of 10^3 for particles with 1 TeV energy at 90% electron efficiency.

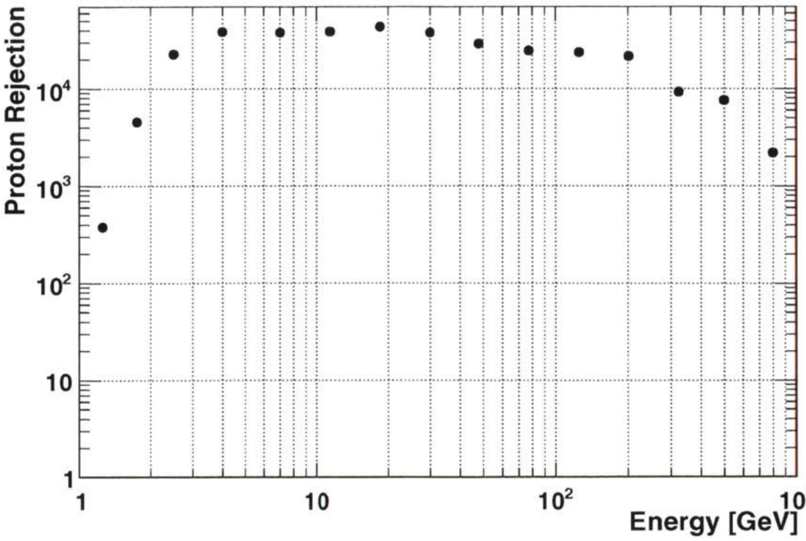


Figure 2.12: Proton rejection versus energy at 90% electron efficiency for the AMS BDT model. A rejection power of 10^3 is achieved around 1 TeV [39].

The AMS LHD is a relatively newer method developed to better identify positrons at TeV energies when compared to the BDT model. It uses 7 main variables extracted from the shower (shower energy, x-, y-, and z-coordinate of the shower maximum, the zenith and azimuth angles that the shower makes, and the distance along the shower axis from the start of the shower to the shower maximum) along with 9 other correlated variables (number of cells in the shower, energy deposition around the shower axis for the first two layers, the energy deposition around the shower axis for the 3rd layer, etc.) to create an ECAL Likelihood, Λ_{ECAL} that gives the log-likelihood of the shower being either a proton or a positron/electron. Figure 2.13 shows that the BDT achieves a proton rejection power of about 10^4 for particles with 1 TeV energy at 90% electron efficiency. A marked improvement over the AMS BDT.

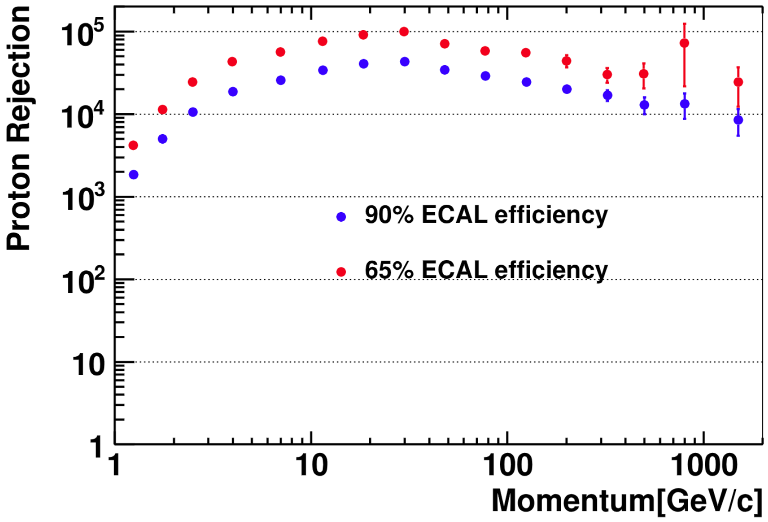


Figure 2.13: Proton rejection versus energy at 90% and 65% electron efficiency for the AMS LHD model [40]. A rejection power of about 10^4 is achieved around 1 TeV at both electron efficiencies.

2.2.5.2 AMS Subdetector Signatures

As a cosmic ray passes through each of the subdetectors, it leaves energy depositions that can be used to identify it. A representation of these energy deposition signatures can be seen in Figure 2.14. For protons and positrons, all of the subdetector signatures are similar except for the ones given by the TRD and ECAL. Additionally, proton and electron signatures look the same for both the TRD and ECAL.

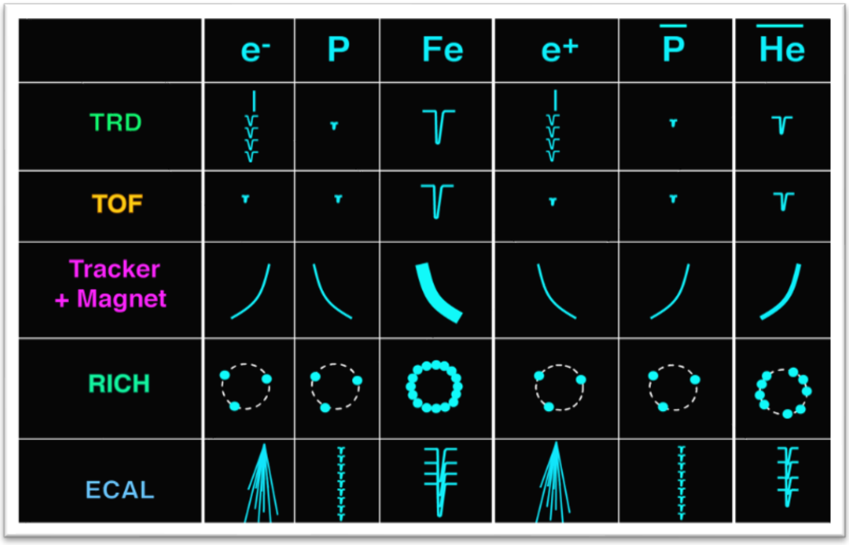


Figure 2.14: A representation of the signatures left by various types of cosmic rays as they pass through each of the subdetectors. Note that for protons (P) and positrons (e^+), all of the subdetector signatures are similar except for the ones given by the TRD and the ECAL.

2.3 Difficulty in Measuring Positron Flux Measurement at High Energies

The measurement of the cosmic positron flux is an important task needed to better understand a variety of astronomical phenomena [43]. This can include astrophysical events and objects such as gamma-ray bursts and pulsars [44][45], cosmic ray collisions with interstellar medium [46], and as Figure 1.1 shows, the annihilation of dark matter particles [6][17]. Towards this, AMS has published a variety of results focusing on their positron measurements [3][4][47][48], helping in the development of several models that attempt to explain the anomalous results [49][50][51][52].

At higher energies, however, the task of separating pure positrons from the abundant proton background becomes increasingly difficult. Figure 2.15 shows the AMS positron flux measurement up to 2021. At 284 GeV, the measured flux falls sharply with a power law of -2.744 ± 0.025. Between the energy ranges of 700-1000 GeV, the measured positron flux is $(1.927 \pm 1.087(\text{statistical}) \pm 0.444(\text{systematic})) \times 10^{-8}$ [m^2 sr s GeV]$^{-1}$. In contrast, between the energy ranges of 822-1130 GeV, the measured proton flux is $(9.068 \pm 0.029(\text{stat}) \pm 0.351(\text{sys})) \times 10^{-5}$ [m^2 sr s GeV]$^{-1}$

[4]. This results in there being approximately 4700 protons for each positron around 1 TeV with the ratio increasing at even higher energies.

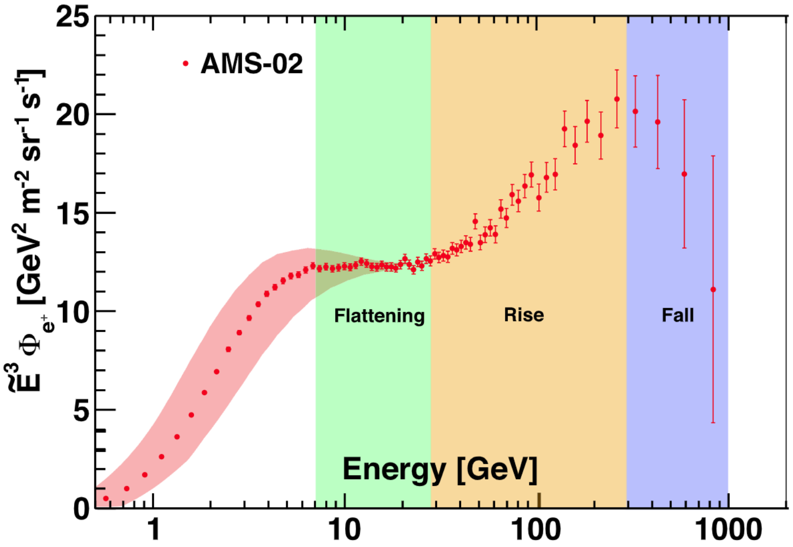


Figure 2.15: The AMS positron flux measurement up to 2021 [4]. The positron spectrum, $\tilde{E}^3\Phi_{e+}$, is shown as a function of energy, E, where $\tilde{E}$ is the spectrally weighted mean energy for a flux proportional to E^{-3}. The red band showcases the measurement variation as a result of solar modulation. The colored background depicts the changing behavior of the flux. The flux, which follows a power law, unexpectedly softens at 25.2 GeV before hardening again 284 GeV.

The large class imbalance problem and, as noted in Section 2.2.1, the TRD being unable to accurately separate positrons from the abundant proton background above 100 GeV, makes measuring a pure positron sample at TeV energies a difficult task that relies entirely upon the ECAL performance.

While the AMS LHD presents a clear improvement over the AMS BDT at 1 TeV, Figure 2.15 nonetheless shows large error bars present at high energies. This is due to the electromagnetic shower maximum occurring at deeper depths with increasing energy. With 17 radiation lengths, the ECAL can contain about 75% of a shower's energy for 1 TeV positrons or electrons [8]. At even higher energies, a rear-leakage occurs where the shower maximum starts to occur beyond the maximum depth of the ECAL. This results in the energy depositions and signal profiles looking increasingly similar to that of rarely showering protons, resulting in a weaker proton rejection and larger error bars. In terms of machine learning, this represents a domain shift between above and below 1 TeV particles.

Our goal is to investigate a deep learning approach to remedy these problems. Instead of using hand-crafted variables, we elect to use all the ECAL cells as inputs to our models to evaluate their effectiveness at extracting more information than the hand-crafted variables, and to measure their proton rejection power.

CHAPTER 3

MACHINE LEARNING

3.1 A Brief History

A relatively modern and commonly used explanation of machine learning (ML) is as follows:

“A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.” [53].

For our case, E is the process of a deep learning model analyzing signals in the ECAL generated by cosmic protons and electrons, T is particle classification, and P is proton rejection.

Coined in 1959 [54], the goal of machine learning is to build computer methods that use data to learn a particular task, such as image classification and natural language processing, without the need for hard coding decisions [53]. This is often done by "training" a model. In supervised learning, this process is carried out by labeled data, such as images, being passed to the model which then outputs a score. The score is then compared to the correct, i.e. ground truth, label using a loss function, which measures the "incorrectness" of a model. The output of the loss function is then used to train the model to perform better, often by updating certain weights that are multiplied by the input in order to get a score close to the ground truth. The "learning" refers to the weights being updated to values that give better overall results without the need for explicit weight assignment. Thus, the model can learn how to interpret data and make the correct choice, even when faced with a new sample.

The first instance can be traced to a mathematical model that was developed in 1943 as an attempt to replicate the functions of a human brain's neuron [55], as depicted in Figure 3.1. However, it was not until Rosenblatt's perceptron model in 1957 [56] that a practical method was developed. The perceptron was designed as a pattern recognition algorithm intended for image recognition, but it was quickly evident that it had many limitations, including being able to learn only linearly separable patterns and being unable to learn XOR functions [57]. It took almost a decade for multiple layers of perceptrons, known as a neural network, to be built in 1967 [58] in order to overcome the limitations of a single or single group of perceptrons. Unfortunately, computers at the time lacked the processing power needed to handle large neural networks [57] and interest in machine learning died down.

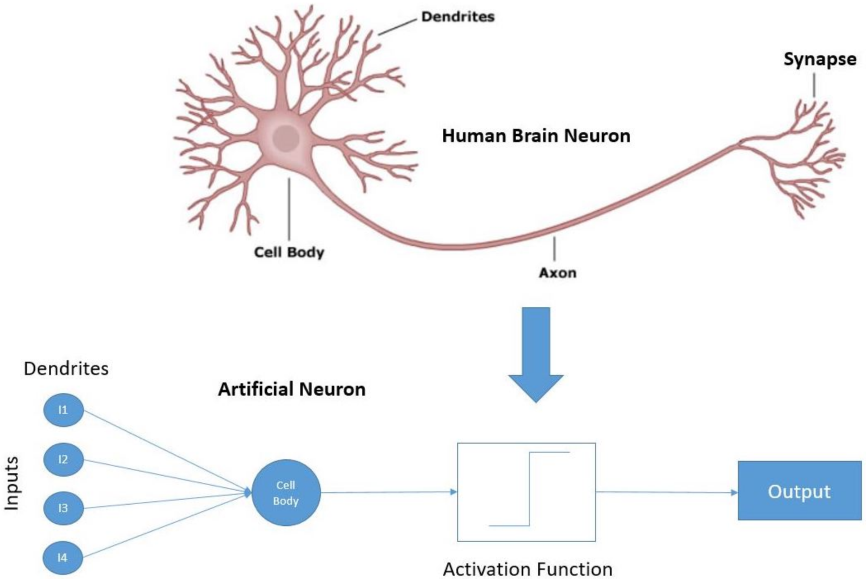


Figure 3.1: Similarity between an artificial neuron and a human brain's neuron [59]. The inputs act like dendrites, which are then processed and stored by the neuron. The human brain neuron then passes the information through the axon to the synapse, which then further feeds it along to the next neuron. In the computer replication, the artificial neuron undergoes an activation function before the final value is output, which can then be used as an input to the next neuron.

The development of backpropagation, coined in 1986 [60], restarted interest and development in machine learning. Backpropagation computes the gradient of the loss function with respect to the trainable parameters of the model using the chain rule from calculus. It then updates the weights in a manner that decreases the gradient, thereby finding a combination of weights that give a minimal loss value. While much of the theoretical development took place in the 1960s [60], it was first adopted for use in computers in 1970 [61] and analyzed for use in neural networks in 1974 [62]. Backpropagation allowed a more efficient method of training neural networks by calculating gradients of a loss function with respect to a given set of neural network weights. The success of the multilayer perceptron (MLP) model gave birth to universal approximation theorems [63] which proved that for any possible continuous mapping, there exist a neural network structure capable of replicating it, but the exact structure, number of neurons, and weights are not given by the theorems. In time, the popularity of multilayer perceptrons and neural networks died down in favor of statistical methods such as log-likelihood and support vector machines [64] that used task-specific, handcrafted features. This was in part due to computers not being fast enough to train neural networks, incorrect non-linear activation functions, and weights being incorrectly initialized.

In the early 2010s, neural networks in the form of deep learning made a comeback and saw a huge rise in popularity. While deep learning in the simplest definition means a neural network with multiple layers, the modern understanding is taken to be a neural network with a large number of layers consisting of a limited number of neurons (deep rather than wide networks) [65]. In 2009, Hinton and his team achieved state-of-the-art results in speech recognition by training each layer individually in an unsupervised manner before training all the layers together in a supervised manner [66]. In 2011, Ciresan and his team built a pattern recognizing deep learning model that achieved superhuman performance in a traffic sign recognition competition [67]. Most famously, the ImageNet dataset was created in 2009 [68] which provided a large amount of real world images enough to sufficiently train deep networks without overfitting. This enabled the development of AlexNet [69] in 2012, which held the standard for image classification for the next 3 years. AlexNet also popularized the use of Graphic Processing Units (GPUs) for parallel matrix multiplications, which were found to make training deep learning models incredibly efficient [70], kickstarting the deep learning revolution.

Since then, deep learning improved performance in a variety of fields, such as beating humans in Go [71], predicting protein structures [72], and text-to-image generation [73].

3.2 Applications of Machine Learning in Physics

Machine learning had a delayed introduction within the physics community, only recently gaining popularity in the 1990s. This was partly due to the strong preference of the physics community, especially in particle physics, to use statistics and well-defined physics models to make their inferences [74]. However, with the creation of large physics experiments and even larger datasets, understanding the underlying physics phenomenon or searching for rare and anomalous events became increasingly difficult. Many physicists saw the need to switch over to using machine learning or at the very least incorporate it into their analysis in some way. In the early 1990s, neural networks were commonly used to classify events and reconstruct jets [75][76][77]. But in the mid 2000s, the model of choice became the BDT [42][78] due to its more transparent nature [79], a feature very important to physicists who are accustomed to their models being completely predictable from a theoretical standpoint versus the "blackbox" nature that other machine learning models provide.

More recently, deep learning has been adopted by the particle physics community in increasingly large numbers [74][80][81] for the purposes of anomaly detection [82], event classification [83], event reconstruction [84], and simulations using GANs [85].

2022 saw the rise of transformers being used in a variety of fields in physics, such as particle transport [86], cross section regression [87], and jet-tagging classification [88][89]. The main deep learning model that we focus on is also a transformer, and more specifically, a Convolutional vision Transformer (CvT) [10].

3.3 Machine Learning Models Used

3.3.1 Basic Models

We first use basic machine learning models on selected MC events with a generated energy between 200–600 GeV (see Section 4.1) to evaluate their performance on the AMS ECAL dataset and provide an initial benchmark for our deep learning models. We use scikit-learn [90] to train and test 3 non-deep learning models: a basic logistic regression (LogReg) [91], a support vector machine (SVM) [64], and a histogram-based gradient boosting decision tree (HistBDT), which is a more efficient version of a BDT for larger datasets [92]. The default settings, optimizers, and learning rates were used. In addition, each model employed early stopping with a patience of 20 epochs of no improvement and a tolerance of 1.0e-5 on the validation loss.

3.3.2 Multilayer Perceptron (MLP)

The multilayer perceptron (MLP), also known as a feed-forward neural network, is one of the oldest and most commonly known architectures. MLPs essentially group a number of neurons into a layer that process information and pass it on to the next layer. The choice of deep (more layers) or wide (more neutrons in a layer) MLPs is still often debated [93][94]. The key element is the neuron, described in Figure 3.2. Each neuron takes in all the values of each neuron in the previous layer, multiplies them with a unique and trainable weights matrix, adds a bias value, and passes them to an activation function before outputting them for the next layer. Activation functions add nonlinearity to the neural networks, making them excellent function approximators. The choice of ideal activation function is an often researched topic [95][96][97], with a popular choice being the Rectified Linear Unit (ReLU) [98].

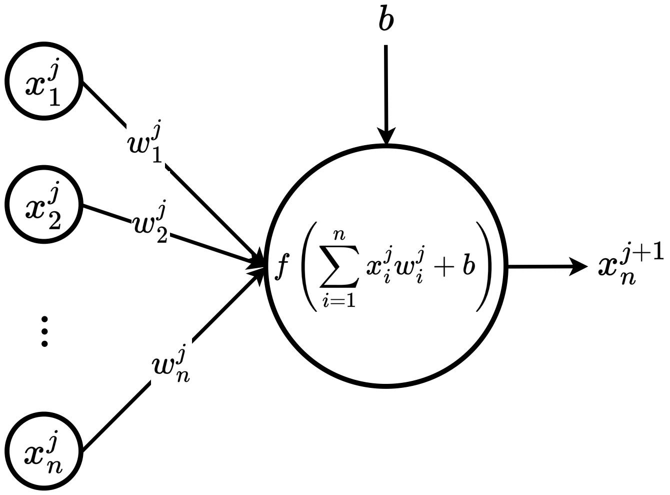


Figure 3.2: Schematic of the artificial neuron used in neural networks. Each neuron takes an input, x_i, from either another neuron in layer j or from the data/image itself. In the case of images, this can be pixels. All the inputs are multiplied by a weight parameter, w_i^j, that is constantly updated each training cycle and then summed together. A trainable bias parameter, b, is then added to to help offset results if need be. The result is then passed to an activation function, $f(\cdot)$, before being sent to the next layer or treated as the final output.

Our particular multilayer perceptron (MLP), denoted as SimpleMLP, is shown in Figure 3.3. It consists of 4 hidden layers that take either a $2 \times 18 \times 72$ or $1 \times 18 \times 72$ input (see Section 4.1.1) and outputs a single classification score.

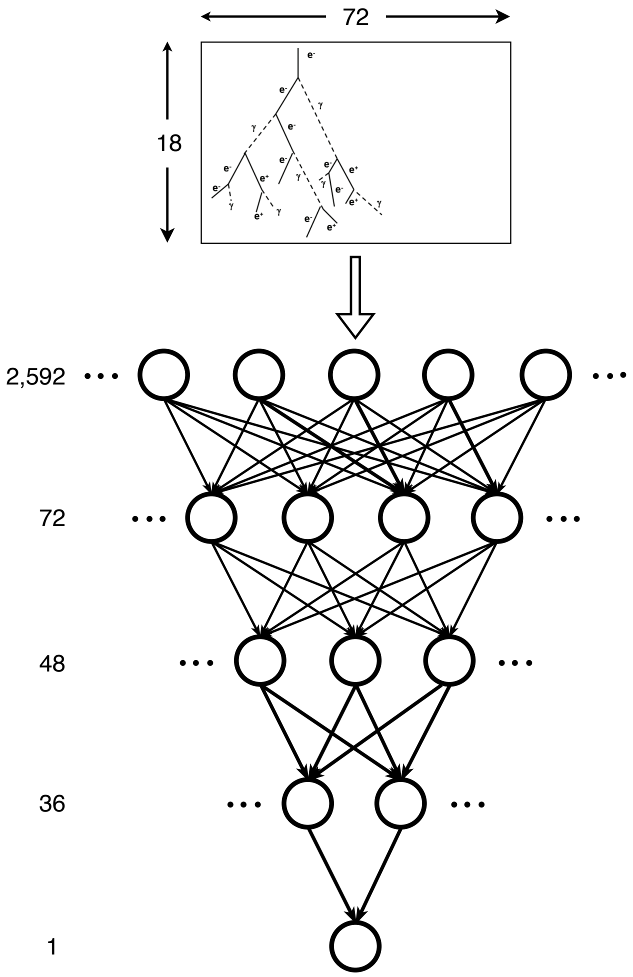


Figure 3.3: The architecture of our SimpleMLP model, with the number of neurons in each layer printed on the side.

3.3.3 Convolutional Neural Network (CNN)

The convolutional neural network (CNN) is one of the most popular models for image classification and has provided many breakthroughs in image, video, speech and audio processing [65]. The key feature is the convolution, shown in Figure 3.4 for the case of an image. For each channel in the input, a different kernel containing weight parameters slides over the image, performs element-wise multiplication, and sums the output of each channel's kernel into a single pixel. Depending on the number of output channels or feature maps desired, this process is repeated multiple times, with each kernel have different weights. Convolutions are vastly more efficient than MLPs because a single set of shared weights in a kernel are repeatedly used throughout the image, reducing the overall number of trainable parameters and regularizing the network for better generalization. The weights in a kernel are essentially trained to identify particular features that activate the kernel (i.e. give the largest output) and give the largest model score for a particular class. Thus, each kernel "searches" for a pattern it can identify, making CNNs translation (shift) equivariant.

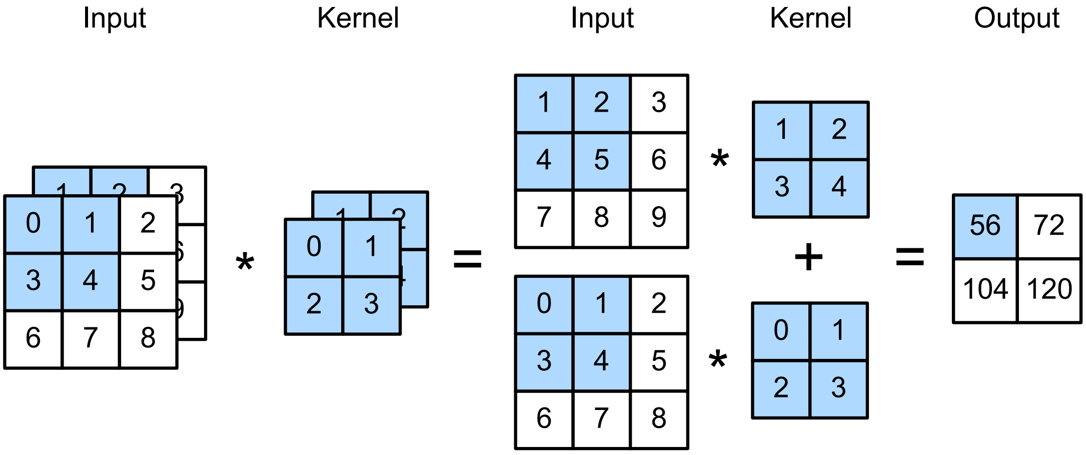


Figure 3.4: Example of a convolution [99] for an image. For each channel in the input, a different kernel containing weight parameters slides over the image, performs element-wise multiplication, and sums the output of each channel's kernel into a single pixel. Depending on the number of output channels or feature maps desired, this process is repeated multiple times, with each kernel have different weights.

Our convolutional neural network (CNN), denoted as SimpleCNN, is shown in Figure 3.5. It consists of 3 convolutions, 2 max pooling layers [100], and 2 linear layers that take in either a $2 \times 18 \times 72$ or $1 \times 18 \times 72$ input (see Section 4.1.1) and outputs a single classification score.

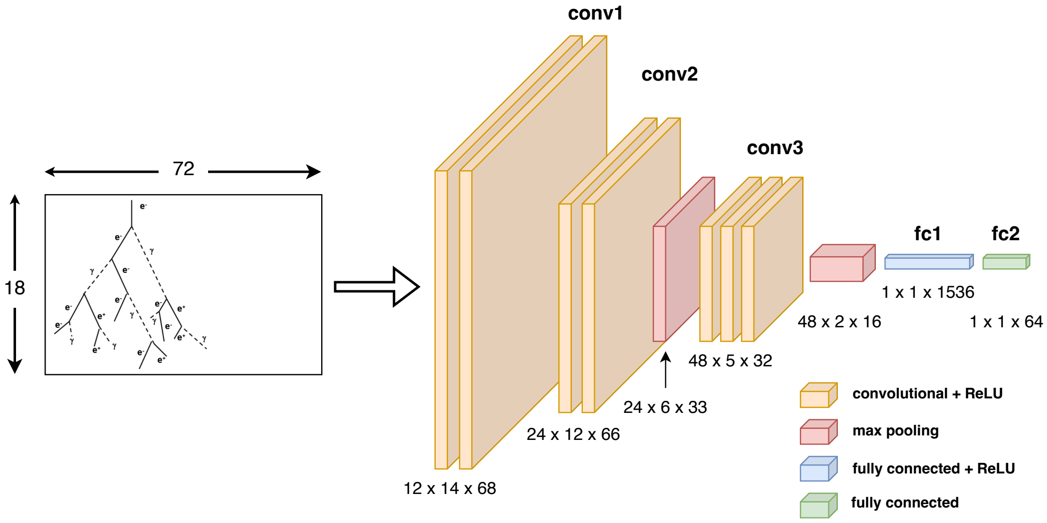


Figure 3.5: The architecture of our SimpleCNN Model. The dimensions of the resulting feature map are listed below each corresponding step.

