Jet Flavor Tagging in ALICE3 – a Summer Student Project Report

Szymon Slawinski
Under supervision of:
Nima Zardoshti & Luigi Dello Stritto

September 2024

1 Jets

Jets are high p_{T} objects composed of many particles created in hadronization of a parton. The structure of a jet is dependent on the mass and the color charge of the initiating parton. The inability to identify the type of the initial parton greatly contributes to the uncertainties in jet measurements and stands in the way of studying the mass and color charge effects.

Heavy-flavor jets, originating from charm and beauty quarks, are especially interesting, because of the jet tagging techniques that allow us determining the type of the initial quark and creating jet samples enriched with a single flavor. This capability is crucial for studying mass effects in the low-p_{T} regime, which is accessible by the ALICE experiment, and plays a key role in investigating energy loss mechanisms in the quark-gluon plasma (QGP) through the use of pure heavy-flavor jet samples."

1.1 Tagging heavy-flavor jets

Current flavor tagging methods used at ALICE require a full reconstruction of a heavy-flavour hadron. This is good for studying the substructure of jets, but reduces the number of jets that can be flavor tagged. The heavy hadrons can be reconstructed only through some decay channels, so the efficiency of this method is limited by the branching ratio of the channels that are suitable for reconstruction.

A more efficient way of flavor tagging would be to use the tracks and secondary vertices (SV) belonging to a given jet, without a full reconstruction of the hadron. Because of their relatively long lifetime, heavy-flavor hadrons travel a measurable distance before decaying. The decay length and the impact parameter (Figure 2) can be used to infer the presence of a heavy-flavor hadron in a jet, and to tag it as a heavy-flavor jet.

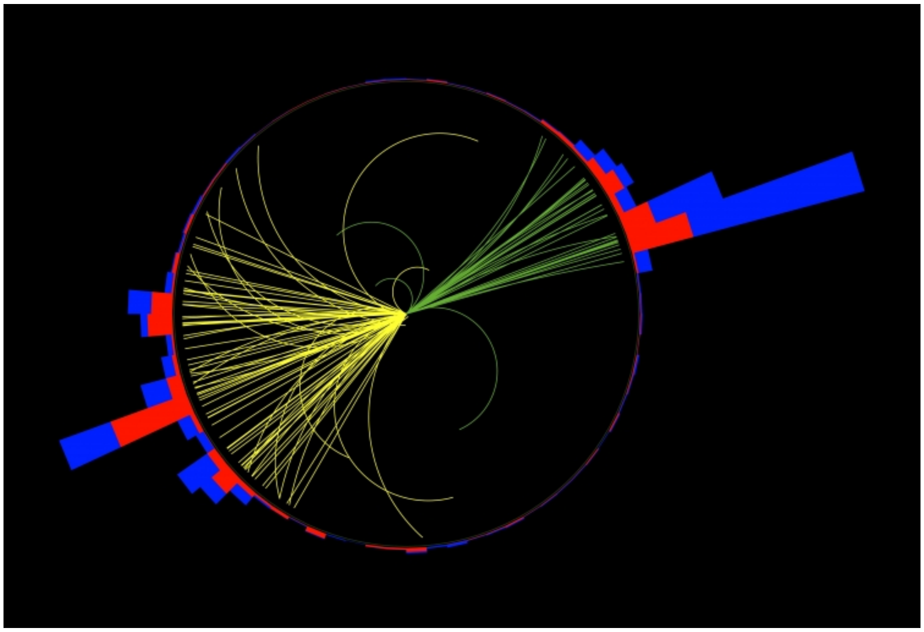

Figure 1: A visualization of a 2 jet event. The tracks belonging to two different jets are marked with different colors [1].

