IRIS-HEP Summer Student Internship 2022
Towards a deep learning framework for efficient jet tagging

Kateryna Skurativska

July 2022

Introduction

The interest of the particle theory community in machine learning techniques has exploded in recent years. One of the most active topics of research has been the development of machine learning based boosted-object tagging algorithms, see e.g. [4] for a review of boosted top quark taggers. The tagging problem is a typical classification problem: given the dataset that contains objects we predict whether it contains or not a specific particle. In our case, the objects are so-called jets - a collimated spray of particles called hadrons. On the machine learning side, different proposals have been made, such as the ones based on convolutional neural networks [3] or graph neural networks (LundNet [1]). We will consider both of these approaches in the following.

Project description

In this project, we first plan to investigate the dependence of the performances of the taggers on the theoretical characteristics of the training samples (e.g. perturbative order, shower, tune of the showers, etc.). Second we asses the possibility of using transfer learning techniques in experimental analysis and provide accurate and efficient predictions, as recently pointed out in Ref. [2]. As the next step, we will consider novel model architectures based on quantum computing hardware [5, 6] and compare their performances to the classical state-of-the-art machine learning techniques. The final target of this project is the development and deployment of a software library for jet tagging benchmarking with the entire pipeline given as in the following:

- data loading and preprocessing;
- fitting abstraction for multiple model architectures and tuning of the hyperparameters;
- quality estimators in order to judge the performance of the model choice.

Project schedule

The duration of the project is a three-month period July-September 2022. This project will be supervised by Dr. Emanuele Angelo Bagnaschi (CERN) and Prof. Stefano Carrazza (University of Milano).

Week 1 - 2

We review the literature for the project background and understanding of the data characteristics. After that, we load and preprocess the input data, which are simulated with the Monte Carlo event generator to speed up the learning process. Preprocessing includes localization of the leading maxima in the images, shifting the images to the maximum global center, rotating, flipping, and cropping images to the proper size. We are creating the library with methods for loading and preprocessing data.

Week 3 - 5

Using the architecture developed in [3], we build a Convolutional Neural Network (CNN), train, validate and optimize it. We learn the dependence between different training sets and the classification accuracy for CNN.

Week 6 - 7

We change CNN's architecture to potentially improve the training performance. We apply the transfer learning approach inside the CNN to reuse the pre-trained model to detect new features.

Week 8 - 9

We consider the graph neural network LundNet as a possible solution for the problem, mentioned above. We download the implementation of LundNet and understand its mechanism, train, and validate it. After that, we perform the same study as for CNN and compare them. We add methods to fit abstraction for multiple model architectures and to tune the hyper-parameters, and methods to estimate the performance of the different model choices.

Week 10 - 11

We consider quantum machine learning models to classify images in the same way as previous models but using quantum circuit simulation through the `Qibo` framework [7]. We search the quantum model with fewer parameters than the classical counterpart, which minimizes the energy consumption.

Week 12

Wrapping up the project. Prepare a report on the results of the research. General cleanup of the library's code including in terms of readability, and documentation. Create and present a final presentation of the project.

Software Deliverables

To cope with the fact that the supervision will be remote, we have set up a `Slack` channel for direct communication and a shared `GitHub` repository for the code. Moreover, the integrated Wiki in the GitHub repository will be used as a central documentation point not only for the code itself but for the project in general (project description, references, work plan, etc.). The language to perform all tasks is chosen Python, together with the `Tensorflow` and `Keras` machine learning frameworks and `Qibo` for quantum simulation.

Outlook

In future research, our results can be used to analyze more complicated features. The developed library can be used as a tool in experimental and theory studies to understand the characteristics of jet tagging algorithms. The combination of machine learning and quantum computing methods can potentially be used to classify data more efficiently, than classical machine learning methods.

References

[1] F. A. Dreyer and H. Qu, JHEP **03** (2021), 052 doi:10.1007/JHEP03(2021)052 [arXiv:2012.08526 [hep-ph]].

[2] F. A. Dreyer, R. Grabarczyk and P. F. Monni, [arXiv:2203.06210 [hep-ph]].

[3] G. Kasieczka, T. Plehn, M. Russell and T. Schell, JHEP **05** (2017), 006 doi:10.1007/JHEP05(2017)006 [arXiv:1701.08784 [hep-ph]].

[4] G. Kasieczka, T. Plehn, A. Butter, K. Cranmer, D. Debnath, B. M. Dillon, M. Fairbairn, D. A. Faroughy, W. Fedorko and C. Gay, *et al.* SciPost Phys. **7** (2019), 014 doi:10.21468/SciPostPhys.7.1.014 [arXiv:1902.09914 [hep-ph]].

[5] A. Pérez-Salinas, J. Cruz-Martinez, A. A. Alhajri and S. Carrazza, Phys. Rev. D **103** (2021) no.3, 034027 doi:10.1103/PhysRevD.103.034027 [arXiv:2011.13934 [hep-ph]].

[6] C. Bravo-Prieto, J. Baglio, M. Cè, A. Francis, D. M. Grabowska and S. Carrazza, [arXiv:2110.06933 [quant-ph]].

[7] S. Efthymiou, S. Ramos-Calderer, C. Bravo-Prieto, A. Pérez-Salinas, D. García-Martín, A. Garcia-Saez, J. I. Latorre and S. Carrazza, Quantum Sci. Technol. **7** (2022) no.1, 015018 doi:10.1088/2058-9565/ac39f5 [arXiv:2009.01845 [quant-ph]].