3.3.4 Residual Neural Network (ResNet)

The residual neural network (ResNet) [9] is a CNN architecture that employs skip connections to allow training of very deep networks. As the number of layers increases, neural networks are subject to vanishing gradients [101]. ResNets overcome this by adding the output of previous layers to future layers, known as a skip connection, for certain layers. Thus, if during backpropagation the partial derivative of a layer's weight would be close to 0 without skip connections, the addition of a previous layer's output would prevent the partial derivative from "vanishing" and allow gradients to flow more easily back through the network. In essence, this allows the network to skip entire layers, adding regularization and simplifying the network.

We use two variants, the ResNet10 and ResNet18, shown in Figure 3.6. For the ResNet18 variant, only the initial and final layers were modified to take in either the $2 \times 18 \times 72$ shaped or $1 \times 18 \times 72$ shaped input and output to 1 neuron for binary classification. The ResNet10 model was custom built to create a model with half the number of trainable parameters, in order to make the number of parameters equivalent to the CvT's number of trainable parameters (see Table 3.2).

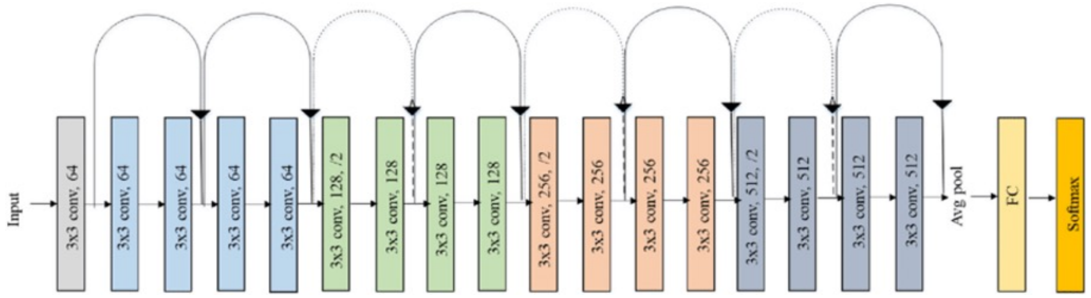


Figure 3.6: The architecture of the ResNet18 Model we used [9].

3.3.5 Convolutional vision Transformer (CvT)

The Convolutional vision Transformer (CvT) [10], shown in Figure 3.7, is a transformer model, which is an architecture that processes a connected set of units, or tokens, using self-attention. These connected set of units can be a sequence of words, such as for natural language processing [102], or pixels in an image [103]. The CvT model first takes in an input image and passes it to a convolutional layer (Convolutional Token Embedding) which reduces the resolution, increase the depth, and extracts feature vector tokens. The output is passed to the convolutional transformer block (Figure 3.7), where convolutions are used to build the Q, K, and V matrices that are then passed to the attention mechanism, where they are combined via the scaled dot-product attention,

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3.1}$$

where Q (Query) represents the features the attention mechanism is focusing on and K (Key) is a representation of each element of the feature maps. Their dot product gives us the attention scores, which measure the similarity between the query and each element of the feature maps and are used to weigh the contribution of each element to the final output of the attention mechanism. The scores are scaled by $\sqrt{d_k}$ to normalize the outputs, where d_k is the dimension of K. We then use the scaled attention scores to weigh V (Value), which represents the attention mechanism's outputted feature maps. In doing so, the attention mechanism can learn to focus on regions of the feature map that are most discriminative for a class.

The final output of the attention block is then passed through an MLP before being reshaped into a 2D feature map and passed to the next stage. At the final stage, a classification token (cls) is added, which is trainable feature vector that extract features most relevant to the classification task, before being passed to an MLP for classification.

The CvT combines the benefits of CNNs with that of Transformers, such as better generalization, better focus on key areas and global context. This is important for ECAL showers as shower images can appear differently depending on the type of particle, energy of the particle, the angle of incidence and the point of entry.

We modified our implementation to better suit the AMS ECAL data. Fellow METU student Berk Türk determined our particular configuration by modifying the hyperparameters until a consistent decrease on the loss function was noted, and informed us via personal communication. Our final choice of configuration uses 4 transformer blocks, with Stage 3 having 2 blocks. For the three stages, there are 1, 3, and 6 attention heads, the kernel sizes chosen were 2×6, 3×4, and 3×3, and the stride lengths were 1×2, 1×2, and 1×1, respectively. For the Phys+CvT (small kernels) variant (see Section 5.5), in an attempt to better suit the more narrow dataset, the kernel sizes chosen were 3×1, 4×3, and 3×3, and the stride lengths chosen were 1×1, 2×1, and 1×1, respectively. We were unable to test additional configurations for this variant due to time limitations.

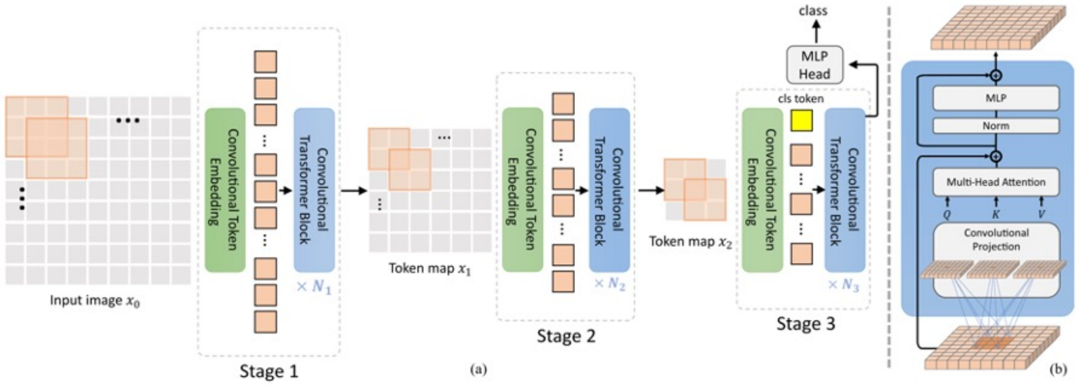


Figure 3.7: The pipeline of the Convolutional vision Transformer architecture. (a) Overall architecture, showing the hierarchical multi-stage structure facilitated by the Convolutional Token Embedding layer. (b) Details of the Convolutional Transformer Block, which contains the convolution projection as the first layer [10].

3.4 Training Procedures and Configurations

Certain hyperparameters were shared between all the DL models for fairness. These can be seen in Table 3.1. They were chosen based on the the best practices as of 2022. For each model, we searched for the optimal learning rate using grid search on a log-scale, and 1.0e-4 was found to be the best for all models. During training, early stopping was always employed so that after a number of epochs with no validation loss improvement (patience) greater than the tolerance value, we would stop the training and save the trained model with the best score on the validation set. Each model's learning capacity (the total number of trainable parameters) can be seen in Table 3.2 to give a better idea of the complexity of the models. This later comes in to play in Section 5.5, when we evaluate the performance of the ResNet10 against the ResNet18.

For the datasets containing selected MC events with a generated energy between 200–600 GeV and selected ISS data events with a reconstructed energy between 50–70 GeV, the standard ML training, validation, and testing procedure was used. The datasets underwent a 60/20/20 split into train/validation/test sets, where the models were trained on the train set and evaluated on the validation set after each epoch. The trained model with the best validation set performance is then tested on the test set using a variety of performance metrics (see Section 4.4).

For datasets containing selected MC events with a reconstructed energy between 200 GeV to 2 TeV, a slightly modified version of the standard train/val/test procedure was used. The datasets were first split into events with reconstructed energy below 1 TeV and events with reconstructed energy above 1 TeV. The below 1 TeV set was then split 50/50 into a train/test set while the above 1 TeV was split 50/50 into a validation/test set. As such, training was conducted exclusively on below 1 TeV particles while validation was performed on above 1 TeV particles. Our goal was to train models that were less dependent on the total energy. By only training on lower energies, models that regularized, generalized and focused more on the shower shape would perform better on the higher energy events in the unseen validation set. After training, the models are tested on both the above and below 1 TeV test sets using the aforementioned performance metrics, namely, the proton rejection vs. electron efficiency.

For most of the trainings and testings, the IVMER DL workstation was used. Occasionally, the ImageLab DL Node 3 was used when IVMER DL was unavailable. The hardware configurations of both setups can be seen in Table 3.3. On average, the CvT models and ResNet models would take 48 and 18 hours to train, respectively, before early stopping was triggered on these setups.

Table 3.1: Training hyperparameters for the deep learning models.

Hyperparameter	Value
Batch Size	128
No. of Workers	4
Loss Function	Weighted Binary Cross Entropy with Logits
Activation Function after Last Layer	Sigmoid
Optimizer	Adam
Learning Rate for Optimizer	1.0e-4
Early Stopping Patience	20
Early Stopping Tolerance	1.0e-5

Table 3.2: Total number of trainable parameters for each of the deep learning models.

Model	Trainable Parameters
SimpleMLP	192,001
SimpleCNN	106,317
ResNet10	4,900,033
ResNet18	11,170,753
CvT	4,895,873
Phys+CvT	4,896,641

Table 3.3: List of core components used to train the deep learning models.

Type	IVMER DL	ImageLab DL
GPU	2 x Nvidia RTX 2080Ti	2 x Nvidia RTX 2080Ti
CPU	Intel Core i9-10980XE CPU @ 3.00GHz	Dual Intel Xeon CPU E5-2630 v3 @ 2.40GHz
RAM	256 GB DDR4	189 GB DDR4

CHAPTER 4

ECAL SHOWER CLASSIFICATION USING DEEP LEARNING

4.1 AMS ECAL Dataset

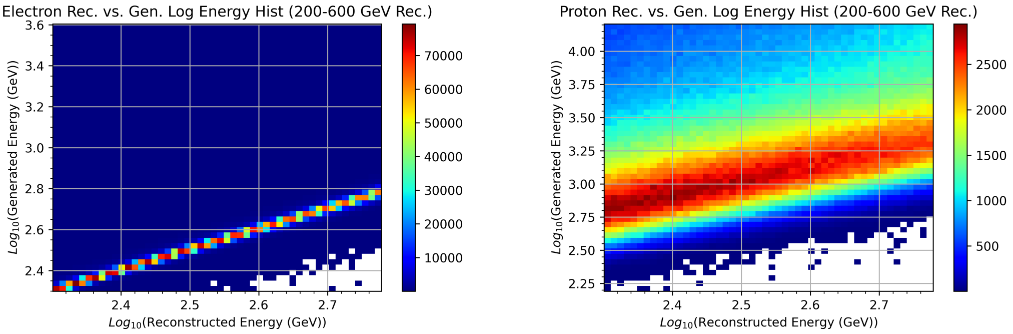
As particles pass through the AMS's subdetectors, they generate electronic signals that are packed into frames, which contain all data collected during the same minute. These one-minute frames are transferred by the Tracking and Data Relay Satellites to White Sands, New Mexico, before being transferred to the AMS Payload Operations Control Center (POCC) at CERN. There, frames from a quarter of the ISS orbit ($\sim$23 minutes) are labeled as one run, and data from each run is repacked into an AMS RAW file which is then converted into an AMS ROOT file and stored on CERN's EOS servers [104]. Monte Carlo (MC) generation follows a similar process, where subdetector signals are simulated but then are directly saved in the AMS RAW format before being converted into AMS ROOT files using the same process.

These AMS ROOT files can be accessed using a specially made AMS Framework that adds on to the CERN ROOT software [105] for use with AMS's detectors and analysis techniques. An unfortunate byproduct of this is the slow updates to the AMS ROOT software, which prevents it from getting the latest features pushed by CERN ROOT, such as v6.22's better Python integration and deep learning support [106].

To remedy this, experts at AMS use AMS ROOT to extract the necessary variables and data they need from the AMS ROOT files and pack it into ordinary CERN ROOT TTree files, which can then benefit from the Python integration. In order to do this ourselves, we prepared a script that parses through any given AMS dataset, applies specific event selection cuts and thresholds, and saves those events into our own CERN ROOT TTree files. These cuts and thresholds represent the energy range, the type and value of the particle's charge, number of showers in an event, whether the particle passes through all the subdetectors, and more (see Appendix A for a list of acceptance cuts used in our experiments).

Since electrons and positrons are not separated by the ECAL (see Section 2.2.5), this thesis uses electrons, which have higher statistics in space, to train our DL models.

We performed the data extraction process four times as progress was made and new datasets were required, with the first three being on MC datasets and the last being on ISS data. The energy cuts of the first data extraction was based on generated energy while the other three were based on reconstructed energy. Generated energy is the original energy of the particle before it passes through the AMS subdetectors. This can only be known through MC simulations (see section 4.2), where every property of the particle is specifically selected by the experts at AMS. In contrast, reconstructed energy is the energy the ECAL subdetector calculates from the signals left by the particle. When analyzing ISS data, only reconstructed energy can be used. A 2D histogram we made to compare the two energy types for particles with a reconstructed energy between 200–600 GeV can be seen in Figure 4.1. Electrons have a better correlation between generated and reconstructed energies while the protons have more variation. This is partially due to the fact that only high energy protons shower and deposit a small portion of their generated energy, so their reconstructed energies are not always closely correlated to their original generated energies.



(a)

(b)

Figure 4.1: 2D Histogram showing the correlation between generated and ECAL reconstructed energy. Electrons (left) have a better correlation between the two while the protons (right) have more variation. This is partially due to the fact that only high energy protons shower and deposit a small portion of their generated energy, so their reconstructed energies are not always closely correlated to their original generated energies.

The first dataset we extracted consists of MC particles with a generated energy between 200–600 GeV. At this energy range, there is a large difference between proton and electron showers, making it easier for classifiers to differentiate between the two. This dataset was mainly used to build the AMS ROOT ECAL-to-Machine Learning (ECAL-to-ML) pipeline and to make early comparisons between the currently used AMS models, simple ML models, and our DL models.

The second and third dataset we extracted consists of particles with a reconstructed energy between 200 GeV–2 TeV, which would make classification more difficult and allow us to train models with a reduced dependence on energy (see Section 3.4). The third dataset, while having the same cut on energy, included additional variables from the Tracker to perform the physics-based feature engineering preprocessing (see Section 4.3). In the time between the two extractions, additional MC was produced, which increased the total number of events we extracted (see Table 4.1).

Finally, the fourth dataset we extracted was from ISS data and consists of particles with a reconstructed energy between 50–70 GeV. In order to compare our DL models to the AMS models in a fair manner, only the TRD and Tracker are used to separate

Table 4.1: Number of events (i.e. images) for each of the datasets.

Source	Dataset	Below 1 TeV (in Millions)		Above 1 TeV (in Millions)	
		Electrons	Protons	Electrons	Protons
MC	200-600 GeV, Generated	4.60	0.16	0	0
MC	200-2000 GeV, Reconstructed	7.02	3.90	2.69	1.19
MC	200-2000 GeV, Rec. + Tracker Variables	7.51	3.98	2.89	1.21
ISS	50-70 GeV, Reconstructed	0.03	1.19	0	0

electrons from protons. For this reason, along with the data being ISS data instead of MC, additional cuts and steps were used to extract the dataset. To get a large and relatively pure sample of electrons and protons with correct labeling, 50–70 GeV was selected because the TRD can more reliably separate electrons from protons at this energy range and because of the increased number of particles in this range present in space [4].

The event selection cuts and criteria applied for all four datasets can be found in Appendix A. The total number of particles (i.e. images) for each of the four datasets can be found in Table 4.1.

4.1.1 AMS ROOT ECAL-to-Machine Learning Pipeline

The AMS ROOT framework organizes every signal into individual classes, which can then be accessed by certain variables. Figure 4.2 shows how ECALHitR, the object containing ECAL hits, is stored and accessed. Data on an a cosmic ray from all the subdetectors is contained within the AMSEventR object, which itself is contained under a parent object called Entries. Entries are linked sequentially to each other and form an array of objects within another parent object called AMSChain. Once access to AMSEvenR is established, various functions can be used to retrieve child objects that contain information from different subdetectors. For the ECAL, this object is known as ECALHitR, which contains variables that tells us information about a singular cell's energy deposition, i.e., the layer number, the cell number, and the energy deposition value. For any given particle event, there can be up to 1296 EcalHitR objects, one for each of the 18×72 cells in the ECAL.

The script we prepared applies the cuts and thresholds necessary (see Appendix A) and saves variables such as ECALHitR into a CERN ROOT TTree. Once this process is complete, we transfer the TTrees from the CERN EOS servers to a local computer for further processing.

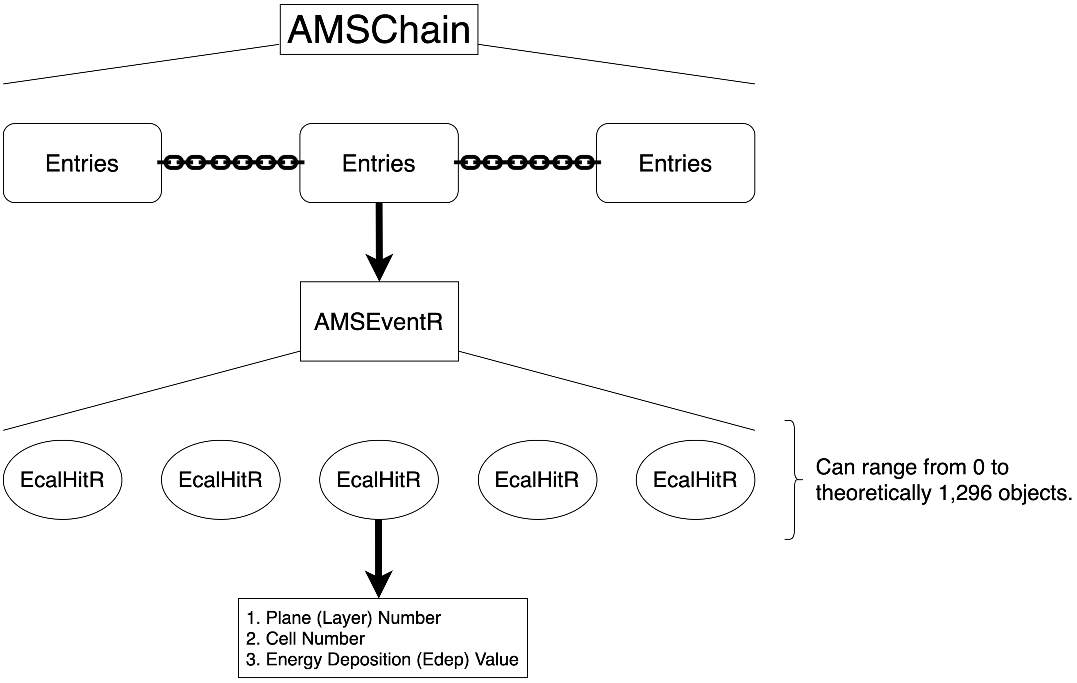


Figure 4.2: A simplified representation of how ECALHitR, the object containing information on the ECAL cells energy deposition, is accessed from AMS ROOT.

On our local computer, each event in the CERN Root files is parsed using PyRoot [107], where the ECALHitR information is reshaped into a 2D or 3D matrix and the additional variables are packed alongside it into a NumPy [107] file. This NumPy file is then read into PyTorch's [108] dataloader function, where it becomes ready to use for machine learning. Alternatively, it can directly be read for machine learning with scikit-learn's simple ML models. Figure 4.3 shows a representation of how the ECAL hits are reshaped into either a 2D ($1 \times 18 \times 72$) or 3D matrix ($2 \times 18 \times 72$). While the 3D matrix was used initially, a switch was made to 2D later on (See Section 5.3).

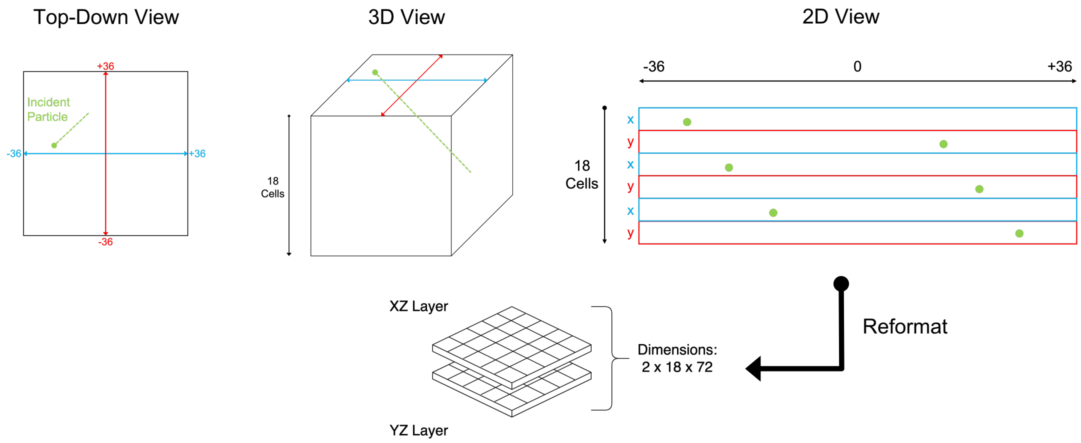


Figure 4.3: A representation of how the ECAL hits are converted into a machine learning format. A visual of a particle progressing through the ECAL is shown in the Top-Down and 3D view. Due to the structure of the ECAL's scintillating fibres (see Section 2.2.5), for every given layer either the X or the Y direction is recorded, leading to the 2D view shown with shape $1 \times 18 \times 72$. This representation can then further be reformatted into a $2 \times 18 \times 72$ shape, where the X and Y channels are separated, to better resemble an image with multiple channels.

4.1.2 Shower Visualization

To visualize the $2 \times 18 \times 72$ ECAL shower images, we built a viewer function that utilizes both Matplotlib [109] and Seaborn [110]. This allowed an understanding of the general shape of the shower, which helped determine other properties later on such as kernel size and how feature engineering should be performed. Figure 4.4 shows examples of a proton and electron of less than 1 TeV reconstructed energy passing through the ECAL. To better understand the general layer where most of the shower energy, or shower maximum, is deposited, averages of the shower were taken over the entire 200–2000 GeV, reconstructed dataset, which can be seen in Figures 4.5 and 4.6. The blank spots represent dead PMTs. A clearer difference is discernable in the former with the electrons depositing more energy in layers 8-13 and the protons depositing more energy in layers 14-17. In the latter, due to their higher energy, electrons travel a larger distance and shower maximum is reached at layers 10-15 (for energies up to 2 TeV), which overlaps with the protons' shower maximum, which is still at layers 14-17. This gives us a visual evidence of how, as the energy increases, the ECAL layers become insufficient in capturing the full shower (see Section 2.2.5) and how the signals of electrons start to look similar to the signals of protons.

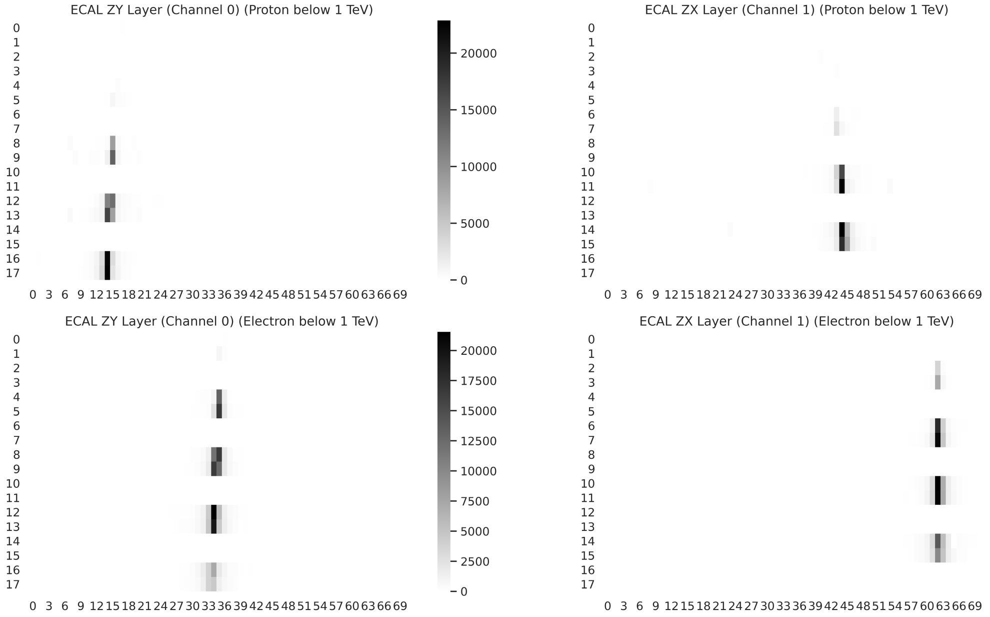


Figure 4.4: Examples of a proton (top) and electron (bottom) passing through and showering in the ECAL.

