Deep-compression for ATLAS data

HSF GSoC'20 Project Proposal

Name and contact information

- **Name**: Honey Gupta
- **Email**: hn.gpt1@gmail.com
- **GitHub**: https://github.com/honeygupta/
- **Address**: 236, Tunga Hostel, IIT Madras, Chennai, India.
- **Mobile**: +91 942 244 6270

Motivation

Storage is one of the main limiting factors to the recording of information from proton-proton collision events at the Large Hadron Collider, at CERN in Geneva. There are approximately 1.7 billion events occurring inside the ATLAS detector, each second. The current technological limitations make it impossible to process and store the enormous amount of data that gets produced, even in the cases where that data is interesting.

Hence, the ATLAS experiment at the LHC uses a trigger system, which selects and sends interesting events to the data storage system while throwing away the rest. However, if interesting events are buried in very large backgrounds and difficult to identify as a signal by the trigger system, they will also be discarded together with the background. Therefore, an improvement in the compression factor by using a better compression algorithm can allow us to store more data and perform searches with increased sensitivity. This can help in detecting weaker signals and hence, can provide deeper insights.

Summary of the proposed project

The traditional autoencoders for data compression contain stacked fully-connected layers and the optimization is performed over the loss function to learn the parameters. However, data generated in HEP is organized into events and the existing compression algorithms [1] use a tree-like data-structure to represent an ordered collection of multiple events. The studies performed so far on using deep-compression for ATLAS data [2,3] have provided sufficient evidence that deep-compression is a viable option, but these models work on individual jets rather than full-events. Hence, the existing autoencoder networks might perform sub-optimally as they might not encapsulate the inbuilt structure of the data. Moreover, the biggest obstacle in using deep-compression for the ATLAS data is to run an autoencoder network in a resource-constrained environment such as an ATLAS trigger-machine. Running a deep-network on such systems might require significant changes to the model, which includes reducing the run-time and memory footprint while preserving the network performance. Also, running a deep-network in a trigger system can have its own challenges as it has not been tried so far.

In this project, I plan to first understand the ATLAS data, the ATLAS compression pipeline and run an autoencoder network on a trigger system to understand the limitations and time and memory constraints. I then plan to design a compression algorithm that works with full events and has a better compression factor, with adequate execution times and memory requirements in order to be implemented on a trigger system.

This project will consist of three stages:

- Run and benchmark the existing deep-compression algorithms or networks to compress ATLAS data in the context of a resource-constrained system such as the ATLAS trigger. Reconstruction error, execution time and memory consumption can be used as metrics to evaluate the algorithm's performance.
- Design an auto-encoder network to work with full events rather than jets and benchmark on real and simulated data (if required). Run the same on the trigger system and analyze its performance.
- (Optional) Extend the autoencoder network to work with sequential data i.e. a series of events and analyze its performance.

Deliverables

- Python scripts to compress and decompress the data using different compression algorithms analyzed during the project
- Scripts to produce the plots for different metrics
- Summary of the algorithms, documentation of the findings and points for future development.

Proposed Timeline

- Before May 4, 2020
  - Familiarise myself with existing ATLAS code, understand ROOT I/O and its functionalities.
- May 4 - May 17, 2020
  - Read up on ATLAS trigger and data formats, evaluate the existing compression algorithms, with hands-on experience if possible.
- May 18 - May 31, 2020

    - Learn about the critical factors concerning the environment, the planned experimental upgrades in 2026 and which factors regarding the ATLAS data compression could be relevant while designing the deep-compression algorithm.
    - Document and discuss the possible directions.
- June 1 - June 28, 2020
    - Experiment and analyze the existing deep-compression algorithms or networks to compress the ATLAS data in the context of a resource-constrained system such as the ATLAS trigger.
    - Document the findings and their performances for the 1st evaluation.
    - (Optional) Perform hyperparameter tuning for a selected set of architectures. This is time-consuming and hence, will be performed if time permits.

June 29 - July 3, 2020 (1st evaluation)

- July 4 - July 19, 2020
    - Explore autoencoder architectures and develop a network that compresses full events rather than individual jets.
- July 20 - July 26, 2020
    - Document the results and conclusions obtained from the experiments for the 2nd evaluation.

July 27 - 31, 2020 (2nd evaluation)

- Aug 1, Aug 15, 2020
    - Develop a deep-compression network that works on entire jets and can run in resource-constrained systems such as the trigger system, based on the previous months' findings. Analyze its performance.
    - (Optional) Explore the possibility of anomaly detection using the designed autoencoder-network.
- Aug 16 - Aug 23, 2020
    - Document the findings and write a white paper if needed.
- Aug 24 - Aug 31, 2020
    - Code submission and final evaluations.

Other commitments during the GSoC period

From the start of May until the end of August, I am completely available to work on the project. The GSoC timeline also aligns with my timeline and according to the proposed timeline mentioned above, I plan to complete the project during these 4 months.

Links to the evaluation assignment submissions

- **Code:** https://github.com/honeygupta/CompressionHEP
- **Presentation:** Deep-compression for HEP data

References
[1] Zhang, Z. and Bockelman, B., 2017. Exploring compression techniques for ROOT IO. arXiv preprint arXiv:1704.06976.
[2] Wulff, E., 2020. Deep Autoencoders for Compression in High Energy Physics.
[3] https://github.com/Skelpdar/HEPAutoencoders