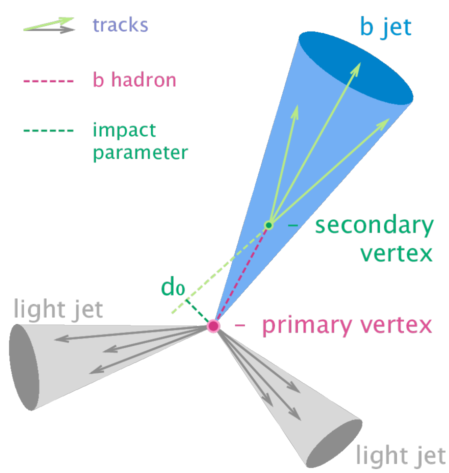

Figure 2: Diagram showing the impact parameter d_0 and the secondary vertex [2].

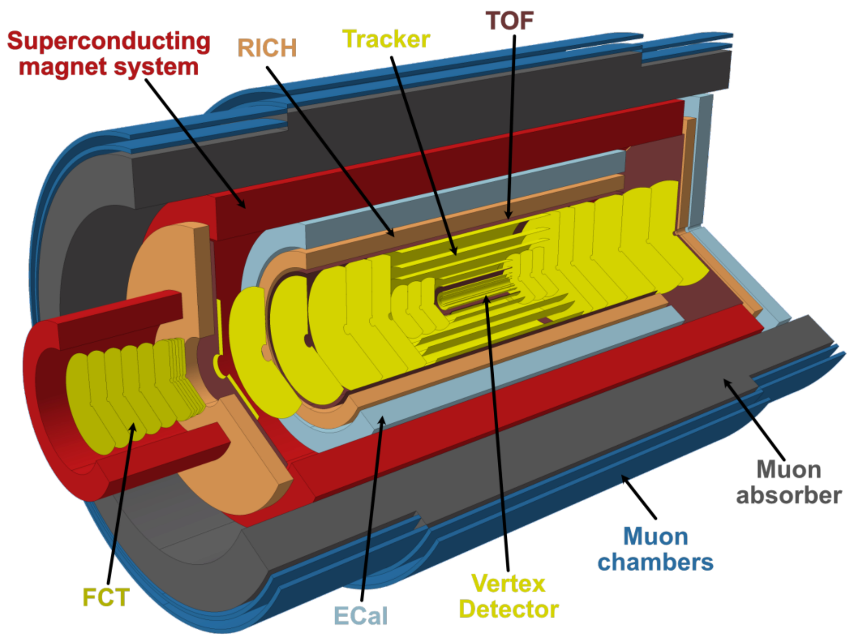

Figure 3: The rendering of a future ALICE3 detector.

2 ALICE 3

ALICE 3 [3] (Figure 3) is a future detector that is going to replace the current ALICE detector after Run 4. Despite the similar name, the new instrument will bear little resemblance to its predecessor and it will be designed and built completely from scratch. Some of the key improvements ALICE 3 will guarantee are:

- Excellent pointing resolution thanks to a tracker placed very close to the interaction point.
- Very large pseudorapidity range $|\eta| < 4$ (compared with $|\eta| < 0.8$ of the current detector).
- 2 T solenoid with dipole magnets on both ends (compared to 0.5 T and a single dipole of the current detector).

2.1 ALICE 3 Configurations

The goal of the project was to test the flavor-tagging performance for different configurations of the ALICE 3 detector. The 4 tested configurations were:

- 1 T solenoid with no end-cap dipoles

- 1 T solenoid with end-cap dipoles
- 2 T solenoid with no end-cap dipoles
- **2 T solenoid with end-cap dipoles** (nominal configuration)

The plots shown in the later sections of this report were done for the nominal configuration. Performance was tested in two pseudorapidity regions: low η ($|\eta| < 2$) and high η ($2 < |\eta| < 4$).

3 Analysis

The goal of the analysis was the classification of jets into one of the 3 types (light, c or b) with a machine learning algorithm known as Boosted Decision Trees (BDT). In the following section the concept of BDT is introduced.