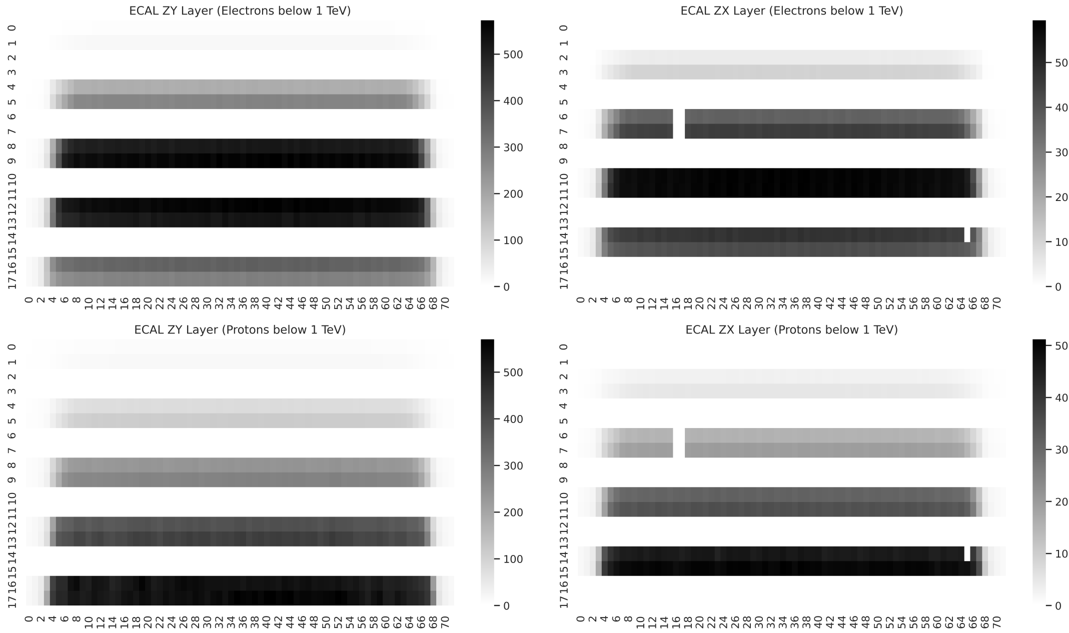
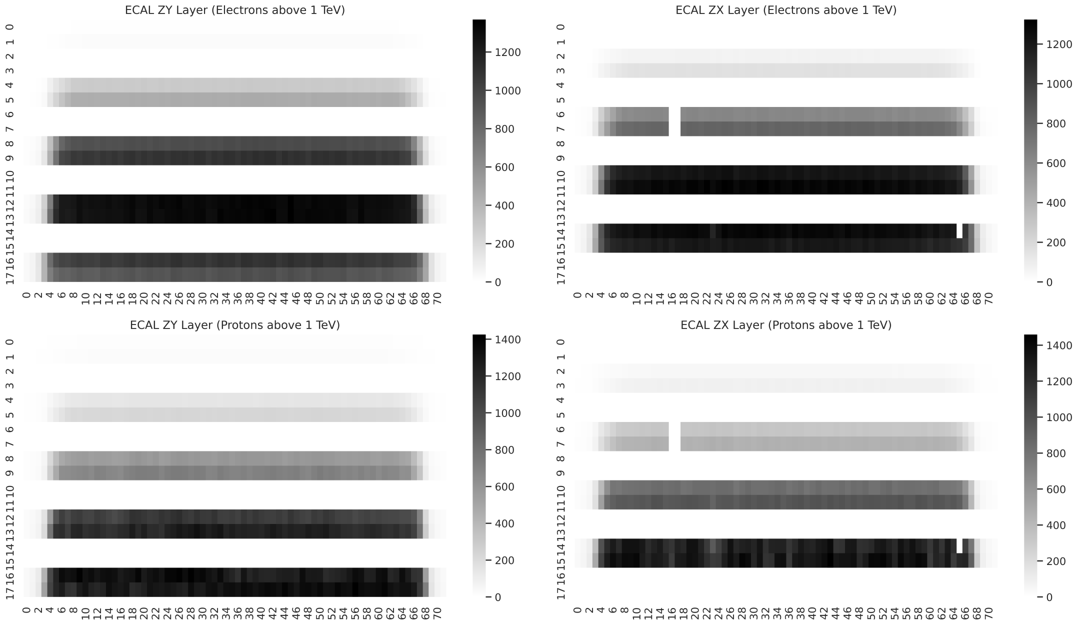


Figure 4.5: Averages for the entire dataset of electrons (top) and protons (bottom) below 1 TeV. For electrons, it can be seen that most of the energy is deposited in layers 8-13 while for protons it is deposited in layers 14-17.


ECAL ZY Layer (Electrons above 1 TeV)

ECAL ZX Layer (Electrons above 1 TeV)

ECAL ZY Layer (Protons above 1 TeV)

ECAL ZX Layer (Protons above 1 TeV)

Figure 4.6: Averages for the entire dataset of electrons (top) and protons (bottom) between 1-2 TeV. For electrons, it can be seen that most of the energy is deposited in layers 10-15, a shift of about 2 layer, while for protons it remains the same at layers 14-17. This gives us an indication that, as the energy increases, the electron shower maximum reaches lower and lower layers, eventually making it difficult to discern from protons.

4.2 Monte Carlo Events and ISS Data

ISS data contains noise and other non-ideal conditions such as multiple particles entering the detector at the same time and detector aging effects.

When comparing models, it is customary in high energy physics to first use the Monte Carlo (MC) method [111]. MC events form a simulated dataset often used to design detectors, optimize settings, create interactive simulations, perform data analysis, and conduct method evaluation. They allow for the control of initial parameters such as the type of particle, generated energy, and the direction and position along which the particle passes through the simulated detector. MC events help determine the best detector geometry configuration to be sensitive to a physics signal with a certain precision. In addition, they often represent the ideal situation that ML models can be tested on before being evaluated on data.

In AMS, MC events are produced using a dedicated program developed by AMS [112] based on a modified version of Geant4 v10.3 [113]. However, currently unquantifiable discrepancies exist between MC events and ISS data protons due to the inaccurate modeling of hadronic showers, which might potentially represent a second domain shift, one between MC and ISS data.

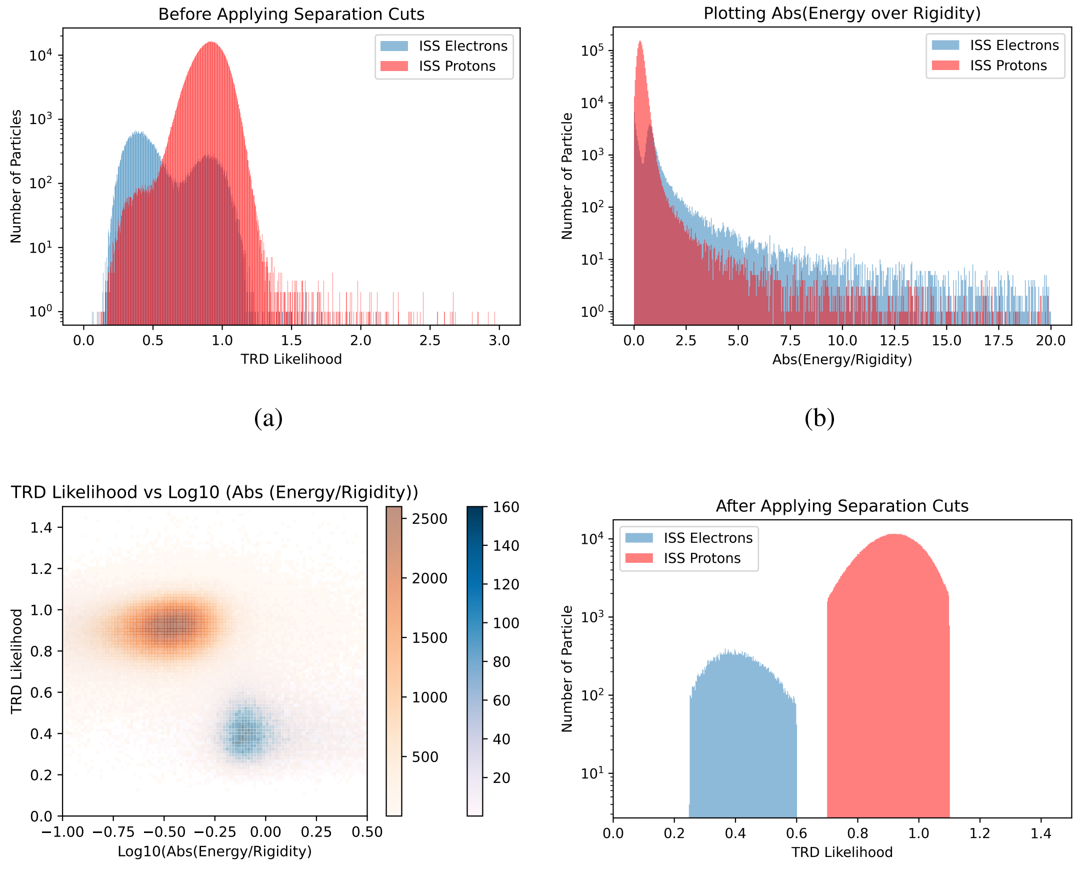
4.2.1 ISS Data Extraction

To test our MC-trained models and train on ISS data itself, we extracted ISS data and used the TRD and Tracker to separate electrons from protons. As explained in Section 2.2.1, the TRD in unable to accurately separate the two at high energies above 100 GeV. In order to ensure a pure sample of electrons and protons with accurate labeling, we only extracted particles with a reconstructed energy between 50–70 GeV.

Figure 4.7 depict the steps taken to separate electrons from protons. Due to the abundance of matter compared to antimatter, positively and negatively charged particles were considered and labeled as ISS protons and electrons, respectively. The TRD likelihood shows some separation of electrons from protons, as Figure 4.7a shows. Some overlap between positively and negatively charged particles are expected due to charge-confused protons [114], antiprotons, and positrons. The TRD likelihood is also highly dependent on energy and additional cuts on both energy, taken from the ECAL, and rigidity, taken from the Tracker, are required. The ECAL only accurately measures the energy of electrons and not protons, on account of it being designed for electromagnetic showers (see Figure 4.1). The Tracker, however, can accurately measure the rigidity of both the electrons and protons well. As such, $|\frac{\text{ECAL Energy}}{\text{Tracker Rigidity}}| = 1$ for electrons, but should be lower for protons.

Figures 4.7b shows a histogram of $|\frac{\text{ECAL Energy}}{\text{Tracker Rigidity}}|$ and Figure 4.7c shows a 2D histogram of the TRD Likelihood vs. $\log_{10}|\frac{\text{Energy}}{\text{Rigidity}}|$, where a clear separation of positively and negatively charged particles can be seen. Here, we note that electrons are indeed closer to $|\frac{\text{ECAL Energy}}{\text{Tracker Rigidity}}| = 1$.

Figure 4.7d, shows a selection of protons and electrons after the energy and rigidity cuts. The number of events in this ISS dataset can be seen in Table 4.1.



(a)

(b)

(c)

(d)

Figure 4.7: Steps taken to separate ISS electrons from protons for the reconstructed energy range of 50–70 GeV. Due to the abundance of matter compared to antimatter, positively and negatively charged particles were labeled as ISS protons and electrons, respectively. The charge sign was determined using the Tracker. (a) Histogram of the TRD Likelihood. (b) Histogram of $|\frac{\text{Energy}}{\text{Rigidity}}|$ (c) 2D histogram of TRD Likelihood vs. $\log_{10}|\frac{\text{Energy}}{\text{Rigidity}}|$. (d) Histogram of the TRD Likelihood after learned cuts (filtering) on TRD Likelihood, energy, and rigidity is made.

4.3 Physics-based Feature Engineering

We developed a physics-based feature engineering method, shown in Figure 4.8, after evaluating on the second dataset (containing particles with a reconstructed energy between 200 GeV–2 TeV) and studying the performance of the CvT model with varying amounts of training data (see Section 5.4). Our goal was to use physics principles to make our CvT more efficient at learning on smaller training sets.

This feature engineering extracts the 20 pixels surrounding the projected line of the particle's shower through the ECAL. To do this, spherical polar coordinates are used as a method to calculate the projected shower line using the Tracker variables. First, the point of incidence is taken as the origin with coordinates $(0, 0, 0)$. Then, the following three spherical coordinate equations are used

$$x = r \sin\theta \cos\phi \tag{4.1}$$

$$y = r \sin\theta \sin\phi \tag{4.2}$$

$$z = r \cos\theta \tag{4.3}$$

Where r is the radius from the defined origin, θ is the zenith, ϕ is the azimuth, and x, y, and z are the distances from the defined origin. Since the distances between the ECAL layers are defined and well known, they are directly used as values for z. To learn the coordinates of where the projected shower line will be for a given z, the fixed Z value of the layer and the incident zenith is substituted into Eq. 4.3 to get r, which is subsequently used to find both x (Eq. 4.1) and y (Eq. 4.2) for that particular layer. Since x, y, and z are distances from the defined origin, they are added to the Tracker's given X-, Y- and Z-coordinates to get the actual distances. A simple algorithm is then used which converts these distances into their pixel equivalent to get the projected cell number for the Tracker extrapolated 3D line.

Finally, the projected pixel and 10 pixels from both the left and right sides are extracted and made into another dataset of dimensions $1 \times 18 \times 21$. A second channel containing the actual depth crossed by the particle was also added. This was done by dividing the vertical distance of the z layers by $\cos\theta$ to get the diagonal, true distance. Thus, the final dimension of the new feature engineered dataset is $2 \times 18 \times 21$.

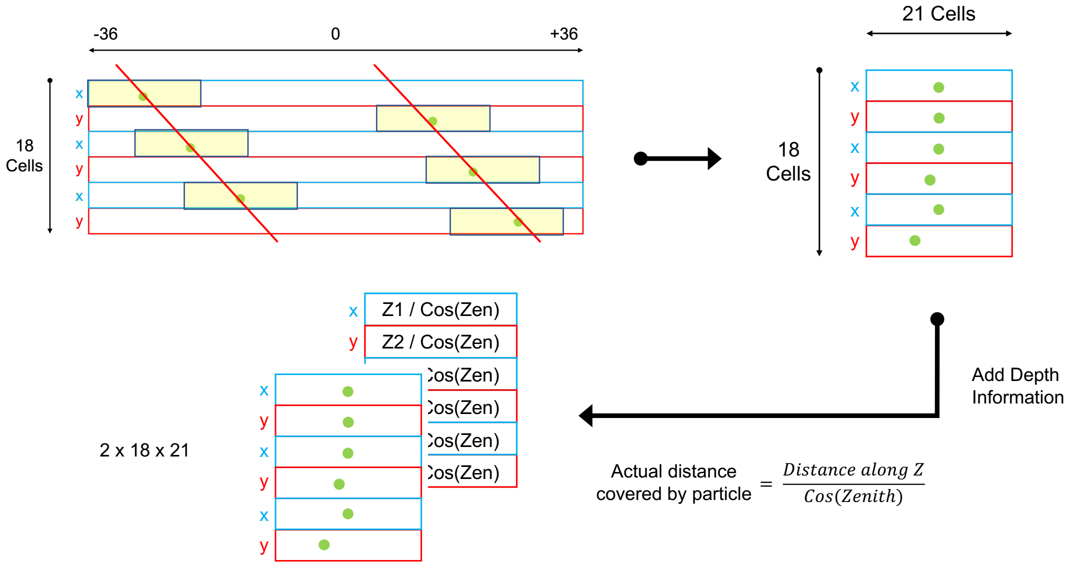


Figure 4.8: The physics-based feature engineering method. Using 5 variables from the Tracker (X, Y, Z, Zenith, and Azimuth of the particle incident on the ECAL), we project the path that the incident particle will follow (red lines). We then extract 10 pixels surrounding the projected locations in each layer (highlighed rectangles), create a 1 x 18 x 21 cell dataset, then add depth information as a second channel, resulting in a 2 x 18 x 72 shape. The depth information consists of the actual depth through the material crossed by the incident particle.

4.4 Performance Metrics

Alongside loss and accuracy curves, we initially use the commonly-used receiver operating characteristic (ROC) curve [115] as a performance metric. As will be shown in Sections 5.1 and 5.2, ROC curves cannot distinguish between the ML models. This motivated us to use performance metrics that are commonly used in high energy physics: proton rejection for a given electron accuracy, denoted as electron efficiency.

Background rejection is a commonly used method to evaluate machine learning models for high energy physics experiments [116][117][118]. For our work on the AMS ECAL, protons are considered the background [40].

The process to find the misidentified protons for a dataset given a certain electron efficiency, $E_{\%}$ is shown in Figure 4.9. The dataset is first split into its constituent proton and electron datasets. The electrons are passed through the model being tested and the model's score for each electron event is recorded. Electron efficiency is the percent of electrons that are correctly classified given a specific model score cutoff, $E_{\%}$. Then, the protons are passed through the model and each proton's score is recorded. For a given $E_{\%}$, proton events that score above this value are classified incorrectly as electrons and are considered misidentified protons. The proton rejection for a given electron efficiency is then calculated as:

$$\text{Proton Rejection} = \frac{\text{Total Number of Protons}}{\text{Number of Protons Misidentified}} \tag{4.4}$$

This process is repeated for every electron efficiency from 1% to 100% and proton rejection versus electron efficiency figures are plotted. Additionally, proton rejection versus reconstructed energy is also plotted. For a given percent of electron efficiency, often 60% and 90%, the proton rejection for each energy bin of the dataset is plotted to give an in-depth understanding of which energy bin the models perform poorly.

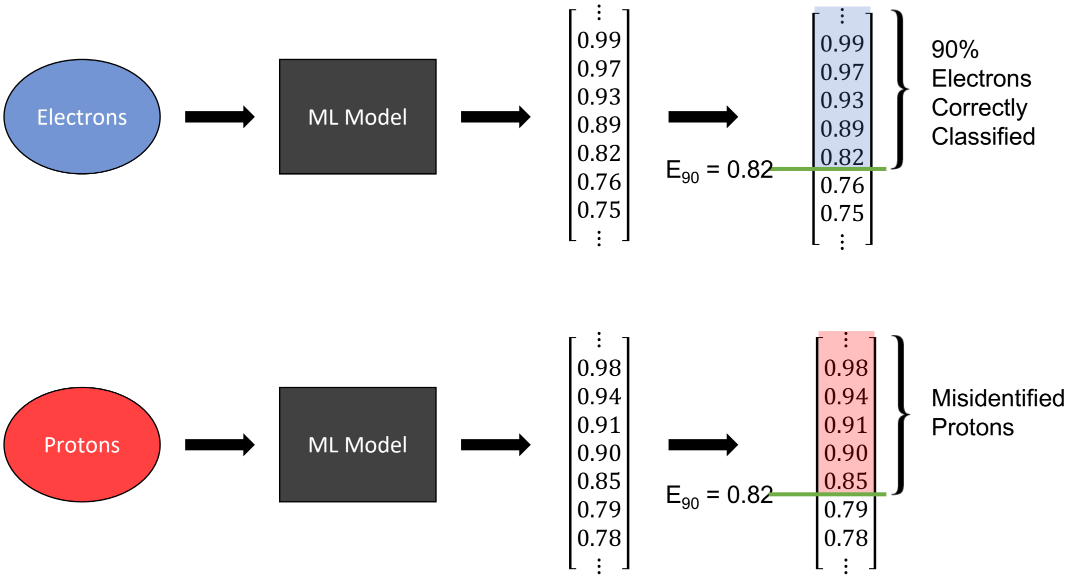


Figure 4.9: The process of how misidentified protons are calculated. The dataset is first split into its constituent proton and electron datasets. The electrons are passed through the model being tested and the model's score for each electron event is recorded. Electron efficiency is the percent of electrons that are correctly classified given a specific model score cutoff, $E_{\%}$. Then, the protons are passed through the model and each proton's score is recorded. For a given $E_{\%}$, proton events that score above this value are classified incorrectly as electrons and are considered misidentified protons.

CHAPTER 5

EXPERIMENTS AND RESULTS

5.1 Initial Comparisons with Simple ML Models

Comparison of the AMS models against simple ML models and DL models (except for the CvT which was not used yet) was done on the first dataset containing MC events with a generated energy between 200–2000 GeV. Figure 5.1 shows the generated energy histograms of the train/val/test sets, to assure that energy distribution and class imbalance remains similar between the three sets. Since the events were generated using a power law similar to that observed in space, there are more lower energy events than higher energy events, making right-skewed histograms. Table 4.1 shows the detailed statistics for the number of protons and electrons. While there is a large class imbalance, it was considered as a first step so no weighing to offset this class imbalance was performed.

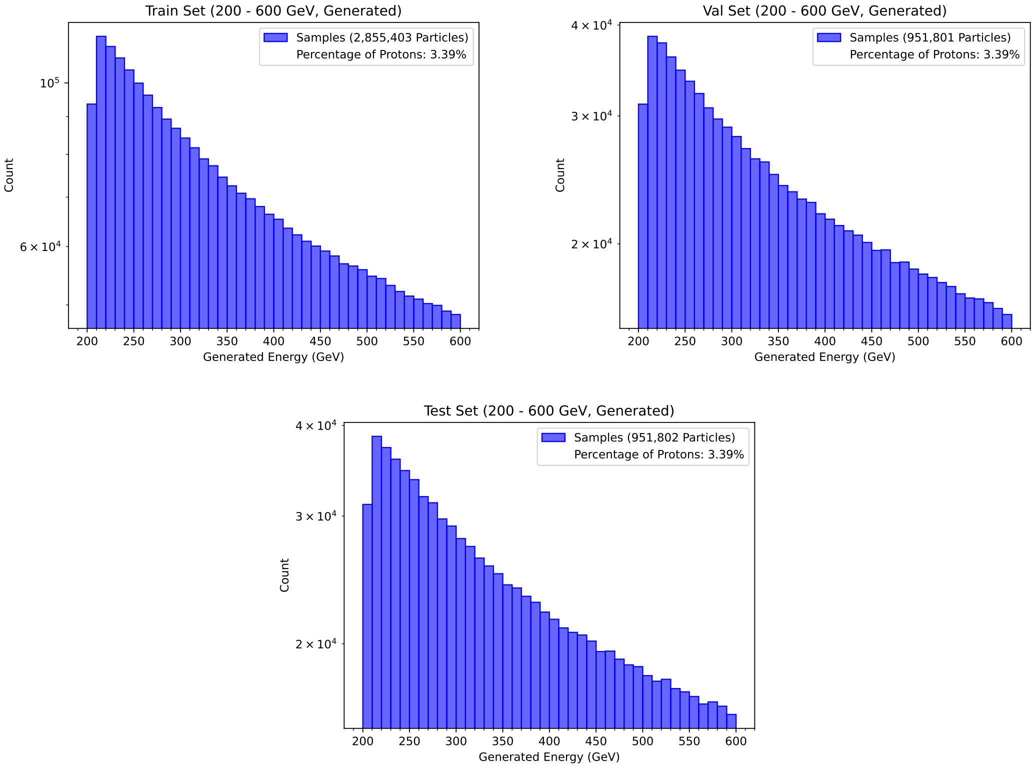


Figure 5.1: Generated energy histograms for the train/val/test sets created from the first dataset, consisting of events with a generated energy between 200–600 GeV. The class imbalance (percentage of protons) and overall energy distribution is similar between the three sets.

After the standard training, validation, and testing procedures, the average accuracy, defined as the ratio between correctly classified particles and the total number of particles, for all the models is shown in Table 5.1. We can see the accuracy scores are very similar, with the exception of the LogReg model. The HistBDT and DL models seem to already outperform the AMS models. A similar result can be seen in the ROC curves, shown in Figure 5.2, where the HistBDT, SimpleCNN, and ResNet18 models perform very similarly to the AMS models. The LogReg model learned to only score 0s and 1s, and so its ROC curve consists of a single point. The results for both these performance metrics do not reveal much information as most of the models have similar results.

Table 5.1: Accuracy Score for the current AMS models, simple ML models, and DL models.

Models	AMS BDT	AMS LHD	LogReg	SVM	HistBDT	SimpleMLP	SimpleCNN	ResNet18
Accuracy	0.973	0.967	0.843	0.911	0.998	0.988	0.999	0.998

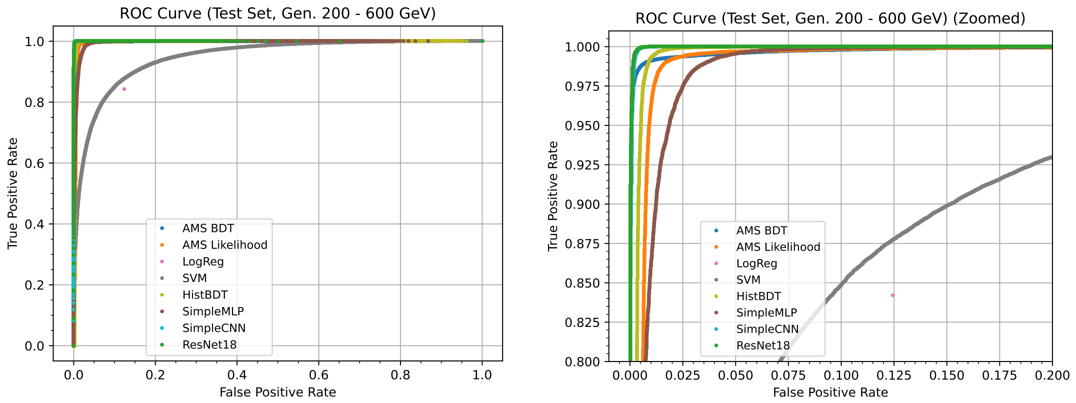


Figure 5.2: ROC curves for the dataset consisting of events with a generated energy between 200–600 GeV. ROC curves (left) for all the current AMS models, simple ML models, and DL models, with a zoomed (right) view around the bending point. The LogReg learned to only give scores of 0 or 1, and thus appears as a single dot.

This motivated us to then use the proton rejection performance metric described in Section 4.4, shown in Figure 5.3. The x-axis consists of electron efficiencies and the y-axis consists of proton rejections achieved using the model score calculated from the corresponding electron efficiencies. Here, we see more distinguishable performances between all the models, with the ResNet18 model outperforming the AMS models and the Simple ML models performing poorly in comparison. Thus, we establish a clear performance benefit of using DL models over our simple ML models.

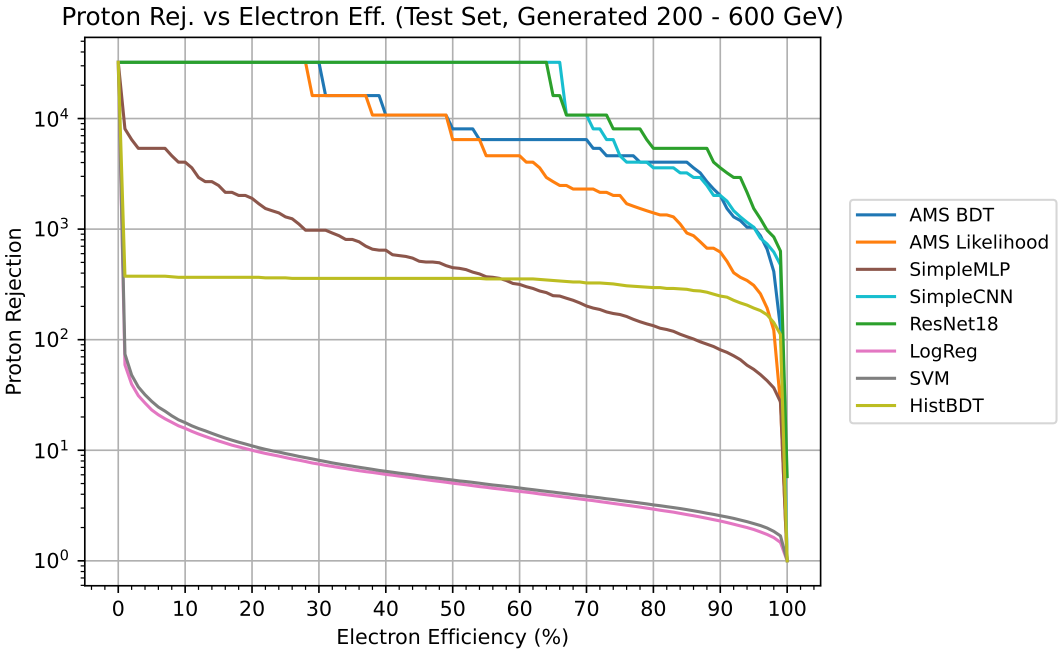


Figure 5.3: Proton rejection versus electron efficiency for the current AMS models, simple ML models and DL models for the dataset consisting of events with a generated energy between 200–600 GeV. The LogReg and SVM models perform poorly while the HistBDT has an average performance. The SimpleCNN and ResNet18 models perform comparably to the currently used AMS models.

