ML4Sci
Vision Transformers for CMS

GSoC 2023 Proposal

Personal Details:

Name: Syed Salman Habeeb Quadri
Preferred Name: Salman
Pronouns: He/His
Email: quadrisyedsalmanhabeeb@gmail.com
Alternate Email: salmangithub_137@gmail.com
University: Indian Institute of Technology Roorkee
Linkedin Profile: Syed Salman Habeeb Quadri
Twitter Profile: SyedSalman_137
Github Profile: SalmanHabeeb

Introduction

I am Salman, a 20 year old from India. I am currently pursuing B.Tech. at IIT Roorkee, with majors in Production Engineering. I am currently in the second academic year. I have a great interest in deep learning, and I desire to pursue research in this field.

Why did I choose this project?

Since high school, I have been fascinated by modern physics, particularly the origins of dark matter, high-energy particle physics, and the possibility of creating black holes in the LHC. As I pursued my studies in university, I became interested in deep learning, exploring various algorithms such as Resnets, Effnets, Transformers, and GPTs.

During my exploration of deep learning, I was introduced to ML4Sci and their incredible projects, including quantum GANs, autoencoders for dark matter detection, and the use of GNNs for particle identification in CMS data. Being a part of this community would be a great opportunity for me to pursue my research in deep learning. It will also help me learn good practices in deep learning research and give me an opportunity to contribute to the scientific community.

Project Overview

CMS Data contains the history of billions of high-energy collisions taking place inside the LHC. Since the amount of data is huge, using deep learning methods to identify the particles generated from collision would be of great help.

Using deep learning requires training a model on a dataset. But deep learning algorithms like CNNs do not generalise well on the out of sample data.

This challenge can be addressed using Vision Transformers. Vision Transformers can generalise well on the out of sample data. Also their efficiency as well as accuracy is better than Resnets with the same no. of parameters.

Project Deliverables

- Training Pipelines including code for design of ViTs
- Trained ViT models
- C++ Compiled ViT models
- Report

Project Proposal

I will train Vision Transformers (ViTs) on a given dataset, with the aim of achieving a level of accuracy that is comparable to that achieved by EffNets and ResNets.

EffNets and ResNets are commonly used deep neural network architectures for computer vision tasks, known for their high accuracy. By training both EffNets and ResNets, I plan to establish a baseline level of accuracy against which the performance of ViTs can be compared.

Once the ViTs have been trained and evaluated, I will compile them for C++ using PyTorch. Compiling models in C++ enables the models to be used more efficiently on large-scale systems.

Then, I will create a report by benchmarking various models.

Technical Details

Vision Transformers (ViTs) are attention based neural networks which can attend to various patches of images. To implement this image is divided into patches. These patches of image are passed into an attention block, which gives out embeddings, known as patch embeddings.

A ViT uses multiple attention blocks to predict the result. This mechanism of handling an image is significantly different from the way it is handled by a purely convolution based neural network. A CNN based model uses a sliding window filter to extract features from an image.

For validating a model I will use Cross KFold validation. In this the data is divided into multiple ‘k’ sized sets. Each set is split into ‘k’ folds, and the model is trained on ‘k-1’ folds and evaluated on ‘k’th fold.

I will use PyTorch in Python as a framework to implement and train the models. Later, I will compile these models to C++ using PyTorch C++ API, for the community to use.

Timeline

Pre GSoC

Understand various transformer architectures like Pyramid Vision Transformer, DeiT and others.

Community Bonding

Choose at least two vision transformer variants I will use in the project. Choose other deep learning architectures for comparison with ViT. Review literature on the use of SOTA deep learning architectures on CMS experiments.

Week 1-3

Develop a training and evaluation pipeline for deep learning models. Test this pipeline.

Train Resnets and Efficient Net models on dataset.

Week 3-5

Design ViT architectures which were chosen earlier.

Train and evaluate at least one ViT variant on the dataset.

Week 5-7

Train remaining ViT architectures.

Week 8-9

Develop C++ models for Vision Transformer in PyTorch.

Week 10

Complete if any unfinished work remains. Start preparing a report.

Week 11-12

Prepare a report on the ViT performance on the dataset while comparing it with other architectures.

About Me

I have been working on Python, C++ and JavaScript. I spend my time doing research on projects I can develop using deep learning, practising PyTorch, or learning new programming languages. I have previously trained UNets for semantic segmentation on Plant Village dataset. In this project I have experimented with adding attention layers to the UNet to improve the dice score. I have also researched the use of deep learning algorithms to develop chess engines. There I tried to use deep learning algorithms to reduce the search space of the minimax algorithm.

I have trained various transformer architectures like GPT on custom data to develop a chatbot for MARS, a club at my institute. I am currently exploring the use of self-supervised deep learning algorithms like DINO and PAWS.