3.1 Boosted Decision Trees

Supervised learning is a class of machine learning (ML) algorithms in which a set of examples is provided to train the model. Each example contains a vector of features and a label. The goal of the ML model is to predict the label given a vector of features. To evaluate the performance of the model and avoid overfitting, where the model becomes too tailored to the training data, a separate testing set is used. This ensures that the model's ability to generalize to new, unseen data.

Decision trees are a ML algorithm named after their tree-like structure (Figure 4). They are composed of nodes, which contain conditions and branches which are connections between nodes. During learning, distributions of input variables are provided, and the conditions assigned to nodes are chosen in a way to separate the different classes as well as possible. Conditions have a form $v_i > x_i$, where v_i is the value of the i-th variable, and x_i is the value that maximizes the separation between different classes. The size of the tree (number of nodes) is controlled by the parameters of the model.

The algorithm begins at the root node, which contains the first condition, and ends at one of the leaf nodes, which contains the classification score. The classification score is a set of numbers and each number corresponds to the probability of the example belonging to one of the classes.

After the model is trained, the classification of an example goes as follows:

1. Start at the root node; check the condition at the root node and follow the appropriate branch to the next node
2. Keep checking the conditions and following the correct branches until you reach a leaf node
3. The leaf node contains the classification score

4. Use a cut on a classification score to obtain the class the example belongs to. This cut is usually chosen by the end user to obtain the desired efficiency and purity (equations 1 and 2 in Section 4)

Using a single decision tree would not provide good classification performance. A small tree will perform badly, because of its low complexity, while a big tree is prone to overfitting (the performance will be good only on a training set, but the model will fail on a testing set). A much better approach is to use Boosted Decision Trees (BDTs).

Boosting is based on producing many small decision trees and combining their classification scores. To create the ensemble of trees for boosting, the training set and the input variable set are subsampled (otherwise all the trees would be the same). BDTs significantly outperform individual decision trees by providing improved classification accuracy and greater resistance to overfitting.

Parameters of the BDT algorithm control the complexity of the model. Tuning those parameters properly is necessary to achieve a good performance and avoid overfitting. Some chosen parameters are described below:

- Tree depth: the maximal number of nodes between the root node and any of the leaf nodes
- Number of trees used for boosting
- Training set subsampling size: the fraction of the training set to randomly sample for training (sampling is repeated for every tree)
- Feature subsampling size: a fraction of features to randomly sample for training (sampling is repeated for every tree)
- Lambda: an L2 regularization coefficient; regularization is a technique used to reduce model complexity
- Early stopping rounds: early stopping stops the learning process if the model doesn't improve after a number of trees controlled by this parameter are added to the model

3.2 MC simulations

The MC simulations, needed to train and test the model, were generated with a Monte Carlo event generator `Pythia8` [5], with hardQCD $b\bar{b}$ and $c\bar{c}$ processes switched on. The generated events were proton-proton collisions at center of mass energy of 13.6 TeV. The detector simulation was performed with a fast response parameterization of ALICE 3 to smear particles to tracks. The jets and the secondary vertices were reconstructed using a heavy-flavor tagging framework in `O2Physics`. The dataset consisted of around 180000 jets for each detector configuration and each η region (to ensure a fair comparison between different scenarios).