5.2 Comparison on the First Reconstructed Energy Dataset

We next compare our DL models against the current AMS models on the second dataset, containing MC events with a reconstructed energy between 200–2000 GeV and the second dataset overall. Figure 5.4 shows the reconstructed energy histograms of the train, val, and two different energy test sets to assure the energy distribution and class imbalance for the above and below 1 TeV sets remained similar. Since the events were generated using a power law similar to that observed in space, we once again get right-skewed histograms.

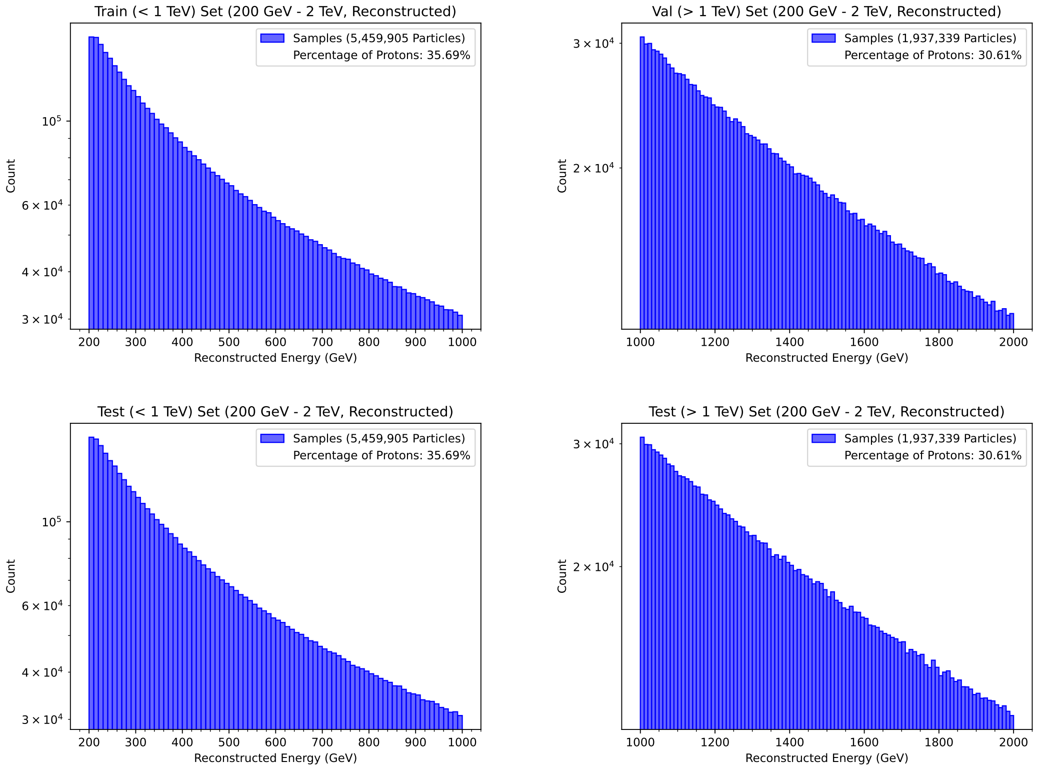


Figure 5.4: Reconstructed energy histograms for the train, val, and two different energy test sets created from the second dataset containing events with a reconstructed energy between 0.2–2 TeV.

The training, validation, and testing procedures described in Section 3.4 were performed. After seeing the proton rejection results on the above 1 TeV test set, the CvT model was developed and compared as well. Figure 5.5 shows the loss and accuracy curves for both the training and validation sets for all 4 models. These show the loss scores and average class accuracy, respectively, after each epoch of training. We can see the loss values generally decreasing and the accuracy generally increasing with the epochs. A more zoomed in view with the y-axis set to the same scale is shown in Figure 5.6, where the SimpleMLP model can no longer be seen. All the models show learning occurring on the train set, with only the CvT and SimpleMLP showing learning on the validation set, with the latter having the largest validation loss of all the models. The SimpleCNN and ResNet18 model have increasing validation loss after a certain point, indicating overfitting on the train set.

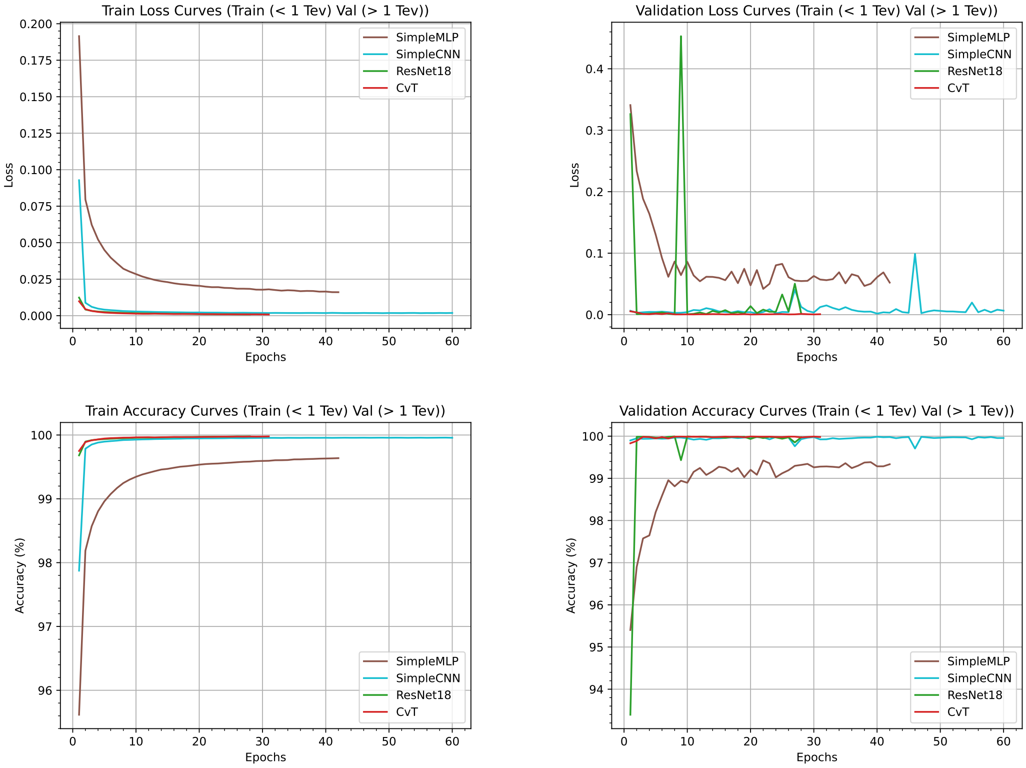


Figure 5.5: Loss (top) and accuracy (bottom) plots for the SimpleMLP, SimpleCNN, ResNet18, and CvT models trained on particles with a reconstructed energy between 200–1000 GeV and validated on particles with a reconstructed energy between 1–2 TeV. The CvT models performs the best on both the train and validation sets. Note: the different training epochs for each model are a result of early stopping being employed (see Section 3.4).

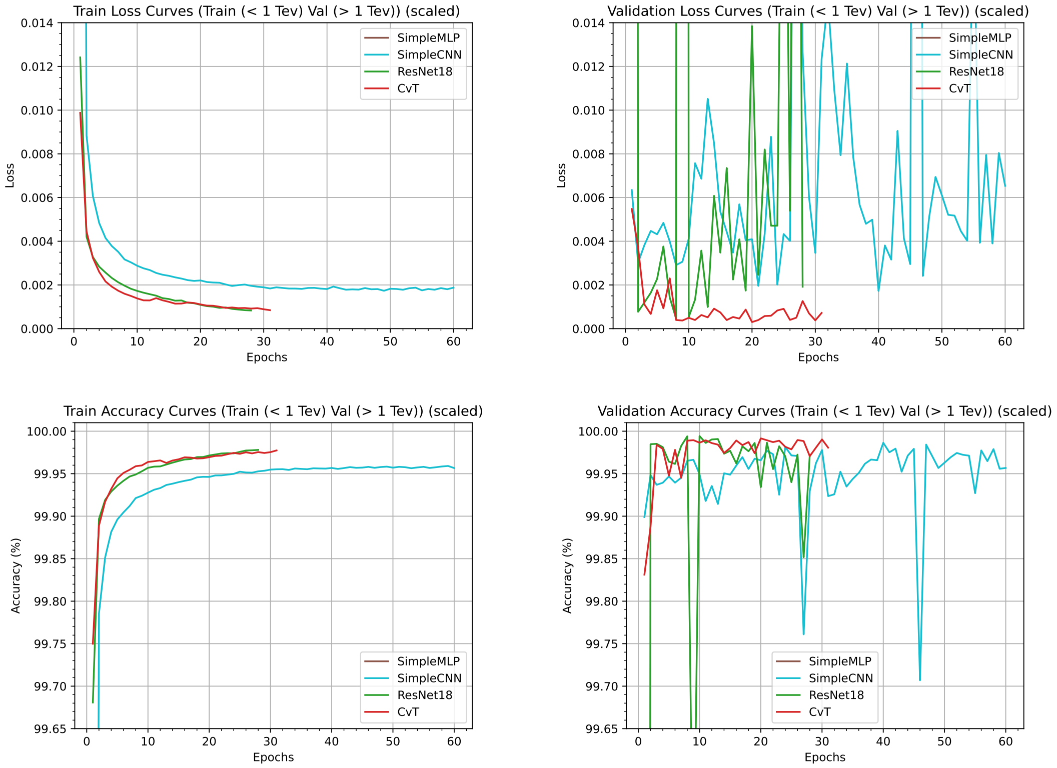
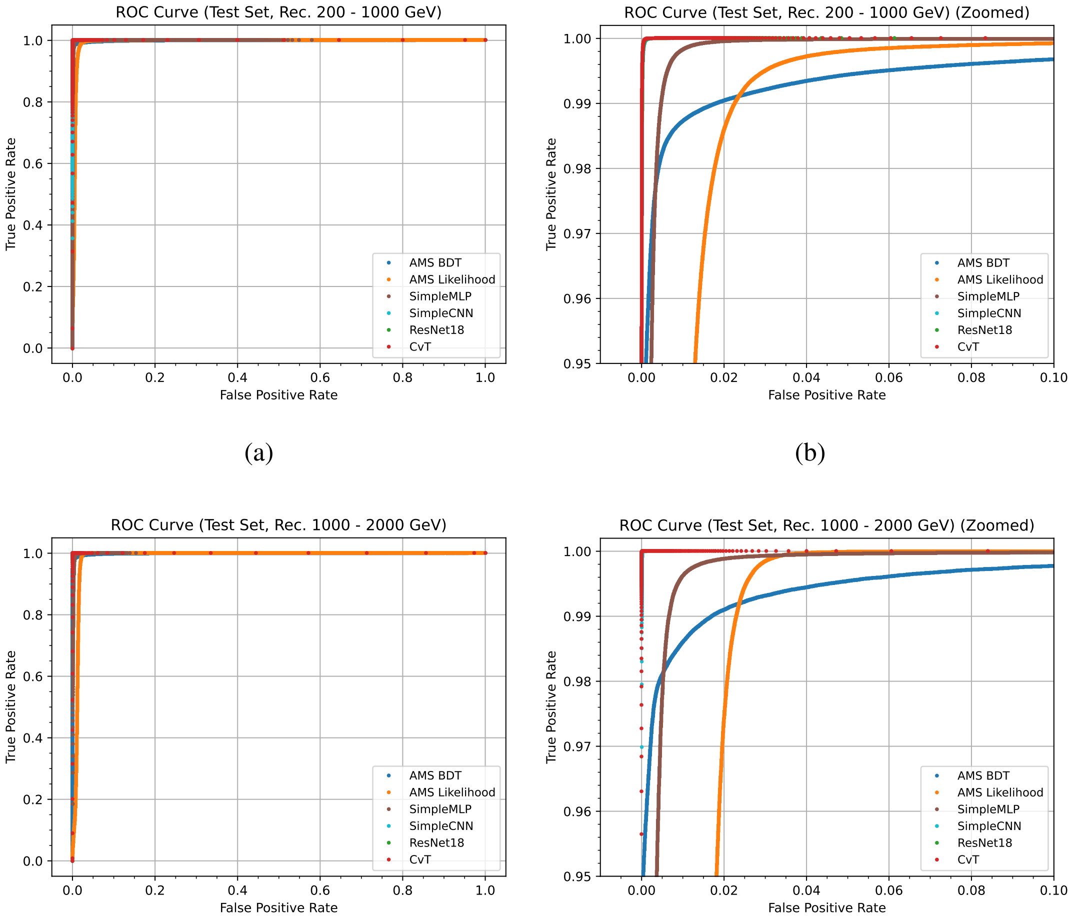


Figure 5.6: Zoomed in and y-axis scaled loss and accuracy plots for the SimpleMLP, SimpleCNN, ResNet18, and CvT models trained on particles with a reconstructed energy between 200–1000 GeV and validated on particles with a reconstructed energy between 1–2 TeV.

The ROC curves for both the below and above 1 TeV test sets are shown in Figure 5.7. The results once again show the models performing too similarly for any meaningful comparison, motivating the need for the proton rejection-based metrics. Figure 5.8 shows the proton rejection versus electron efficiency for all 4 trained models and the currently used AMS models. The ResNet18's underperformance on the above 1 TeV test set compared to the AMS models prompted the building of the CvT model, which then outperformed both the DL and AMS models on both the below and above 1 TeV test sets.

To better understand for which energy bins the models perform poorly, the proton rejection versus energy for the entire energy range was plotted at 60% and 90% electron efficiencies, shown in Figure 5.9. All models see a drop in performance at high energies except for the CvT model. There is also an unexpected drop below 350

GeV at 90% electron efficiency for the CvT which might be an artifact of the not-well understood MC for protons.



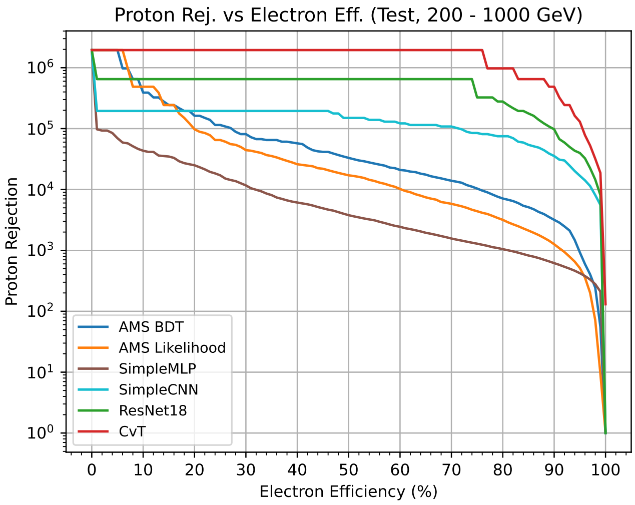
(a)

(b)

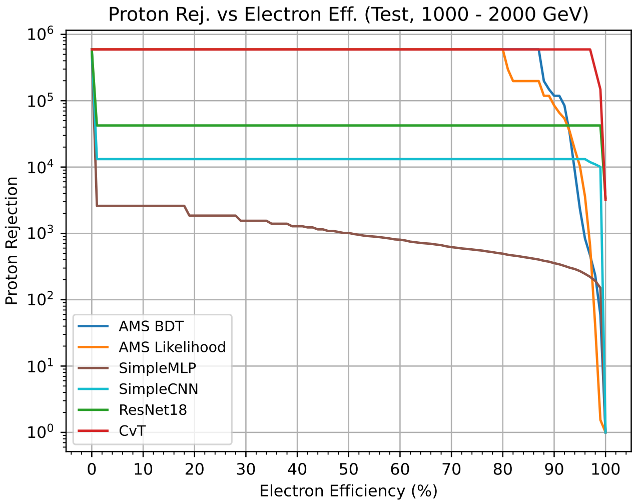
(c)

(d)

Figure 5.7: ROC curves for datasets consisting of events with a reconstructed energy between 200–2000 GeV. ROC curves, (a) and (c), for all the current AMS models and DL models for the below and above 1 TeV test sets, respectively, with their corresponding zoomed views, (b) and (d).

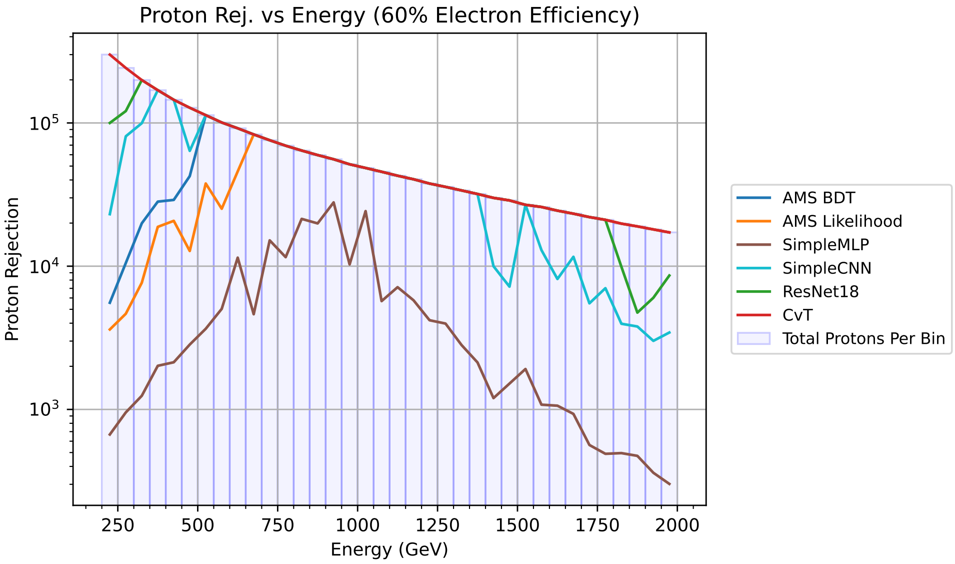


(a)

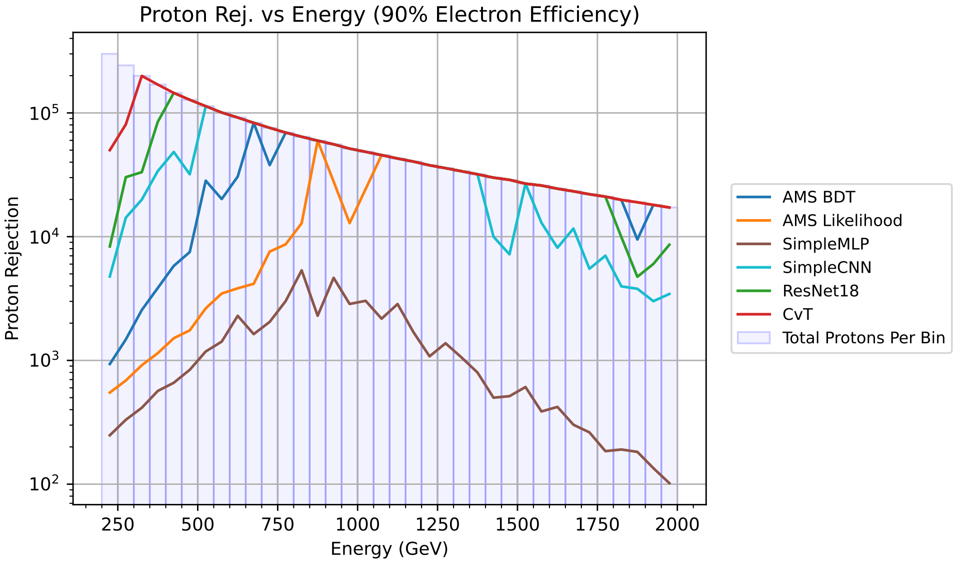


(b)

Figure 5.8: Proton rejection versus electron efficiency for the second dataset, consisting of particles with a reconstructed energy between 200 GeV–2 TeV. (a) At 90% electron efficiency for particles below 1 TeV, the CvT outperforms the ResNet18, SimpleCNN, AMS BDT, AMS Likelihood, and SimpleMLP models by factors of 5, 14, 153, 386 and 789, respectively. (b) At 90% electron efficiency for particles above 1 TeV, the CvT outperforms the AMS BDT, AMS Likelihood, ResNet18, SimpleCNN, and SimpleMLP models by factors of 5, 7, 14, 45, and 1666, respectively.



(a)

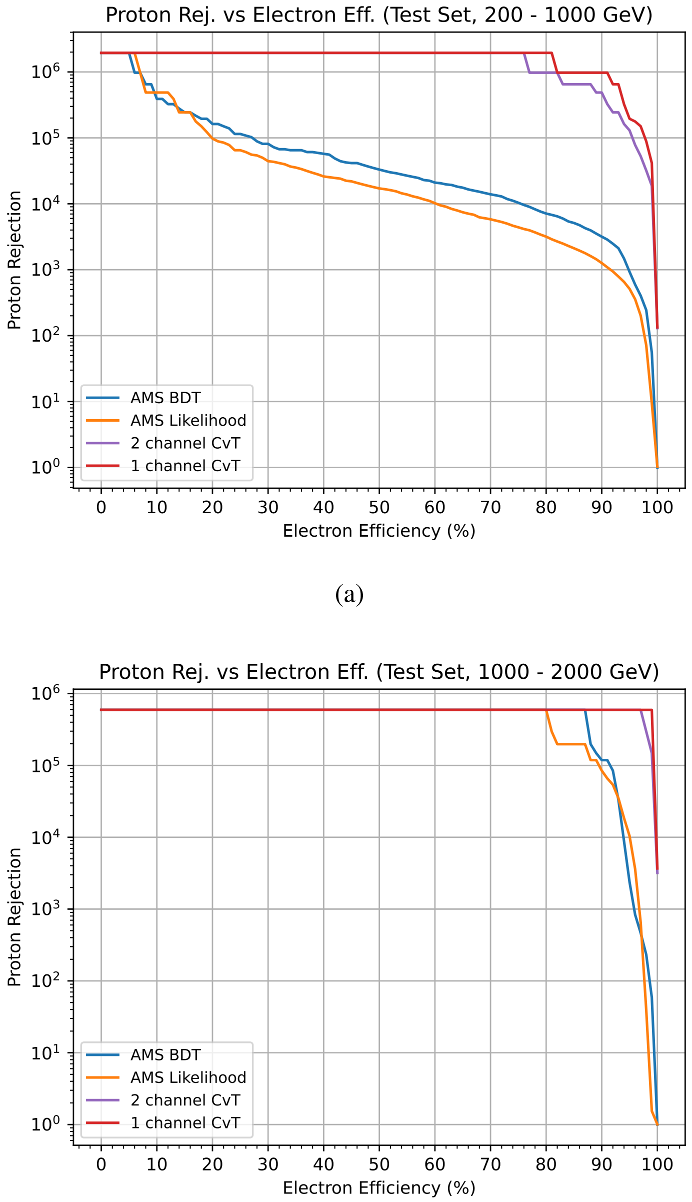


(b)

Figure 5.9: Proton rejection versus energy for the second dataset consisting of particles with a reconstructed energy between 200 GeV–2 TeV. All models exhibit drops in performance at various energy bins. (a) At 60% electron efficiency the CvT completely rejects all the available protons. (b) At 90% electron efficiency, the CvT performance drops for particles below 350 GeV, but is successful at rejecting all available the protons above 350 GeV.

5.3 Two Channels vs. One Channel

A small experiment was conducted to evaluate whether it would be better to train and test on the 2 channel dataset of resolution $2 \times 18 \times 72$ or to switch to a single channel with resolution of $1 \times 18 \times 72$. Initially, the assumption we made was that spatial aspects of the shower might be better captured by convolutions if the X and Y channels were split into two. To test this, we reshaped the second dataset consisting of particles with a reconstructed energy between 200 GeV–2 TeV to have only a single channel, where every 2 rows represents either the x- or y-direction similar to the 2D view in Figure 4.3. Since the CvT model outperformed all the other models, it was taken as the base model to compare between the 2 channel and 1 channel data representations, labeled 2 channel CvT and 1 channel CvT, respectively. Figure 5.10 shows the results of the comparison. The 1 channel CvT performs slightly better than the 2 channel variant below 1 TeV, but performs about the same above 1 TeV. While the slight performance increase could be attributed to stochastic factors during training, the simplicity of using 1 channel and the slightly reduced number of input pixels and training parameters convinced us to use the 1 channel representation for the remainder of the experiments.



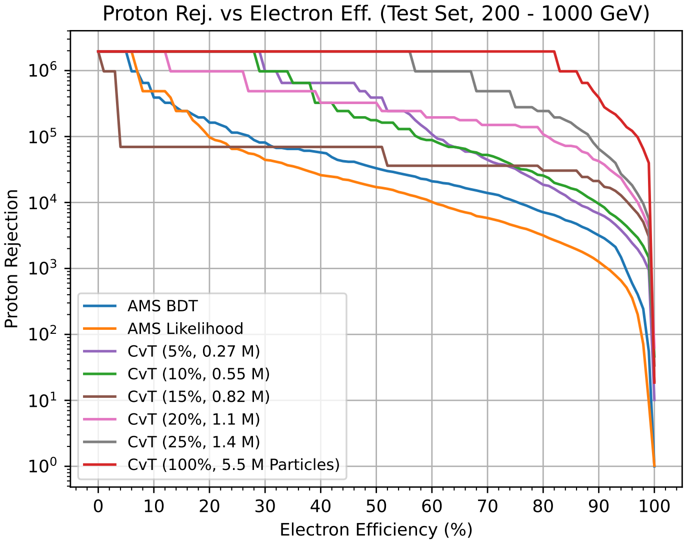
(a)

(b)

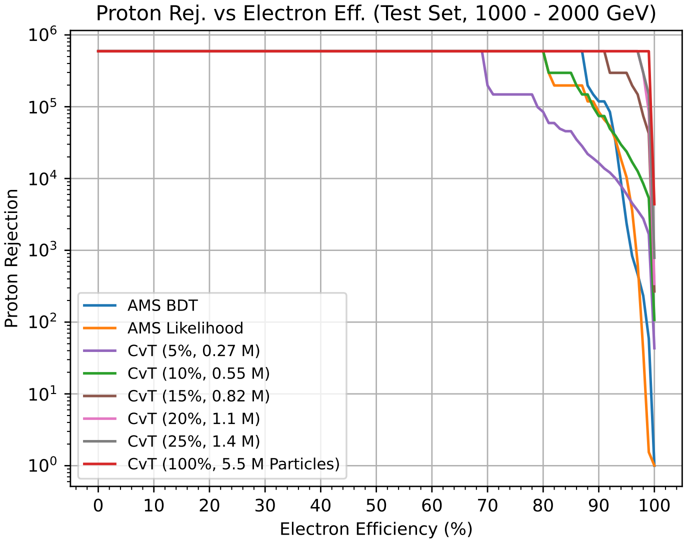
Figure 5.10: Proton rejection versus electron efficiency for the 1-channel and 2-channel CvT variants on the second dataset consisting of particles with a reconstructed energy between 0.2–2 TeV. (a) At 90% electron efficiency below 1 TeV, the 1 channel CvT outperforms the 2 channel variant by a factor of 2. (b) At 90% electron efficiency above 1 TeV, both 1 channel and 2 channel variants perform equally.

5.4 CvT Performance on Smaller Training Sets

Motivated with the knowledge that the amount of ISS data available for training will be significantly less compared to the training data available with MC simulated events, we performed an experiment to study how much data is needed to train a CvT. We trained and compared the CvT model on different amounts of training data, but kept the exact same validation and test sets. The results of the models on both test sets is shown in Figure 5.11, showing the steady increase in performance as more training data is used.



(a)



(b)

Figure 5.11: Proton rejection versus electron efficiency for the CvT model trained on various amounts of training data, show in the parenthesis. The performance on both the (a) above and (b) below 1 TeV test sets show a steady increase in performance as more training data is used.

5.5 Implementing the Physics-based Feature Engineering

The results of the previous experiment on how much the CvT performance depends on the amount of training data motivated the need to improve learning efficiency. Since the physics of electromagnetic showers are generally well understood (see Section 2.2.5.1), a feature engineering method using physics principles was devised (see Section 4.3). The CvT model that use this feature engineering is labeled Phys+CvT. Additionally, the Phys+CvT (small kernels) variant was created with different kernels and stride lengths to better fit the more narrow $1 \times 18 \times 21$ dataset (see Section 3.3.5).

This method required additional variables from the AMS ROOT files, in particular, the X-, Y-, and Z-coordinate of the incoming particle's incidence on the ECAL as well as the incident zenith and azimuth angles, all tracked by the Silicon Tracker. While these values can be calculated from the ECAL itself, the Tracker was chosen in order to provide an independent measurement to give additional information to the CvT. Thus, to get these variables, the second dataset consisting of MC events with a reconstructed energy between 200 GeV–2 TeV, and third dataset overall, was extracted. The slight increase in the number of particles is noted in Table 4.1.

As a first step, the Phys+CvT and Phys+CvT (small kernels) variants were first trained on the full training set, consisting of 5.7 million particles. Figure 5.12 shows the reconstructed energy histograms of the train, val, and two test sets to again assure the energy distribution and class imbalance for the above and below 1 TeV sets remained similar. Training, validation, and testing on the two test sets was redone for all relevant DL models.

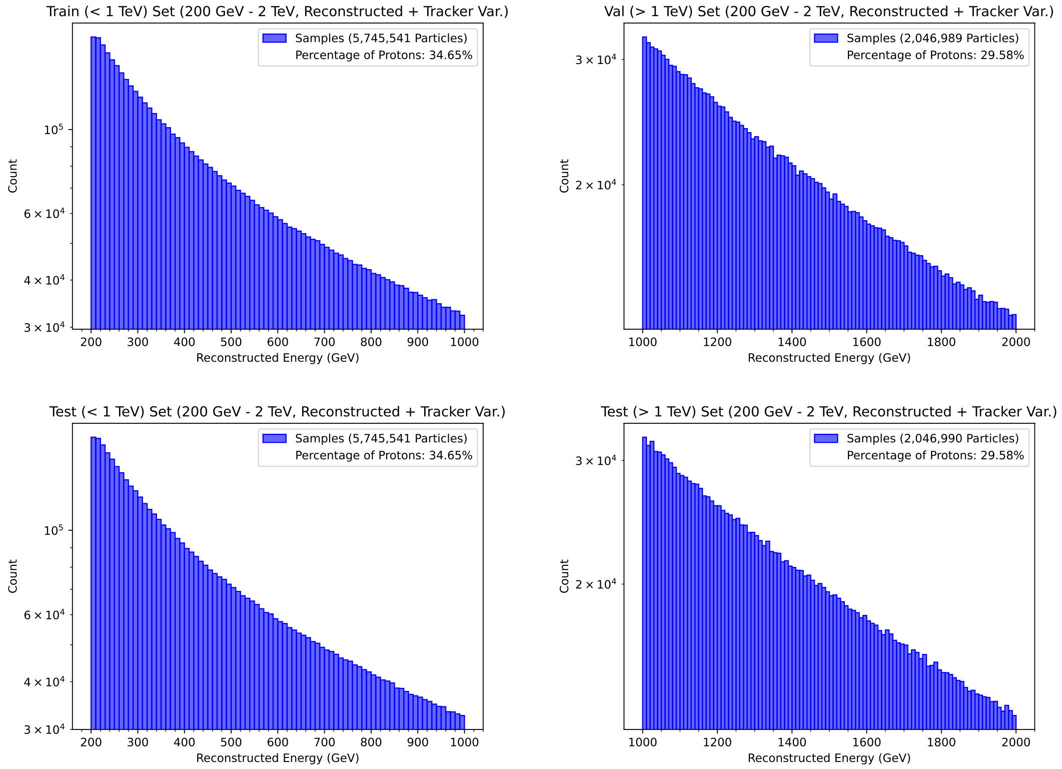


Figure 5.12: Reconstructed energy histograms for the train, val, and two test sets created from the third dataset, containing events with a reconstructed energy between 0.2–2 TeV and additional variables from the Tracker.

In the interest of focus, the SimpleMLP was dropped from evaluations due to its general poor performance. In its place, the ResNet10 model was trained and evaluated to determine whether the reason for the CvT's success was because of the architecture performing better, or because of the ResNet18 having double the number of trainable parameters (see Table 3.2) which was resulting in overfitting to the training set and/or the double descent phenomenon [119]. In addition, the ROC performance metric was also dropped as it was insufficient at providing a meaningful difference between models designed for proton rejection.

Figure 5.13 shows the loss and accuracy curves for all the trained models with the y-axis set to the same scale. Discussion on the ResNets and Phys+CvTs is split into two figures, while it is clear that the SimpleCNN model performs the poorest.

Figure 5.14 focuses on the ResNet10, ResNet18, and CvT models, with the latter serving as a benchmark. All three models have a similar steepness for the train set loss and accuracy curves, while both the ResNet models have large fluctuations on the validation set loss and accuracy curves. The ResNet10 model does not show any clear signs of better performance over the ResNet18.

Figure 5.15 focuses on the Phys+CvT, Phys+CvT (small kernels), and CvT models, with the latter again serving as a benchmark. We can see Phys+CvT (small kernels) has shallower loss and accuracy plots on the train set, indicating it did not train as effectively. In addition, it has large fluctuations for the validation curves compared to the other models. This can be an indication of a poor choice of kernel size and stride length. The Phys+CvT on the other hand appears to be doing similarly to the CvT with a few larger fluctuations on the validation set.

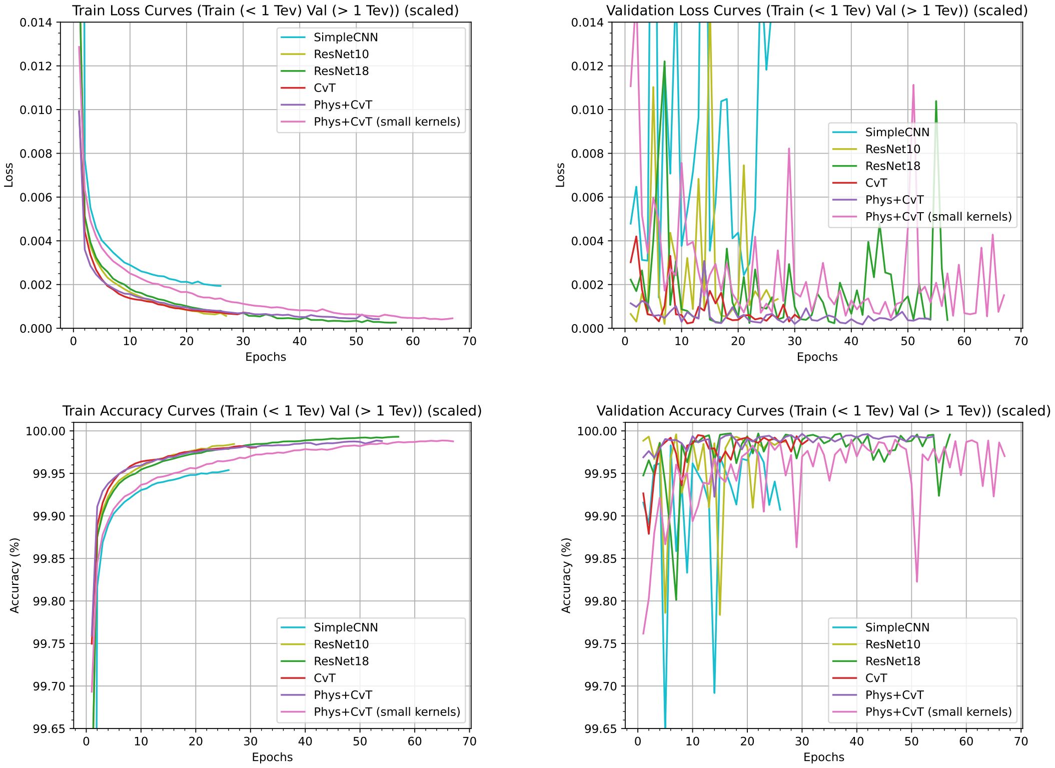


Figure 5.13: Loss (top) and accuracy (bottom) plots for the SimpleCNN, ResNet, and CvT models trained and validated on the third dataset containing events with a reconstructed energy between 0.2–1 TeV and 1–2 TeV, respectively. The y-axis are set to the same scale. All models appear to be learning on the train set, but there are a cluster of plots for the validation sets. See Figures 5.14 and 5.15 for clarity. Note: the different training epochs for each model are a result of early stopping being employed (see Section 3.4).

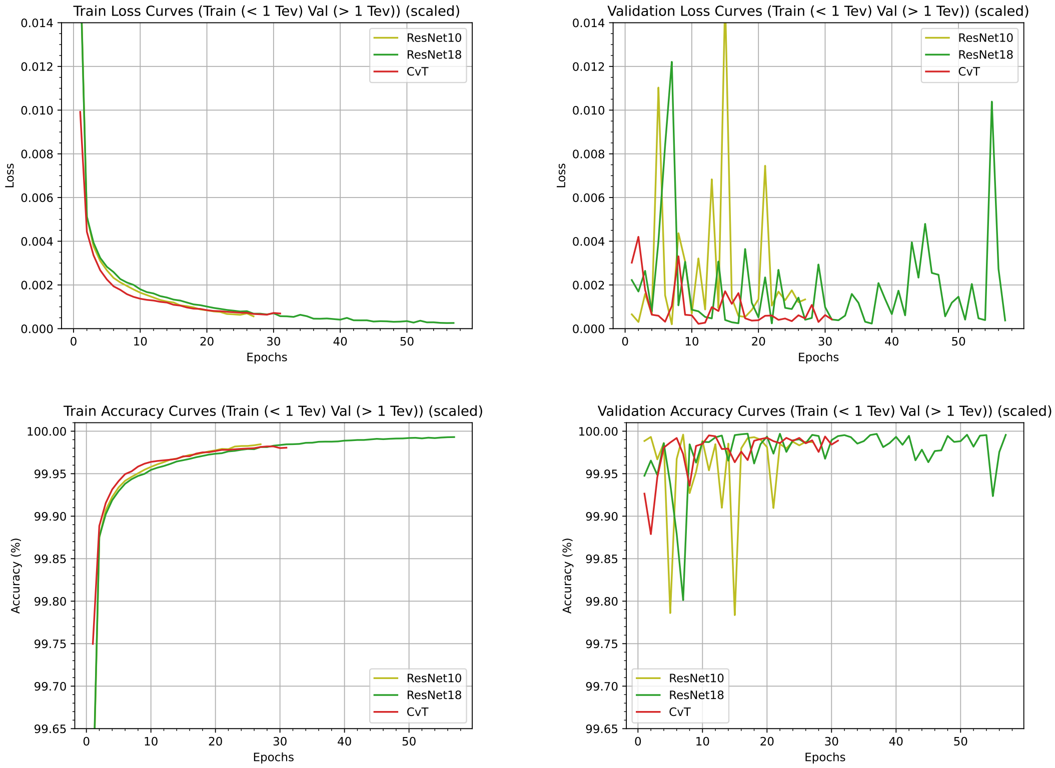


Figure 5.14: Loss (top) and accuracy (bottom) plots from Figure 5.13 but focusing on the ResNet10, ResNet18, and CvT models. All models appear to have a similar learning curve on the train set with lots of fluctuations on the validation set for both the ResNet models compared to the CvT model.

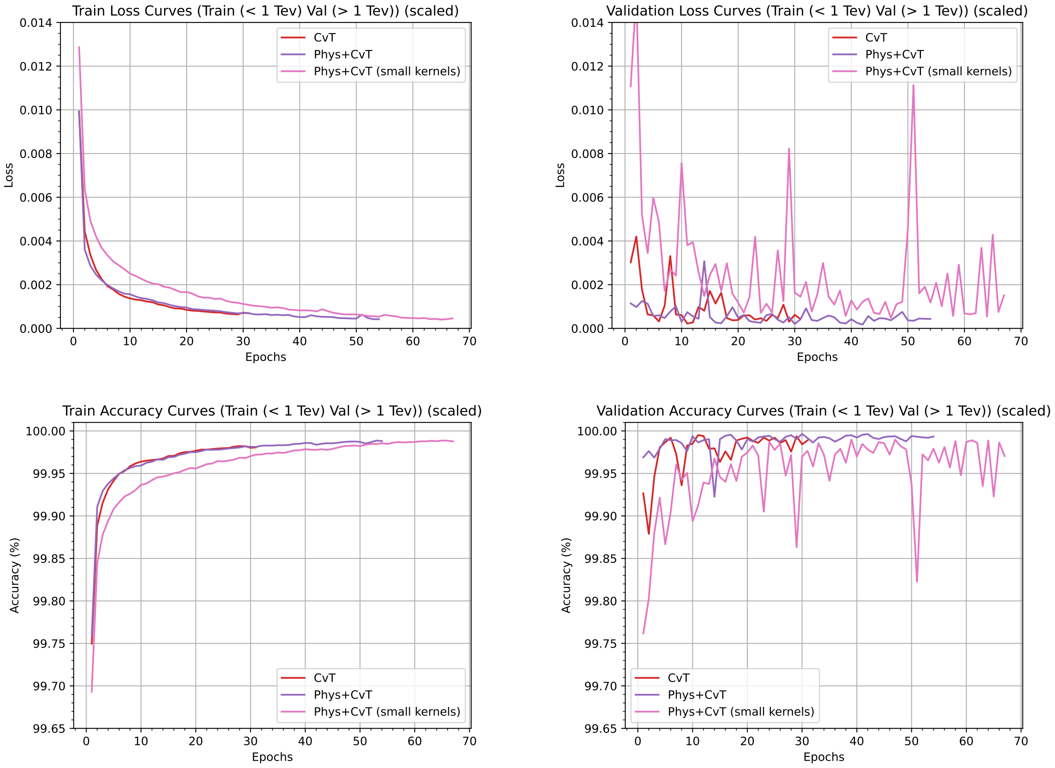
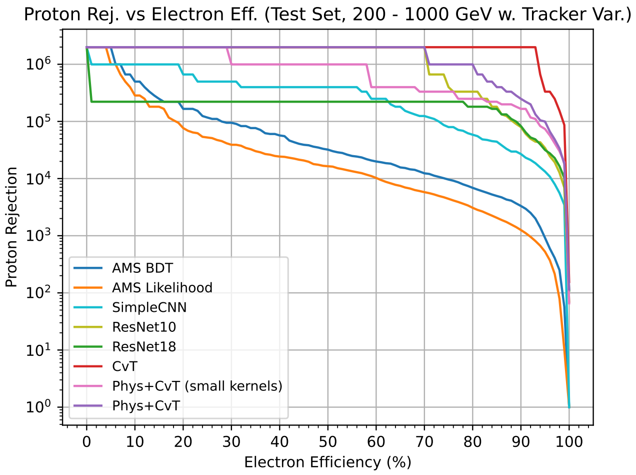


Figure 5.15: Loss (top) and accuracy (bottom) plots from Figure 5.13 but focusing on the Phys+CvT, Phys+CvT (small kernels), and CvT models. Phys+CvT (small kernels) has the shallowest loss curve on the train set and the most fluctuations on the validation set.

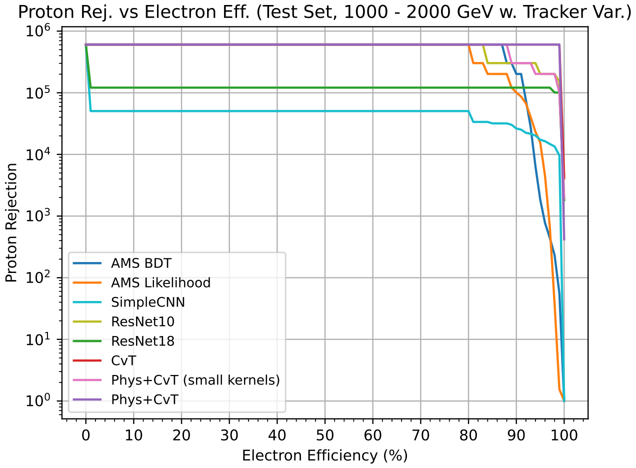
Figure 5.16 shows the proton rejection versus electron efficiency plot. The CvT is still the best performing model, with the Phys+CvT coming in second for below 1 TeV and practically equal for above 1 TeV. Additionally, below 1 TeV, ResNet10 outperforms the ResNet 18 until 86% electron efficiency, after which their performance becomes equal. Above 1 TeV, the ResNet10 outperforms at every electron efficiency. The Phys+CvT (small kernels) is the weakest performing model from the CvTs, but still outperforms the current AMS models, an indicator of the strong performance of a CvT even with unoptimized hyperparameters.

Figure 5.17 shows the CvT models rejecting all available protons at 60% electron efficiency, and only losing performance below 350 GeV at 90% electron efficiency. At 60% electron efficiency, the ResNet18 starts to dip about 50 GeV before the ResNet10. At 90% electron efficiency, while the ResNet10 starts to dip before ResNet18, the latter has a steeper decline, resulting in the performance seen in Figure 5.16.

The results show that the ResNet10 model boasts slightly better performance over the ResNet18, but does not match the performance of the CvT or Phys+CvT model. Between the latter two, the CvT outperforms the Phys+CvT below 1 TeV, but has equal performance for above 1 TeV.

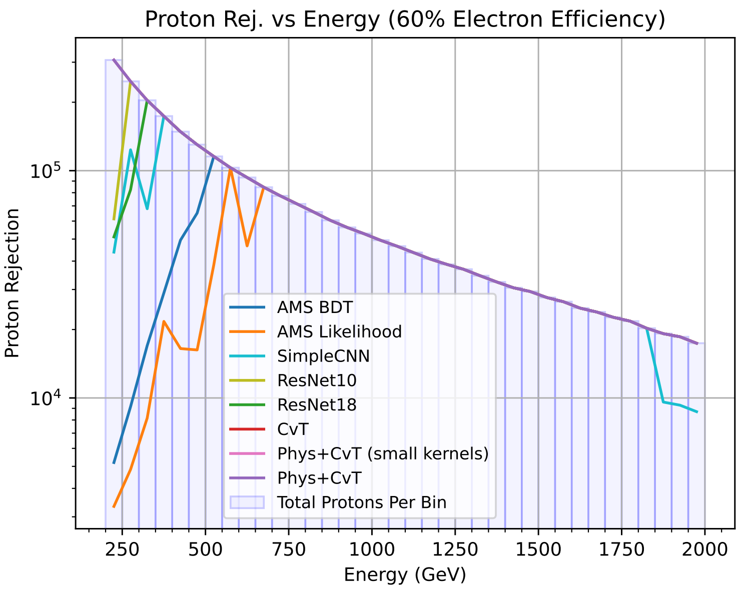


(a)

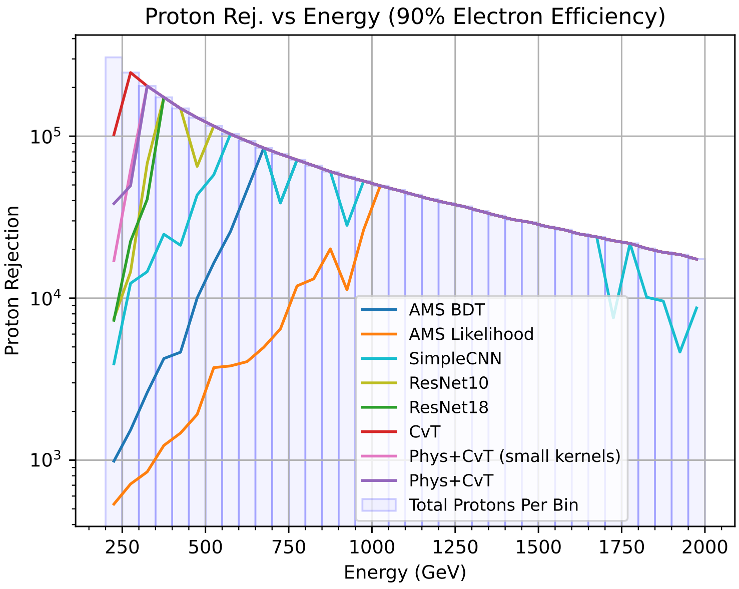


(b)

Figure 5.16: Proton rejection versus electron efficiency for the third dataset consisting of particles with a reconstructed energy between 0.2–2 TeV with additional Tracker variables. (a) At 90% electron efficiency, below 1 TeV, the CvT outperforms the Phys+CvT, Phys+CvT (small kernels), ResNet18, ResNet10, SimpleCNN, AMS BDT, and AMS Likelihood models by factors of 8, 12, 24, 25, 74, 600 and 1581, respectively. (b) At 90% electron efficiency, above 1 TeV, the CvT performs equally with the Phys+CvT, outperforms the ResNet10 and Phys+CvT (small kernels), which perform equally, by a factor of 2 and outperforms the AMS BDT, RestNet18, AMS Likelihood, and SimpleCNN models by factors of 3, 5, 6 and 23, respectively.



(a)



(b)

Figure 5.17: Proton rejection versus energy for the third dataset consisting of particles with a reconstructed energy between 0.2–2 TeV with additional Tracker variables. (a) At 60% electron efficiency all the CvT models completely reject all the available protons. (b) At 90% electron efficiency, the CvT performances drops for particles below 350 GeV, but are successful at rejecting all the protons above 350 GeV.

5.6 Phys+CvT Performance on a Smaller Training Set

To assess any learning efficiency achieved with the physics-based feature engineering, a smaller training set was created using only 5% of data from the previously used training set, resulting in a training set with 0.29 million particles. The validation and two test sets remained exactly the same. Figure 5.18 shows the reconstructed energy histogram to assure that the smaller training set's energy distribution and class imbalance looked reasonably similar to the corresponding test set's energy distribution.

The CvT, Phys+CvT, and Phys+CvT (small kernels) were retrained on this reduced training set and their performances were compared to each other, the same models trained on the full training set, and the currently used AMS models. Figure 5.19 shows the loss and accuracy plots, where it can be seen that the Phys+CvT performs better on the validation set after training on 5% of the training data when compared to the similarly trained CvT model.

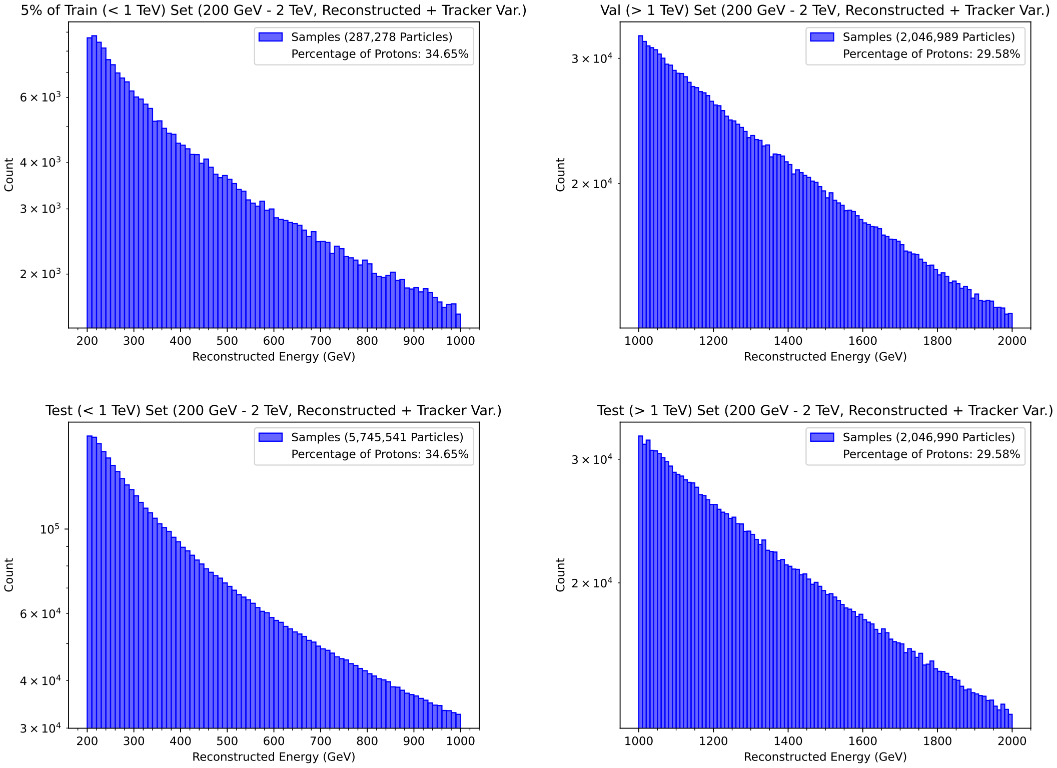


Figure 5.18: Reconstructed energy histograms from the third dataset consisting of particles with a reconstructed energy between 0.2–2 TeV with additional Tracker variables. The smaller train set uses only 5% of the original train set, resulting in only 0.29 million particles. The val and two test sets are exactly the same as before. The train set's histogram is reasonably similar to the test set's histogram for above 1 TeV.

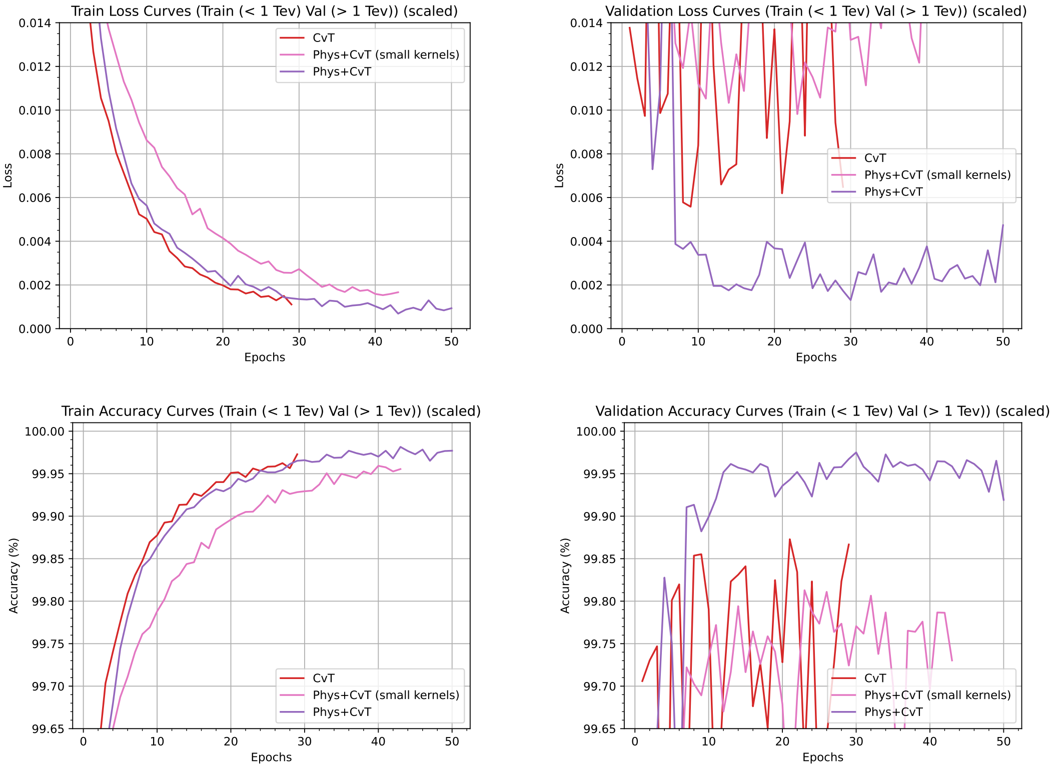
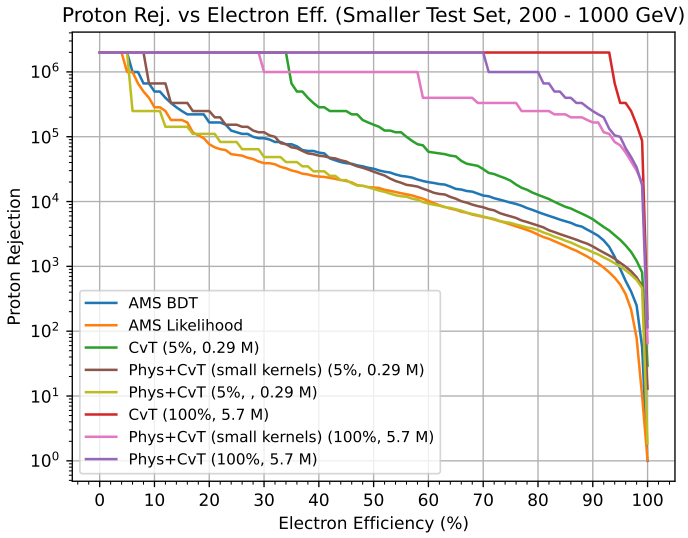


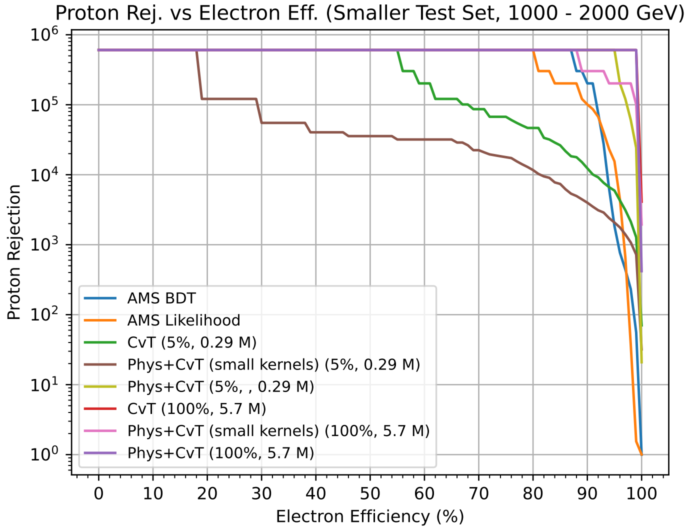
Figure 5.19: Loss (top) and accuracy (bottom) plots for the CvT, Phys+CvT, and Phys+CvT (small kernels) models trained on a smaller amount of data consisting of events with a reconstructed energy between 0.2–2 TeV with additional Tracker variables. The CvT has the steepest loss curve on the train set, but the Phys+CvT greatly outperforms on the validation set.

Figure 5.20 shows the proton rejection versus electron efficiency plots. For below 1 TeV, none of the models trained on 5% of the train set are comparable to the models trained on 100% of the train set. However, for above 1 TeV, the Phys+CvT trained on 5% of the train set comes remarkably close in performance of the CvT and Phys+CvT models trained on 100% of the train set, equaling them at 90% electron efficiency. Figure 5.21 shows the proton rejection versus energy for the entire energy range, where it can be seen that for 90% electron efficiency, the Phys+CvT (5%) performance starts to drop below 800 GeV. This could indicate that the model focused on shower image characteristics that solely helped it perform well on the above 1 TeV validation set, ignoring certain shower image properties possessed by particles below 800 GeV.

The results showed that the physics-based feature engineering improved efficiency in learning for events with a generated energy above 1 TeV, but did not improve efficiency for events below 1 TeV. However, this could be attributed to the inaccurate hadronic shower modeling for MC protons. As such, having shown promise on MC events, our next step is to evaluate on ISS data.

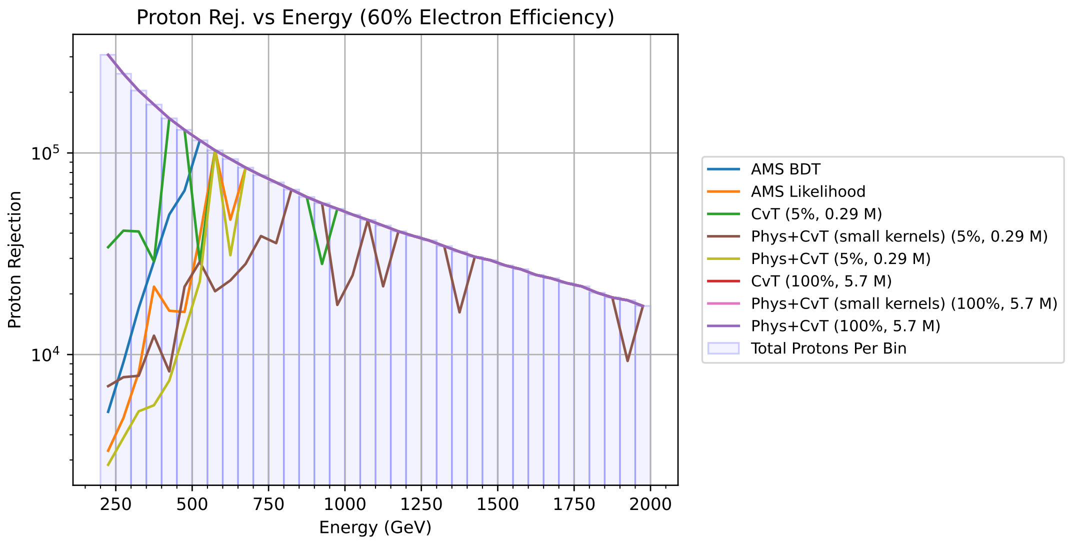


(a)

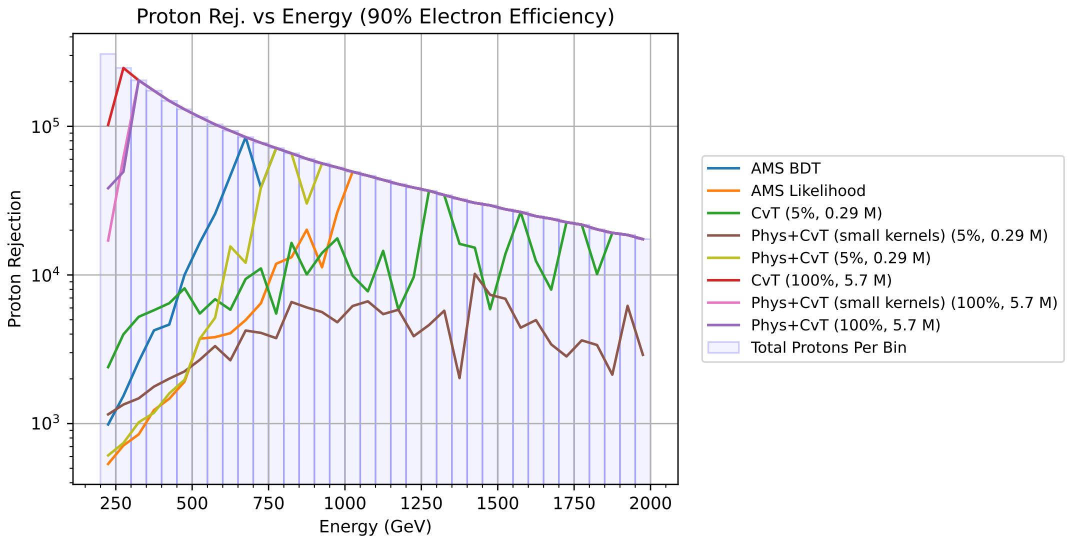


(b)

Figure 5.20: Proton rejection versus electron efficiency for the third dataset comparing models trained on 5% of the train set (0.29 million particles) and 100% of the train set (5.7 million particles). (a) Below 1 TeV, none of the models trained on 5% of the train set compare to the DL models trained on 100% of the data. (b) Above 1 TeV, the Phys+CvT trained on 5% of the train set outperforms the CvT trained on the same amount of data and is close in performance to the DL models trained on 100% of the train set.



(a)



(b)

Figure 5.21: Proton rejection versus energy for the third dataset comparing models trained on 5% of the train set (0.29 million particles) and 100% of the train set (5.7 million particles). (a) At 60% electron efficiency the Phys+CvT (5%) proton rejection power starts to dip for energies below 700 GeV, like many of the other models. (b) At 90% electron efficiency, the Phys+CvT (5%) has a steep drop in proton rejection below 800 GeV while the other models trained on 5% of the train set show consistent drops in almost every energy bin.

5.7 Performance on ISS Data

Using the TRD and Tracker, we selected a sample of electrons and protons with a reconstructed energy between 50–70 GeV from ISS data (see Section 4.2.1), resulting in 1.22 million particles (see Table 4.1). If we use all of this as training data, it would be comparable to training a CvT on about 22.2% of the training data from the experiment in Section 5.4.

Having extracted this fourth dataset, we performed two experiments. We first compared our MC-trained DL models against the current AMS models on the extracted ISS dataset, and then compared ISS data-trained DL models against the AMS models, which were also originally trained on ISS data, for a fairer comparison.

For the first experiment, we tested five DL models (SimpleCNN, ResNet10, ResNet18, CvT and Phys+CvT) that were previously trained and validated on the 3rd extracted MC dataset. The results are shown in Figure 5.22. The DL models performed poorly compared to the current AMS models that were trained on ISS data. This gives us more evidence for discrepancies between MC and ISS data (mentioned in Section 4.2).

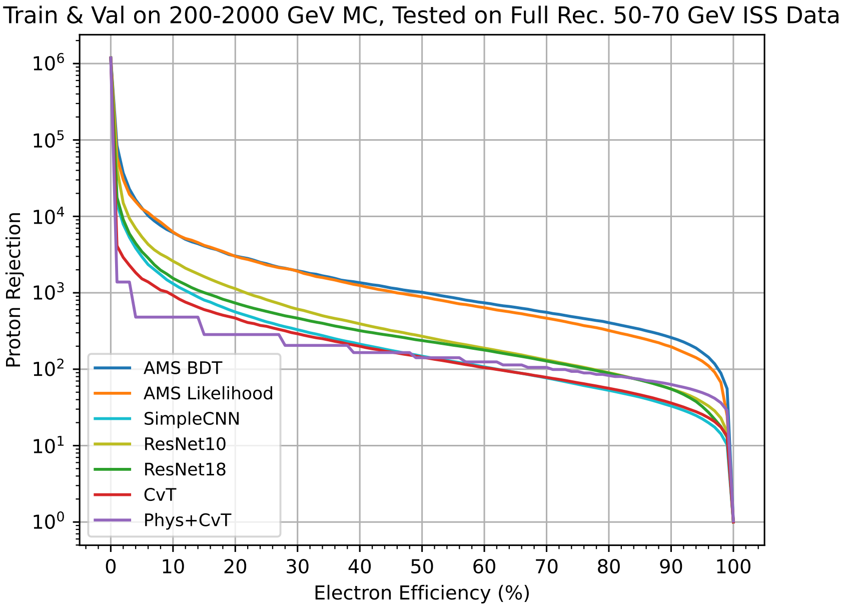


Figure 5.22: Proton rejection vs. electron efficiency on the fourth dataset consisting of ISS data with a reconstructed energy between 50–70 GeV. The DL models are trained on the third MC dataset. The AMS BDT outperforms the Phys+CvT, ResNet 10, ResNet18, SimpleCNN, and CvT models by factors of 4.11, 4.69, 4.72, 7.87 and 9.21, respectively.

Following this, we split the ISS data into the standard 60%/20%/20% train/val/test sets. The reconstructed energy histograms are shown in Figure 5.23, where we can see that the energy distribution and class imbalance are similar for all three sets. This ISS data train set consists of 0.73 million particles, which makes it comparable to training a CvT on about 13.3% of the training data from the experiment in Section 5.4.

We trained and validated the same five models from scratch, with the loss and accuracy curves shown in Figure 5.24. The SimpleCNN learns the best on the train set but overfits and performs the worst on the validation set. The Phys+CvT model, however, performs the best on the validation set.

The proton rejection vs. electron efficiency plot in Figure 5.25 shows the Phys+CvT model outperforming the ResNet10, ResNet18, CvT, AMS BDT, SimpleCNN and AMS Likelihood models by factors of 1.74, 1.79, 2.38, 2.53, 3.18 and 3.39, respectively. As such, the Phys+CvT not only outperforms the other DL models but also outperforms both of the current AMS models. Due to the small energy range, the proton rejection vs. energy performance metric was not performed.

The results show that the Phys+CvT method was successful in being more data efficient on the ISS data. Additionally, almost all of the DL models outperform the current AMS models, with only the SimpleCNN slightly underperforming against the AMS BDT.

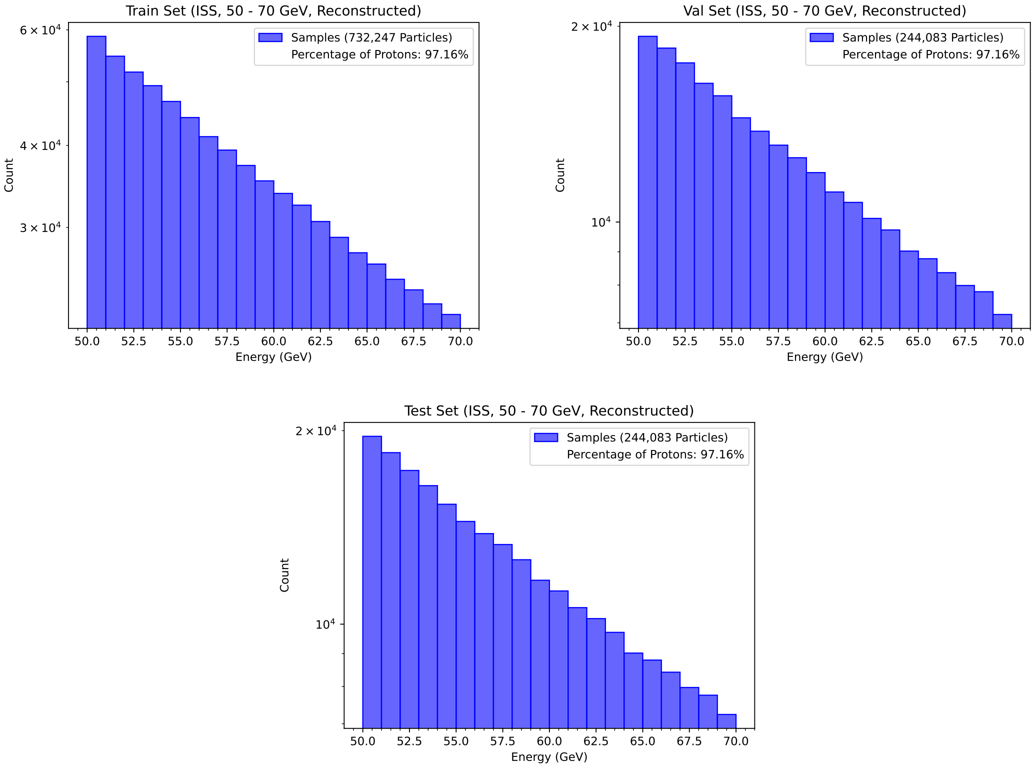


Figure 5.23: Reconstructed energy histograms for the train/val/test sets created from the fourth dataset consisting of ISS data with a reconstructed energy between 50–70 GeV. The class imbalance (percentage of protons) and energy distribution is the similar between the three sets.

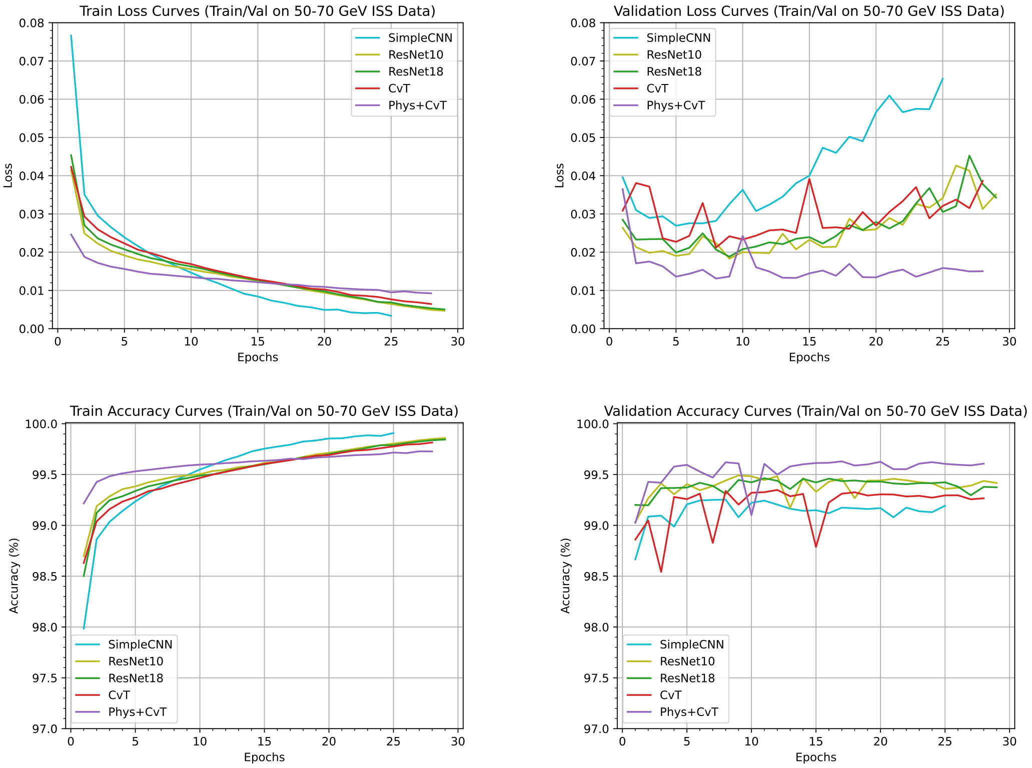


Figure 5.24: Loss (top) and accuracy (bottom) plots for the SimpleCNN, ResNet10, ResNet18, CvT, and Phys+CvT models trained on the fourth dataset consisting of ISS data with a reconstructed energy between 50–70 GeV. The SimpleCNN model performs the best on the train set, but overfits and performs the worst on the validation set. The Phys+CvT model has the best validation performance.

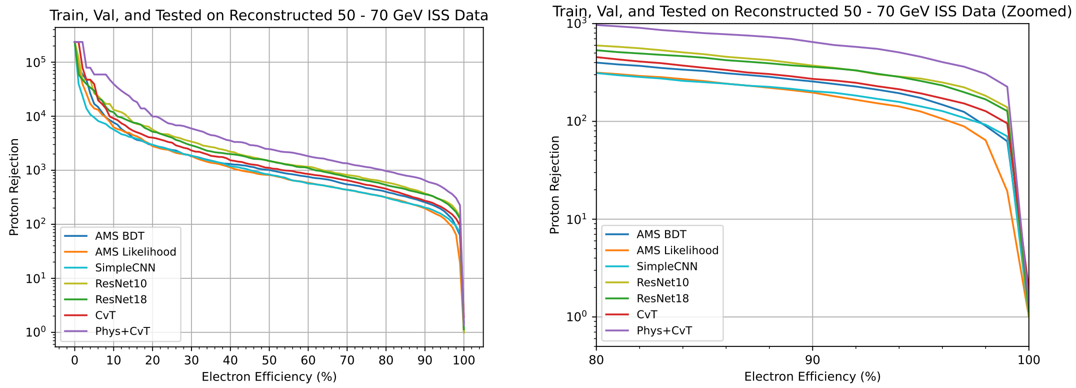
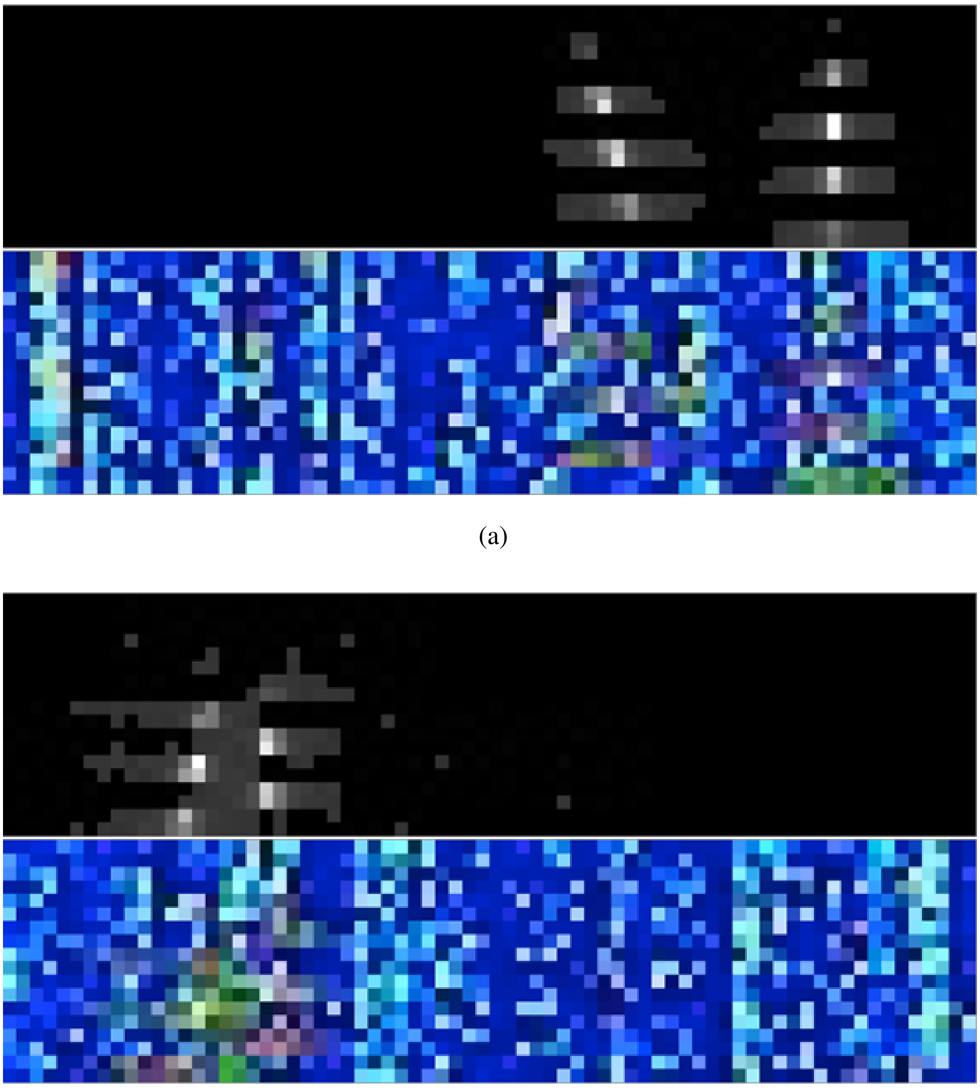


Figure 5.25: Proton rejection versus Electron Efficiency for the fourth dataset consisting of ISS data with a reconstructed energy between 50–70 GeV (left) and a zoomed in version for clarity (right). The DL models were trained on ISS data. The Phys+CvT model outperforms the ResNet10, ResNet18, CvT, AMS BDT, SimpleCNN and AMS Likelihood models by factors of 1.74, 1.79, 2.38, 2.53, 3.18 and 3.39, respectively.

5.8 Miscellaneous Experiment

5.8.1 Visualizing Class Activation Maps

To understand what the CvT model focuses on the most when determining a classification score for MC events, we attempted to visualize the class activation map for it using the Grad-CAM technique [120]. First, shower images in their 1-channel representation (see Section 5.3) were converted into actual images before having the Grad-CAM technique applied on the last convolutional layer of the CvT. The results, shown in Figure 5.26, were heavily pixelated, contained a lot of noise and did not provide any insight.



(a)

(b)

Figure 5.26: (a) Example of an electron shower image and a (b) proton shower image, with their corresponding Grad-CAM results (the blue images).

We then devised an alternative method where a 2×2 sliding window-style zeroing kernel was moved step-wise, with a stride length of 2, through a sample image, as depicted in Figure 5.27. For each location of the sliding window, the pixels within it were zeroed and the image was passed through the CvT model. The difference in model score from the original image and this partially zeroed out image was recorded at the location of the sliding window to create an activation heatmap representing all the pixels that contributed to the model score. Conversely, a heatmap showing the energy depositions that made no contribution to the model score (i.e. when these pixels were masked, the score did not change at all) was also generated.

Figures 5.28, 5.29, 5.30 and 5.31 give examples of two correctly classified and two misclassified events, respectively, at 90% electron accuracy. The layout is as follows: top-left shows the energy deposition map of the shower with the colorbar showing the energy in MeV, top-right shows the CvT activation heatmap with the colorbar showing the model score difference, bottom-left shows the CvT activation heatmap overlaid over the original shower image with the colorbar showing the previously mentioned model score difference, and bottom right shows the energy depositions that made no contributions (no difference) to the model score with the colorbar showing the energy in MeV. The CvT identifies the general area of the showers for protons and treats them as noise or background while it is able to pinpoint the pixels with the largest energy depositions (shower maximums) for electrons/positrons.

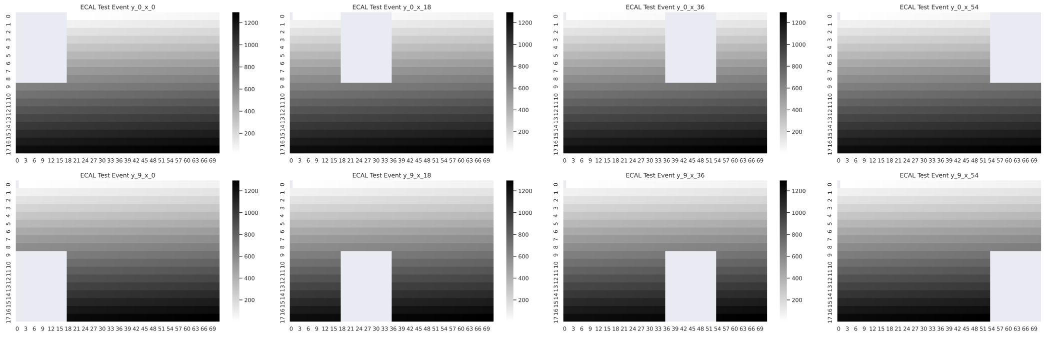


Figure 5.27: Depiction of how a sliding window that zeroes the pixels is moved stepwise through a sample image.

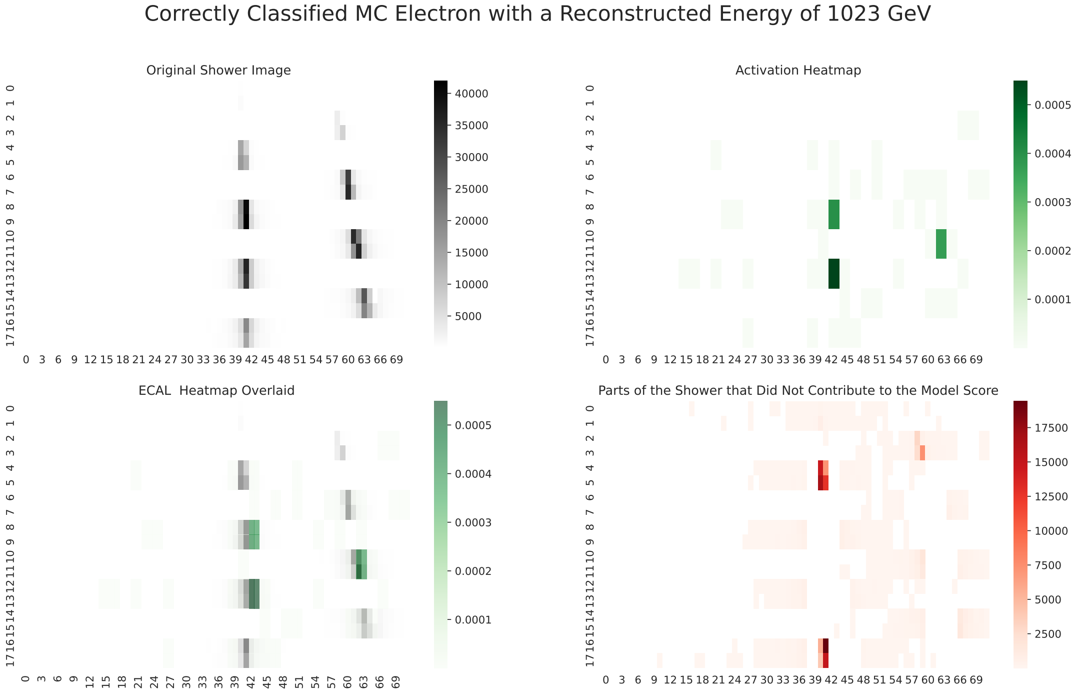


Figure 5.28: Example of a correctly classified MC electron shower with a reconstructed energy of 1023 GeV. The CvT focuses on the shower maximum between layers 8–13 and correctly identifies it as the main contributor to the shower classification.

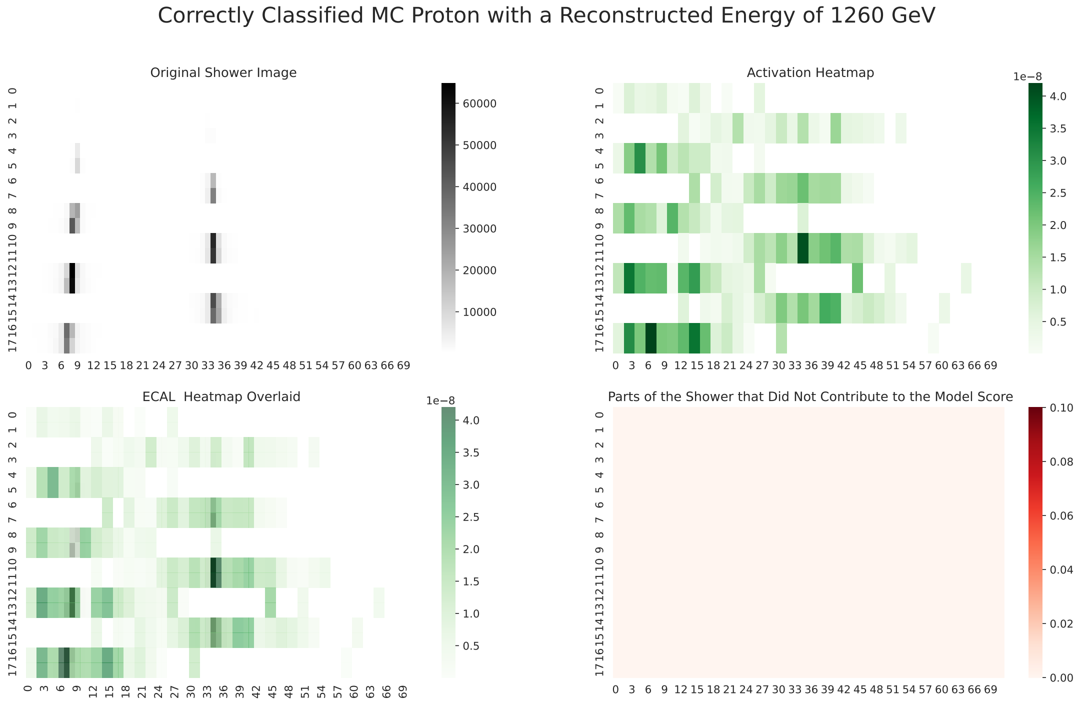


Figure 5.29: Example of a correctly classified MC proton shower with a reconstructed energy of 1260 GeV. The CvT is only able to identify the general area of the shower but does not pinpoint the shower maximum like when classifying electrons, thereby treating it as a background.

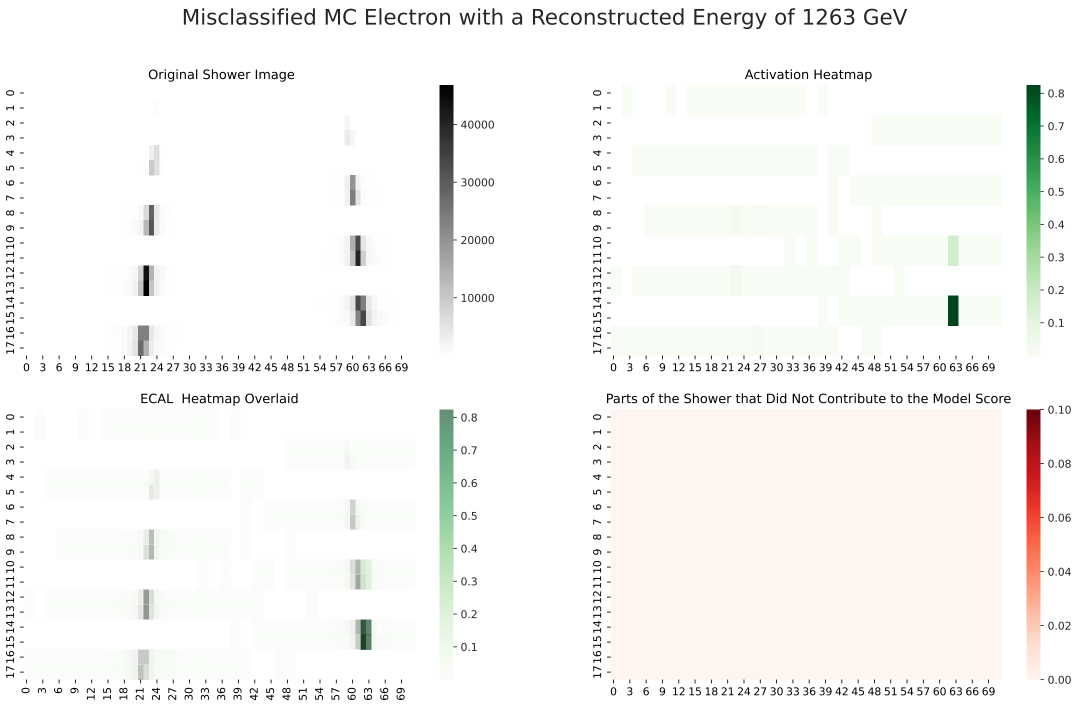


Figure 5.30: Example of a misclassified MC electron shower with a reconstructed energy of 1263 GeV. The CvT does not detect the energy depositions on the left side of the image and classifies the shower as background (i.e. a proton event).

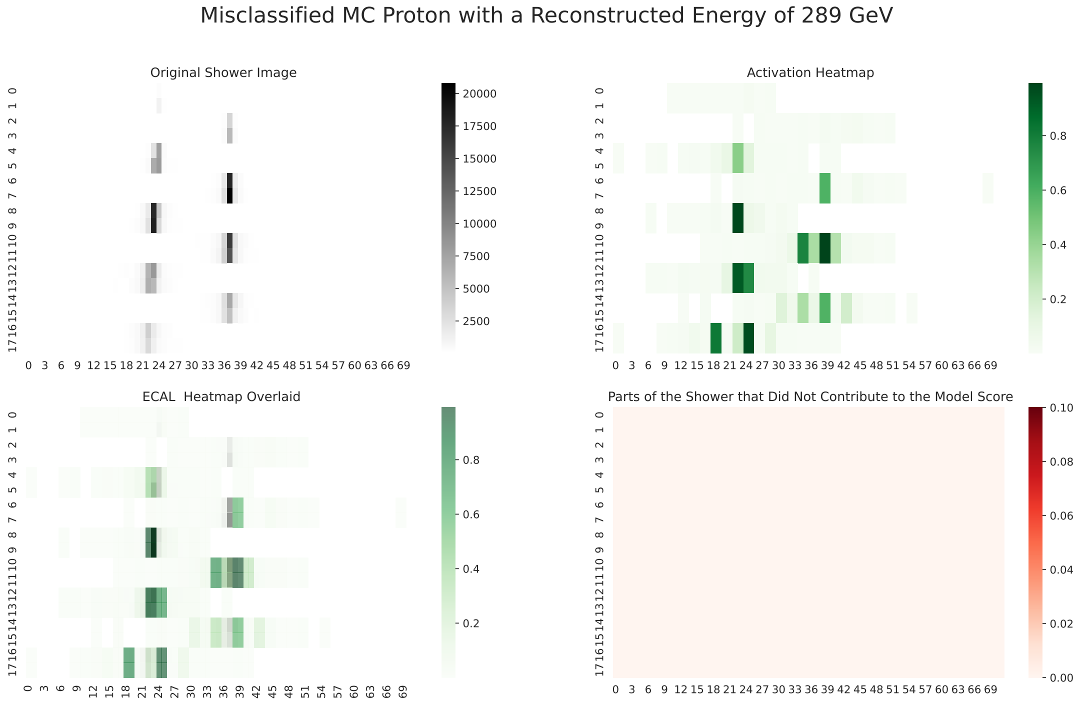


Figure 5.31: Example of a misclassified MC proton shower with a reconstructed energy of 289 GeV. MC protons are rarely misclassified by the CvT, hence the low energy compared to the previous examples. The CvT pinpoints its focus on certain cells that do not contain the shower maximum and therefore wrongly classifies the shower as an electron.

CHAPTER 6

CONCLUSION

6.1 Conclusion

We evaluated the potential of deep learning to separate electrons, and by extension positrons, from protons. We used an MLP (SimpleMLP), a CNN (SimpleCNN), two ResNets (ResNet 10 and ResNet18) and a Convolutional vision Transformer (CvT). In addition, we built a physics-based feature engineering and evaluated its performance with two CvT variants, namely Phys+CvT and Phys+CvT (small kernels).

Using the first dataset extracted, consisting of particles with a generated energy between 200–600 GeV, we established that the DL models outperformed the simple ML models (LogReg, SVM, HistBDT) using the proton rejection vs. electron efficiency metric.

Using the second dataset, consisting of MC events with a reconstructed energy between 200–2000 GeV, we showed the CvT has an excellent performance compared to both the other DL models and current AMS models. The CvT's ability to perform well on above 1 TeV while only training on below 1 TeV particles gives evidence of its potential to generalize and learn shower shapes without being overly dependent on the energy. In addition, using this dataset, we showed that the 1-channel representation is just as effective as using the 2-channel representation, but with half the input pixels, and demonstrated the need for a large amount of training data to effectively train a CvT model.

Using the third dataset, which had the same energy range as the second dataset but re-extracted with additional Tracker variables, we developed a physics-based feature engineering. We showed the Phys+CvT model, trained on the 100% of the train set, was comparable to the CvT model. We also showed that the Phys+CvT model, trained on the 5% of the train set, was comparable in performance to the CvT model trained on 100% of the train set when evaluated on the above 1 TeV test set. In addition, we showed the ResNet10 had a comparable performance to the ResNet18, using only half of the trainable parameters. However, it was not comparable to the CvT models which also have similar number of trainable parameters.

Finally, using the fourth dataset consisting of ISS data with a reconstructed energy between 50–70 GeV, we showed evidence of the discrepancies between MC events and ISS data by observing the poor performance of our MC-trained DL models when compared to the ISS data-trained AMS models. When we trained our DL models on a portion of the ISS data, we observed better performance from almost all of the DL models when compared to the AMS models, with the Phys+CvT model outperforming the best AMS model by a factor of 2.5. This shows that, at least for this small range of ISS data, the physics-based feature engineering improved learning performance for the CvT and that the Phys+CvT model shows potential to more accurately separate positrons/electrons from protons.

In conclusion, we have provided empirical evidence of newer architectures, such as the CvT, being a viable alternative to the commonly used BDTs and CNNs for use in calorimeters, and that they show promise for future use in the AMS experiment.

6.2 Future Direction

In this work, the ResNet18 was used with the standard hyperparameter configurations found in literature and the Phys+CvT (small kernels) model was not properly tuned to the problem. As such, there is a potential for greater efficiency and performance if the correct set of hyperparameters tuned specifically for our problem is found. As such, additional hyperparameter optimizations can be performed to improve performance even further.

A particular problem we hope to directly tackle in the near future is the domain shift that occurs both when moving from below 1 TeV to above 1 TeV particles as well as a potential domain shift that occurs when moving from MC events to ISS data. To address these issues, Figure 6.1 shows an unsupervised domain adaption technique that can be adopted [121]. Using this method, we hope to both make a model that is even less dependent on energy and robust towards the differences between MC events and ISS data. In addition, we hope to use this method to train on high energy ISS data, for which true labeling is difficult.

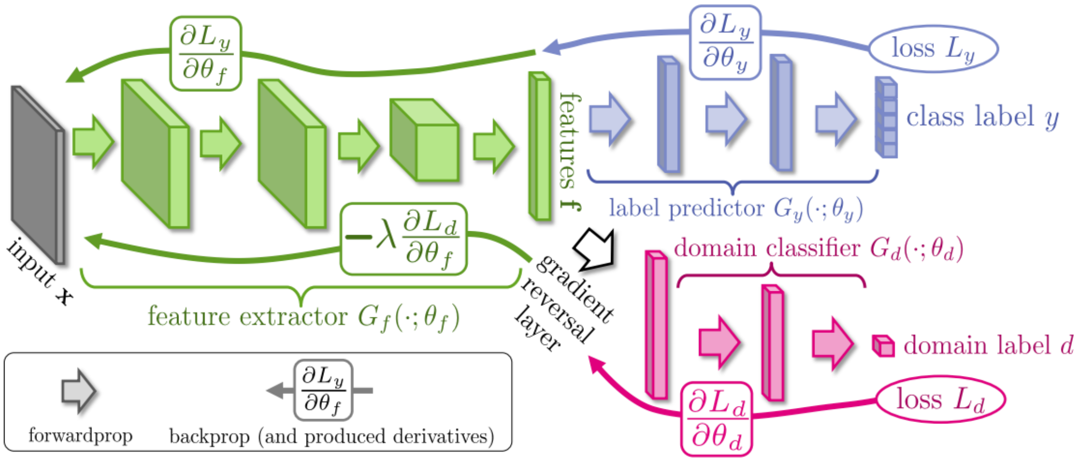

$\frac{\partial L_y}{\partial \theta_f}$
$\frac{\partial L_y}{\partial \theta_y}$
loss L_y
class label y
features f
label predictor $G_y(\cdot;\theta_y)$
$-\lambda\frac{\partial L_d}{\partial \theta_f}$
input x
feature extractor $G_f(\cdot;\theta_f)$
gradient reversal layer
domain classifier $G_d(\cdot;\theta_d)$
domain label d
loss L_d
$\frac{\partial L_d}{\partial \theta_d}$
forwardprop
backprop (and produced derivatives)

Figure 6.1: An unsupervised domain adaption technique [121]. The green portion represents feature extraction while the blue portion represents classification, and together they represent a standard classification ML model. The pink branch is an additional classifier for the *domain* of the data rather than the class itself, i.e. whether it is from MC or ISS, or below or above 1 TeV. This allows training to be done unsupervised, without the need for individual class labels, only the domain label. The loss on the domain is subtracted from the loss on the class label, and therefore, minimizing the total loss results in a balance between reducing the class loss but keeping the domain loss up to get an overall minimal value. This allows the model to improve class label classification while learning domain-invariant features that increase the loss on the domain label.

After extracting a larger reconstructed energy range of ISS data, which will require more physics understanding, and completing the above steps, we will once again evaluate our deep learning models. Our results will then be shown to the AMS collaboration for potential use in their future physics analyses.

REFERENCES

[1] S. Ting, “The Alpha Magnetic Spectrometer on the International Space Station”, *Nucl. Phys. B. - Proceedings Supplements*, vol. 243-244, pp. 12–24, Oct. 2013. DOI: `10.1016/j.nuclphysbps.2013.09.028`.

[2] AMS Collaboration, “AMS-02”, *ams02.space*, 2023.

[3] M. Aguilar, L. Ali Cavasonza, G. Ambrosi, *et al.*, “Towards Understanding the Origin of Cosmic-Ray Positrons”, *Phys. Rev. Lett.*, vol. 122, no. 4, p. 041 102, Jan. 2019. DOI: `10.1103/PhysRevLett.122.041102`.

[4] M. Aguilar, L. Ali Cavasonza, G. Ambrosi, *et al.*, “The Alpha Magnetic Spectrometer (AMS) on the International Space Station: Part II — Results from the First Seven Years”, *Phys. Rep.*, vol. 894, pp. 1–116, Feb. 2021, ISSN: 0370-1573. DOI: `10.1016/j.physrep.2020.09.003`.

[5] R. Trotta, G. Jóhannesson, I. V. Moskalenko, *et al.*, “Constraints on Cosmic-Ray Propagation Models from a Global Bayesian Analysis”, *Astrophys. J.*, vol. 729, no. 2, p. 106, Mar. 2011, ISSN: 0004-637X. DOI: `10.1088/0004-637X/729/2/106`.

[6] J. Kopp, “Constraints on Dark Matter Annihilation from AMS-02 Results”, *Phys. Rev. D*, vol. 88, no. 7, p. 076 013, Oct. 2013, ISSN: 15507998. DOI: `10.1103/PhysRevD.88.076013`.

[7] T. Kirn, “The AMS-02 TRD on the International Space Station”, *Nucl. Instrum. Methods Phys. Res. A*, vol. 706, pp. 43–47, Apr. 2013, ISSN: 01689002. DOI: `10.1016/J.NIMA.2012.05.010`.

[8] C. Adloff, L. Basara, G. Bigongiari, *et al.*, “The AMS-02 Lead-Scintillating Fibres Electromagnetic Calorimeter”, *Nucl. Instrum. Methods Phys. Res. A*, vol. 714, pp. 147–154, Jun. 2013, ISSN: 0168-9002. DOI: `10.1016/J.NIMA.2013.02.020`.

[9] K. He, X. Zhang, S. Ren, *et al.*, "Deep Residual Learning for Image Recognition", in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016. DOI: 10.48550/arXiv.1512.03385.

[10] H. Wu, B. Xiao, N. Codella, *et al.*, "CvT: Introducing Convolutions to Vision Transformers", *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 22–31, 2021, ISSN: 15505499. DOI: 10.1109/ICCV48922.2021.00009.

[11] V. F. Hess, "Über Beobachtungen der durchdringenden Strahlung bei sieben Freiballonfahrten", *Phys.Z.*, vol. 13, pp. 1084–1091, 1912.

[12] A. W. Strong, I. V. Moskalenko, and V. S. Ptuskin, "Cosmic-Ray Propagation and Interactions in the Galaxy", *Annu. Rev. Nucl. Part. Sci.*, vol. 57, pp. 285–327, Oct. 2007, ISSN: 01638998. DOI: 10.1146/annurev.nucl.57.090506.123011.

[13] M. G. Aartsen, M. Ackermann, J. Adams, *et al.*, "Multimessenger Observations of a Flaring Blazar Coincident with High-Energy Neutrino IceCube-170922A", *Science*, vol. 361, no. 6398, p. 147, Jul. 2018, ISSN: 10959203. DOI: 10.1126/science.aat1378.

[14] C. D. Anderson, "The Positive Electron", *Phys. Rev.*, vol. 43, no. 6, pp. 491–494, Mar. 1933, ISSN: 0031899X. DOI: 10.1103/PhysRev.43.491.

[15] A. Kounine, "The Alpha Magnetic Spectrometer on the International Space Station", *Int. J. Mod. Phys. E*, vol. 21, no. 8, Aug. 2012, ISSN: 02183013. DOI: 10.1142/S0218301312300056.

[16] F. Donato, N. Fornengo, and P. Salati, "Antideuterons as a Signature of Supersymmetric Dark Matter", *Phys. Rev. D*, vol. 62, no. 4, p. 043 003, Jul. 2000, ISSN: 05562821. DOI: 10.1103/PhysRevD.62.043003.

[17] M. S. Turner and F. Wilczek, "Positron Line Radiation as a Signature of Particle Dark Matter in the Halo", *Phys. Rev. D*, vol. 42, no. 4, p. 1001, Aug. 1990, ISSN: 05562821. DOI: 10.1103/PhysRevD.42.1001.

[18] P. Blasi, "The Origin of Galactic Cosmic Rays", *Astron. Astrophys. Rev.*, vol. 21, no. 1, pp. 1–73, Nov. 2013, ISSN: 09354956. DOI: 10.1007/s00159-013-0070-7.

[19] E. Postacı, “Conversion Mode Photon Analysis Using the Alpha Magnetic Spectrometer (AMS-02)”, M.S. thesis, Middle East Technical University, 2014.

[20] C. Türkoğlu, “Calorimetric Mode Photon Analysis Using the Alpha Magnetic Spectrometer (AMS-02)”, M.S. thesis, Middle East Technical University, 2015.

[21] Ç. Konak, “Measurement of the Proton Flux and Variability in Low Earth Orbit with the Alpha Magnetic Spectrometer”, M.S. thesis, Middle East Technical University, 2019.

[22] G. Karagöz, “Measurement of Helium Flux and Variability in Low Earth Orbit with the Alpha Magnetic Spectrometer”, M.S. thesis, Middle East Technical University, 2022.

[23] R. K. Hashmani, M. Konyushikhin, B. Shan, *et al.*, “New Monitoring Interface for the AMS Experiment”, *Nucl. Instrum. Methods Phys. Res. A*, vol. 1046, p. 167 704, Jan. 2023, ISSN: 0168-9002. DOI: `10.1016/J.NIMA.2022.167704`.

[24] M. Aguilar, J. Alcaraz, J. Allaby, *et al.*, “A Study of Cosmic Ray Secondaries Induced by the Mir Space Station Using AMS-01”, *Nucl. Instrum. Methods Phys. Res. B*, vol. 234, no. 3, pp. 321–332, Jun. 2005, ISSN: 0168-583X. DOI: `10.1016/J.NIMB.2005.01.015`.

[25] J. Alcaraz, B. Alpat, G. Ambrosi, *et al.*, “Cosmic Protons”, *Phys. Lett. B*, vol. 490, no. 1-2, pp. 27–35, Sep. 2000, ISSN: 0370-2693. DOI: `10.1016/S0370-2693(00)00970-9`.

[26] M. Aguilar, J. Alcaraz, J. Allaby, *et al.*, “The Alpha Magnetic Spectrometer (AMS) on the International Space Station: Part I – Results from the Test Flight on the Space Shuttle”, *Phys. Rep.*, vol. 366, no. 6, pp. 331–405, Aug. 2002, ISSN: 0370-1573. DOI: `10.1016/S0370-1573(02)00013-3`.

[27] G. Gallucci and (for the AMS-02 ECAL group), “Performance of the AMS-02 Electromagnetic Calorimeter in Space”, *J. Phys: Conf. Ser.*, vol. 587, no. 1, p. 012 028, Feb. 2015, ISSN: 1742-6596. DOI: `10.1088/1742-6596/587/1/012028`.

[28] F. Hauler, A. Bartoloni, U. Becker, *et al.*, “The AMS-02 TRD for the International Space Station”, *IEEE Trans. Nucl. Sci.*, vol. 51, no. 4 I, pp. 1365–1372, Aug. 2004, ISSN: 00189499. DOI: 10.1109/TNS.2004.832901.

[29] M. B. Demirköz, “A Transition Radiation Detector and Gas Supply System for AMS”, M.S. thesis, Massachusetts Institute of Technology, 2004.

[30] T. Räihä, A. Bachlechner, B. Beischer, *et al.*, “Monte Carlo Simulations of the Transition Radiation Detector of the AMS-02 Experiment”, *Nucl. Instrum. Methods Phys. Res. A*, vol. 868, pp. 10–14, Oct. 2017, ISSN: 0168-9002. DOI: 10.1016/J.NIMA.2017.06.042.

[31] AMS Collaboration, “AMS on ISS: Construction of a Particle Physics Detector on the International Space Station”, *AMS Internal Note*, 2004.

[32] J. Alcaraz, B. Alpat, G. Ambrosi, *et al.*, “The Alpha Magnetic Spectrometer Silicon Tracker: Performance Results with Protons and Helium Nuclei”, *Nucl. Instrum. Methods Phys. Res. A*, vol. 593, no. 3, pp. 376–398, Aug. 2008, ISSN: 0168-9002. DOI: 10.1016/J.NIMA.2008.05.015.

[33] R. L. Workman *et al.*, *34. Passage of Particles Through Matter*. PTEP, 2022, vol. 2022, pp. 549–564. DOI: 10.1093/ptep/ptac097.

[34] K. Lübelsmeyer, A. Schultz Von Dratzig, M. Wlochal, *et al.*, “Upgrade of the Alpha Magnetic Spectrometer (AMS-02) for long term operation on the International Space Station (ISS)”, *Nucl. Instrum. Methods Phys. Res. A*, vol. 654, no. 1, pp. 639–648, Oct. 2011, ISSN: 0168-9002. DOI: 10.1016/J.NIMA.2011.06.051.

[35] D. Casadei, L. Baldini, V. Bindi, *et al.*, “The AMS-02 Time of Flight system”, *Nucl. Phys. B. - Proceedings Supplements*, vol. 113, no. 1-3, pp. 133–138, Dec. 2002, ISSN: 0920-5632. DOI: 10.1016/S0920-5632(02)01832-7.

[36] P. von Doetinchem, W. Karpinski, T. Kirn, *et al.*, “The AMS-02 Anticoincidence Counter”, *Nucl. Phys. B. - Proceedings Supplements*, vol. 197, no. 1, pp. 15–18, Dec. 2009, ISSN: 0920-5632. DOI: 10.1016/J.NUCLPHYSBPS.2009.10.025.

[37] M. Aguilar-Benitez, L. Arruda, F. Barao, *et al.*, “In-beam Aerogel Light Yield Characterization for the AMS RICH Detector”, *Nucl. Instrum. Methods Phys. Res. A*, vol. 614, no. 2, pp. 237–249, Mar. 2010, ISSN: 0168-9002. DOI: 10.1016/J.NIMA.2009.12.027.

[38] W. R. Leo, “Techniques for Nuclear and Particle Physics Experiments”, *Techniques for Nuclear and Particle Physics Experiments*, 1994. DOI: 10.1007/978-3-642-57920-2.

[39] M. S. Krafczyk, “A Precision Measurement of the Cosmic Ray Positron Fraction on the International Space Station”, Ph.D. dissertation, Massachusetts Institute of Technology, 2016.

[40] A. Kounine, Z. Weng, W. Xu, *et al.*, “Precision Measurement of 0.5 GeV–3 TeV Electrons and Positrons Using the AMS Electromagnetic Calorimeter”, *Nucl. Instrum. Methods Phys. Res. A*, vol. 869, pp. 110–117, Oct. 2017, ISSN: 0168-9002. DOI: 10.1016/J.NIMA.2017.07.013.

[41] R. E. Schapire, “The Boosting Approach to Machine Learning: An Overview”, in *Lecture Notes in Statistics*, vol. 171, Springer, New York, NY, 2003, pp. 149–171. DOI: 10.1007/978-0-387-21579-2_9.

[42] B. P. Roe, H. J. Yang, J. Zhu, *et al.*, “Boosted Decision Trees as an Alternative to Artificial Neural Networks for Particle Identification”, *Nucl. Instrum. Methods Phys. Res. A*, vol. 543, no. 2-3, pp. 577–584, May 2005, ISSN: 0168-9002. DOI: 10.1016/J.NIMA.2004.12.018.

[43] P. D. Serpico, “Astrophysical Models for the Origin of the Positron Excess”, *Astropart. Phys.*, vol. 39-40, no. 1, pp. 2–11, Dec. 2012, ISSN: 0927-6505. DOI: 10.1016/J.ASTROPARTPHYS.2011.08.007.

[44] K. Ioka, “A Gamma-Ray Burst/Pulsar for Cosmic Ray Positrons with a Dark Matter-Like Spectrum”, *Prog. Theor. Phys.*, vol. 123, no. 4, pp. 743–755, Apr. 2010, ISSN: 0033-068X. DOI: 10.1143/PTP.123.743.

[45] T. Linden and S. Profumo, “Probing the Pulsar Origin of the Anomalous Positron Fraction with AMS-02 and Atmospheric Cherenkov Telescopes”, *Astrophys. J.*, vol. 772, no. 1, Jul. 2013, ISSN: 15384357. DOI: 10.1088/0004-637X/772/1/18.

[46] P. Lipari, "Interpretation of the Cosmic Ray Positron and Antiproton Fluxes", *Phys. Rev. D*, vol. 95, no. 6, p. 063 009, Mar. 2017, ISSN: 24700029. DOI: 10.1103/PhysRevD.95.063009.

[47] M. Aguilar, G. Alberti, B. Alpat, *et al.*, "First Result from the Alpha Magnetic Spectrometer on the International Space Station: Precision Measurement of the Positron Fraction in Primary Cosmic Rays of 0.5-350 GeV", *Phys. Rev. Lett.*, vol. 110, no. 14, p. 141 102, Apr. 2013, ISSN: 00319007. DOI: 10.1103/PhysRevLett.110.141102.

[48] M. Aguilar, D. Aisa, A. Alvino, *et al.*, "Electron and Positron Fluxes in Primary Cosmic Rays Measured with the Alpha Magnetic Spectrometer on the International Space Station", *Phys. Rev. Lett.*, vol. 113, no. 12, p. 121 102, Sep. 2014, ISSN: 10797114. DOI: 10.1103/PhysRevLett.113.121102.

[49] C. Evoli, E. Amato, P. Blasi, *et al.*, "Galactic Factories of Cosmic-Ray Electrons and Positrons", *Phys. Rev. D*, vol. 103, no. 8, p. 083 010, Apr. 2021, ISSN: 24700029. DOI: 10.1103/PhysRevD.103.083010.

[50] L. Orusa, S. Manconi, F. Donato, *et al.*, "Constraining Positron Emission from Pulsar Populations with AMS-02 Data", *J. Cosmol. Astropart. Phys.*, vol. 2021, no. 12, p. 014, Dec. 2021, ISSN: 1475-7516. DOI: 10.1088/1475-7516/2021/12/014.

[51] S. Gabici, C. Evoli, D. Gaggero, *et al.*, "The origin of Galactic Cosmic Rays: Challenges to the Standard paradigm", *Int. J. Mod. Phys. D*, vol. 28, no. 15, p. 1 939 922, Dec. 2019. DOI: 10.1142/S0218271819300222.

[52] M. Kachelrieß and D. V. Semikoz, "Cosmic Ray Models", *Prog. Part. Nucl. Phys.*, vol. 109, p. 103 710, Nov. 2019, ISSN: 0146-6410. DOI: 10.1016/j.ppnp.2019.07.002.

[53] T. M. Mitchell, *Machine Learning*, 1st. McGraw-Hill, 1997.

[54] A. L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers", *IBM J. Res. Dev.*, vol. 3, no. 3, pp. 210–229, Jul. 1959, ISSN: 0018-8646. DOI: 10.1147/RD.33.0210.

[55] W. S. McCulloch and W. Pitts, "A Logical Calculus of the Ideas Immanent in Nervous Activity", *Bull. Math. Biophys.*, vol. 5, no. 4, pp. 115–133, Dec. 1943, ISSN: 00074985. DOI: `10.1007/BF02478259/METRICS`.

[56] F. Rosenblatt, "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain", *Psychol. Rev.*, vol. 65, no. 6, pp. 386–408, Nov. 1958, ISSN: 0033295X. DOI: `10.1037/H0042519`.

[57] Marvin Minsky and Seymour A. Papert, *Perceptrons: An Introduction to Computational Geometry*, 1st. MIT Press, 1969.

[58] A. G. Ivachnenko and V. G. Lapa, *Cybernetics and Forecasting Techniques*. American Elsevier, 1967, ISBN: 0444000208.

[59] S. Jain, *What is Neuron and Artificial Neuron in Deep Learning?*, Nov. 2017.

[60] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Representations by Back-Propagating Errors", *Nature 1986 323:6088*, vol. 323, no. 6088, pp. 533–536, 1986, ISSN: 1476-4687. DOI: `10.1038/323533a0`.

[61] S. Linnainmaa, "Taylor Expansion of the Accumulated Rounding Error", *BIT*, vol. 16, no. 2, pp. 146–160, 1976. DOI: `10.1007/BF01931367`.

[62] P. J. Werbos, *The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting*. Wiley, Mar. 1994, p. 319, ISBN: 978-0-471-59897-8.

[63] K. Hornik, M. Stinchcombe, and H. White, "Multilayer Feedforward Networks are Universal Approximators", *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jan. 1989, ISSN: 0893-6080. DOI: `10.1016/0893-6080(89)90020-8`.

[64] C. Cortes and V. Vapnik, "Support-Vector Networks", *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995. DOI: `10.1007/BF00994018`.

[65] Y. Lecun, Y. Bengio, and G. Hinton, "Deep Learning", *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015. DOI: `10.1038/nature14539`.

[66] G. Hinton, L. Deng, D. Yu, *et al.*, "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups", *IEEE Signal Process Mag.*, vol. 29, no. 6, pp. 82–97, 2012, ISSN: 10535888. DOI: `10.1109/MSP.2012.2205597`.

[67] D. Cireşan, U. Meier, J. Masci, *et al.*, “Multi-Column Deep Neural Network for Traffic Sign Classification”, *Neural Netw.*, vol. 32, pp. 333–338, Aug. 2012, ISSN: 0893-6080. DOI: `10.1016/J.NEUNET.2012.02.023`.

[68] J. Deng, W. Dong, R. Socher, *et al.*, “ImageNet: A Large-Scale Hierarchical Image Database”, pp. 248–255, Mar. 2010. DOI: `10.1109/CVPR.2009.5206848`.

[69] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks”, *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, ISSN: 0001-0782. DOI: `10.1145/3065386`.

[70] K. S. Oh and K. Jung, “GPU Implementation of Neural Networks”, *Pattern Recogn.*, vol. 37, no. 6, pp. 1311–1314, Jun. 2004, ISSN: 0031-3203. DOI: `10.1016/J.PATCOG.2004.01.013`.

[71] BBC News, *Google AI Defeats Human Go Champion*, May 2017.

[72] M. Eisenstein, “Artificial Intelligence Powers Protein-Folding Predictions”, *Nature*, vol. 599, no. 7886, pp. 706–708, Nov. 2021, ISSN: 14764687. DOI: `10.1038/D41586-021-03499-Y`.

[73] R. Rombach, A. Blattmann, D. Lorenz, *et al.*, “High-Resolution Image Synthesis With Latent Diffusion Models”, in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 10 684–10 695.

[74] D. Bourilkov, “Machine and Deep Learning Applications in Particle Physics”, *Int J Mod Phys A*, vol. 34, no. 35, p. 1 930 019, Jan. 2020. DOI: `10.1142/S0217751X19300199`.

[75] B. Denby, “Neural Networks and Cellular Automata in Experimental High Energy Physics”, *Comput. Phys. Commun.*, vol. 49, no. 3, pp. 429–448, Jun. 1988, ISSN: 0010-4655. DOI: `10.1016/0010-4655(88)90004-5`.

[76] L. Lönnblad, C. Peterson, and T. Rögnvaldsson, “Finding Gluon Jets with a Neural Trigger”, *Phys. Rev. Lett.*, vol. 65, no. 11, p. 1321, Sep. 1990, ISSN: 00319007. DOI: `10.1103/PhysRevLett.65.1321`.

[77] L. Lönnblad, C. Peterson, and T. Rögnvaldsson, “Using Neural Networks to Identify Jets”, *Nucl. Phys. B*, vol. 349, no. 3, pp. 675–702, Feb. 1991, ISSN: 0550-3213. DOI: `10.1016/0550-3213(91)90392-B`.

[78] H. J. Yang, B. P. Roe, and J. Zhu, "Studies of Boosted Decision Trees for MiniBooNE Particle Identification", *Nucl. Instrum. Methods Phys. Res. A*, vol. 555, no. 1-2, pp. 370–385, Dec. 2005, ISSN: 0168-9002. DOI: 10.1016/J.NIMA.2005.09.022.

[79] P. C. Bhat, "Multivariate Analysis Methods in Particle Physics*", *Annu. Rev. Nucl. Part. Sci.*, vol. 61, pp. 281–309, Nov. 2011, ISSN: 01638998. DOI: 10.1146/annurev.nucl.012809.104427.

[80] P. Baldi, P. Sadowski, and D. Whiteson, "Searching for Exotic Particles in High-Energy Physics with Deep Learning", *Nat. Commun. 2014 5:1*, vol. 5, no. 1, pp. 1–9, Jul. 2014, ISSN: 2041-1723. DOI: 10.1038/ncomms5308.

[81] D. Guest, K. Cranmer, and D. Whiteson, "Deep Learning and Its Application to LHC Physics", *Ann. Rev.*, vol. 68, pp. 161–181, Oct. 2018, ISSN: 01638998. DOI: 10.1146/ANNUREV-NUCL-101917-021019.

[82] J. Collins, K. Howe, and B. Nachman, "Anomaly Detection for Resonant New Physics with Machine Learning", *Phys. Rev. Lett.*, vol. 121, no. 24, p. 241 803, Dec. 2018, ISSN: 10797114. DOI: 10.1103/PhysRevLett.121.241803.

[83] M. Andrews, M. Paulini, S. Gleyzer, *et al.*, "End-to-End Event Classification of High-Energy Physics Data", *J. Phys: Conf. Ser.*, vol. 1085, no. 4, p. 042 022, Sep. 2018. DOI: 10.1088/1742-6596/1085/4/042022.

[84] M. Huennefeld, R. Abbasi, M. Ackermann, *et al.*, "Combining Maximum-Likelihood with Deep Learning for Event Reconstruction in IceCube", vol. 395, p. 1065, Mar. 2022. DOI: 10.22323/1.395.1065.

[85] L. de Oliveira, M. Paganini, and B. Nachman, "Controlling Physical Attributes in GAN-Accelerated Simulation of Electromagnetic Calorimeters", *J. Phys: Conf. Ser.*, vol. 1085, no. 4, p. 042 017, 2018, ISSN: 1742-6596. DOI: 10.1088/1742-6596/1085/4/042017.

[86] O. Pastor-Serrano and Z. Perkó, "Learning the Physics of Particle Transport via Transformers", *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 11, pp. 12 071–12 079, Jun. 2022, ISSN: 2374-3468. DOI: 10.1609/AAAI.V36I11.21466.

[87] A. Alnuqaydan, S. Gleyzer, and H. Prosper, "SYMBA: Symbolic Computation of Squared Amplitudes in High Energy Physics with Machine Learning", Jun. 2022. DOI: 10.48550/arxiv.2206.08901.

[88] V. Mikuni and F. Canelli, "Point Cloud Transformers Applied to Collider Physics", *Mach. Learn.: Sci. Technol.*, vol. 2, no. 3, p. 035 027, Jul. 2021, ISSN: 2632-2153. DOI: 10.1088/2632-2153/ac07f6.

[89] H. Qu, C. Li, and S. Qian, "Particle Transformer for Jet Tagging", Feb. 2022. DOI: 10.48550/arxiv.2202.03772.

[90] F. Pedregosa, G. Varoquaux, A. Gramfort, *et al.*, "Scikit-learn: Machine Learning in Python", *J. Mach. Learn. Res.*, vol. 12, no. 85, pp. 2825–2830, 2011.

[91] D. R. Cox, "The Regression Analysis of Binary Sequences", *J. R. Stat. Soc. Series. B. Stat. Methodol.*, vol. 20, no. 2, pp. 215–232, Jul. 1958, ISSN: 2517-6161. DOI: 10.1111/J.2517-6161.1958.TB00292.X.

[92] G. Ke, Q. Meng, T. Finley, *et al.*, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree", in *Adv. Neural. Inf. Process. Syst.*, vol. 30, 2017.

[93] Z. Wu, C. Shen, and A. van den Hengel, "Wider or Deeper: Revisiting the ResNet Model for Visual Recognition", *Pattern Recogn.*, vol. 90, pp. 119–133, Jun. 2019, ISSN: 0031-3203. DOI: 10.1016/J.PATCOG.2019.01.006.

[94] F. Xue, J. Chen, A. Sun, *et al.*, "Deeper vs Wider: A Revisit of Transformer Configuration", May 2022. DOI: 10.48550/arxiv.2205.10505.

[95] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation Functions in Deep Learning: A Comprehensive Survey and Benchmark", *Neurocomputing*, vol. 503, pp. 92–108, Sep. 2022, ISSN: 0925-2312. DOI: 10.1016/J.NEUCOM.2022.06.111.

[96] T. Szandała, "Review and Comparison of Commonly Used Activation Functions for Deep Neural Networks", *Stud. Comput. Intell.*, vol. 903, pp. 203–224, 2021, ISSN: 18609503. DOI: 10.1007/978-981-15-5495-7_11.

[97] B. Ding, H. Qian, and J. Zhou, "Activation Functions and their Characteristics in Deep Neural Networks", *Proceedings of the 30th Chinese Control and Decision Conference, CCDC 2018*, pp. 1836–1841, Jul. 2018. DOI: 10.1109/CCDC.2018.8407425.

[98] A. M. Fred Agarap, “Deep Learning using Rectified Linear Units (ReLU)”, Mar. 2018. DOI: 10.48550/arxiv.1803.08375.

[99] A. Zhang, Z. C. Lipton, M. Li, *et al.*, “Dive into Deep Learning”, *arXiv preprint arXiv:2106.11342*, 2021.

[100] M. Ranzato, F. J. Huang, Y.-L. Boureau, *et al.*, “Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition”, in *2007 Proc. IEEE Int. Conf. Comput. Vis.*, IEEE, Jun. 2007, pp. 1–8, ISBN: 1-4244-1179-3. DOI: 10.1109/CVPR.2007.383157.

[101] S. Basodi, C. Ji, H. Zhang, *et al.*, “Gradient Amplification: An Efficient Way to Train Deep Neural Networks”, *Big Data Mining and Analytics*, vol. 3, no. 3, pp. 196–207, Sep. 2020. DOI: 10.26599/BDMA.2020.9020004.

[102] A. Vaswani, G. Brain, N. Shazeer, *et al.*, “Attention is All you Need”, *Adv. Neur. In.*, vol. 30, 2017.

[103] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”, Oct. 2020. DOI: 10.48550/arxiv.2010.11929.

[104] V. Choutko, O. Demakov, A. Egorov, *et al.*, “Storage Strategy of AMS Science Data at Science Operation Centre at CERN”, *J. Phys: Conf. Ser.*, vol. 898, no. 6, p. 062 039, Oct. 2017, ISSN: 1742-6596. DOI: 10.1088/1742-6596/898/6/062039.

[105] R. Brun and F. Rademakers, “ROOT — An Object Oriented Data Analysis Framework”, *Nucl. Instrum. Methods Phys. Res. A*, vol. 389, no. 1-2, pp. 81–86, Apr. 1997. DOI: 10.1016/S0168-9002(97)00048-X.

[106] Rene Brun and Fons Rademakers, *ROOT Version 6.22 Release Notes*, Aug. 2020.

[107] M. Galli, E. Tejedor, and S. Wunsch, “A New PyROOT: Modern, Interoperable and More Pythonic”, *EPJ Web Conf.*, vol. 245, p. 06 004, 2020, ISSN: 2100-014X. DOI: 10.1051/EPJCONF/202024506004.

[108] A. Paszke, S. Gross, F. Massa, *et al.*, “PyTorch: An Imperative Style, High-Performance Deep Learning Library”, in *Adv. Neural. Inf. Process. Syst. 32*, 2019, pp. 8024–8035.

[109] J. D. Hunter, "Matplotlib: A 2D Graphics Environment", *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 90–95, 2007. DOI: 10.1109/MCSE.2007.55.

[110] M. L. Waskom, "seaborn: statistical data visualization", *J. Open Source Softw*, vol. 6, no. 60, p. 3021, Apr. 2021. DOI: 10.21105/JOSS.03021.

[111] C. Z. Mooney, *Monte Carlo Simulation*. Sage Publications, 1997.

[112] V. Choutko, A. Eogorov, A. Eline, *et al.*, "Offline Software Management of the AMS Experiment", *EPJ Web Conf.*, vol. 245, p. 05 033, 2020, ISSN: 2100-014X. DOI: 10.1051/EPJCONF/202024505033.

[113] S. Agostinelli, J. Allison, K. Amako, *et al.*, "Geant4 - A Simulation Toolkit", *Nucl. Instrum. Methods Phys. Res. A*, vol. 506, no. 3, pp. 250–303, Jul. 2003, ISSN: 0168-9002. DOI: 10.1016/S0168-9002(03)01368-8.

[114] AMS Collaboration, "Charge Sign Identification Improvement in the Tracker Analysis", *ams02.space*, Jul. 2019.

[115] T. Fawcett, "An Introduction to ROC Analysis", *Pattern Recogn. Lett.*, vol. 27, no. 8, Jun. 2006. DOI: 10.1016/j.patrec.2005.10.010.

[116] A. Chilingarian, "Neural Classification Technique for Background Rejection in High Energy Physics Experiments", *Neurocomputing*, vol. 6, no. 5-6, pp. 497–512, Oct. 1994. DOI: 10.1016/0925-2312(94)90002-7.

[117] D. Guest, J. Collado, P. Baldi, *et al.*, "Jet Flavor Classification in High-Energy Physics with Deep Neural Networks", *Phys. Rev. D*, vol. 94, no. 11, p. 112 002, Dec. 2016, ISSN: 24700029. DOI: 10.1103/PhysRevD.94.112002.

[118] M. Kekic, C. Adams, K. Woodruff, *et al.*, "Demonstration of Background Rejection Using Deep Convolutional Neural Networks in the NEXT Experiment", *J. High Energy Physics 2021 2021:1*, vol. 2021, no. 1, pp. 1–22, Jan. 2021, ISSN: 1029-8479. DOI: 10.1007/JHEP01(2021)189.

[119] OpenAI, "Deep Double Descent", *openai.com*, Dec. 2019.

[120] R. R. Selvaraju, M. Cogswell, A. Das, *et al.*, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization", *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2017-October, pp. 618–626, Dec. 2017, ISSN: 15505499. DOI: 10.1109/ICCV.2017.74.

[121] Y. Ganin and V. Lempitsky, “Unsupervised Domain Adaptation by Backpropagation”, *32nd International Conference on Machine Learning, ICML 2015*, vol. 2, pp. 1180–1189, Sep. 2014. DOI: `10.48550/arxiv.1409.7495`.

APPENDIX A

EVENT SELECTION CUTS FOR DATA EXTRACTION

Certain standard event acceptance cuts and criteria were taken from AMS's private Twiki page, which is unfortunately not accessible to the general public. Tables A.1 and A.2 contain some of these standard cuts and criteria, along with our specific cuts used to accept only certain type of events. Additionally, an acceptance energy cut is also applied, depending on the type of energy (generated or reconstructed) and the specific energy range required for each of our 4 extracted datasets.

Table A.1: List of event acceptance cuts for both MC events and ISS data.

Cut Description	Cut Value
No. of ECAL Showers per Event	nEcalShower() ≥ 1
Minimum Energy in ECAL	E ≥ 10 GeV
ECAL Shower is Contained	ecalEntry and ecalExit ≥ 3 cells from the border
No. of Tracks in the Tracker	nTrTrack() $= 1$
Track Matches ECAL Shower Position	fabs(deltaX) < 3 & fabs(deltaY)< 10
ToF and Tracker Match	betah->IsTkTofMatch() $=$ True
No. of TRD Tracks	nTrdHTrack() ≥ 1
Remove Non-relativistic Particles	betah_tof > 0.01

Table A.2: List of additional event acceptance cuts for ISS data, used alongside cuts in Table A.1.

	Cut Description	Cut Value
Tracker	Certain Hits on Tracker	Hits on L2 & (L3 or L4) & (L5 or L6) & (L7 or L8)
	Cut on Inner Tracker Charge	$0.7 <$ Q$_{inn}$ < 1.4
	Cut on Inner Tracker Charge RMS	Q_RMS$_{inn}$ < 0.4
	Cut on χ^2 Electron and Proton Fit	chisy_electron_fit < 10 & chisy_proton_fit < 10
ToF	Remove Non-relativistic Particles	betah_tof $>= 0.04$
	Build Type	GetBuildType() $= 1$
	Cut on Upper ToF Charge	$0.5 <$ betaH_Q$_U$ < 2.5
	Good Beta	betah->IsGoodBeta() $= 1$
TRD	Particle Passes Through TRD	IsInsideTRD()$= 1$
	Read Alignment is Okay	IsReadAlignmentOK$= 1$
	Read Calibration is Okay	IsReadCalibOK$= 1$