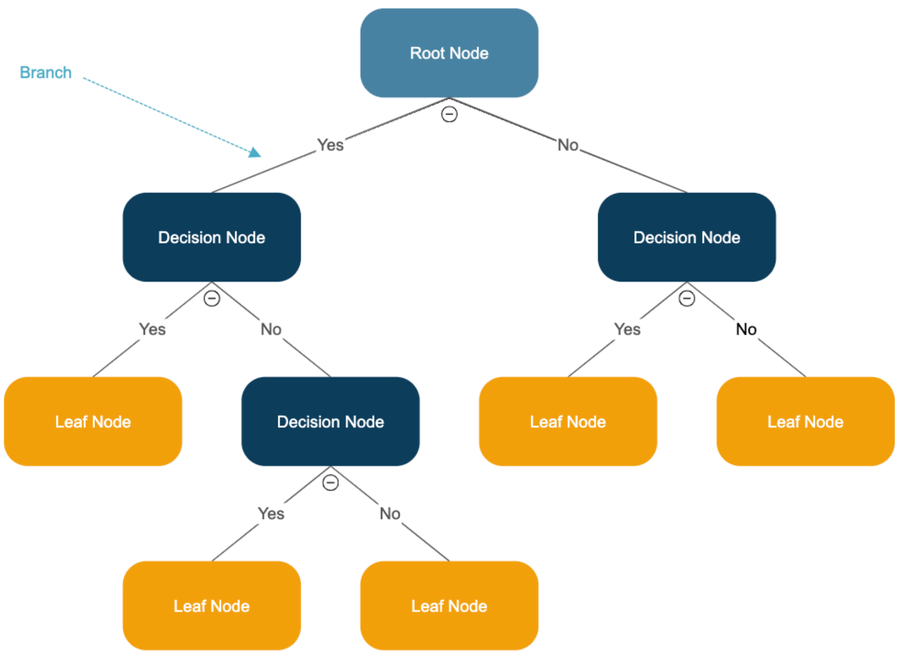

Figure 4: A schematic decision tree [4]

3.3 Training

The variables describing the jet, and the tracks and secondary vertices belonging to it were used to train the BDT model. It is important to note that different jets can contain different number of tracks and secondary vertices. This variability poses a problem for the machine learning model, which requires input data of fixed length. The issue was solved by padding: for each jet, only the 7 tracks with the largest impact parameter and 2 SV with the largest decay length were used. The surplus was ignored, while missing values were filled with a constant value (-999). All the variables were appended into a one dimensional vector to be fed to the model.

To avoid introducing bias into jet substructure studies, only variables that are unrelated to the jet substructure were carefully selected to train the model.

- Tracks:
  - Impact parameter
  - ΔR between the track and its closest SV
- SV:
  - Decay length
  - Cosine pointing angle

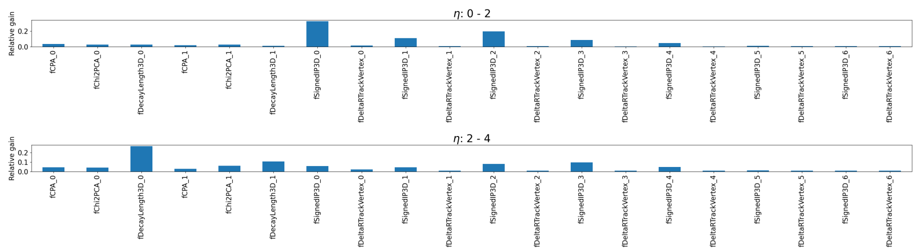

Figure 5: The feature importance for low and high η regions. The impact parameter is more important for low η, while for high η the decay length is more important.

- χ^2 of the SV fit

The 3 jet classes (light, c and b) were balanced to ensure equal representation in the training dataset. A transverse momentum cut $p_\mathrm{T} > 20$ GeV was applied. Half of the dataset was used for training, while the other half was used for testing.

The maximum number of trees used for boosting was set to 2000, the maximum depth of a tree was set to 4. Subsampling size was set to 0.7. L2 regularization and early stopping were used to limit overtraining.

Figure 5 shows the importance of the different features used for the training. It is interesting to note that the decay length becomes more important for high pseudorapidity. This phenomenon occurs because, in order to achieve high transverse momentum in the high η region, jets must have very high energy, resulting in a significant Lorentz boost. Although the $p_\mathrm{T} > 20$ GeV cut is applied uniformly, jets at large η need much higher energy to reach this p_T threshold compared to jets at smaller η. As a result, the decay length is larger in the high η region due to this increased energy and corresponding boost. This effect is evident in the decay length distribution shown in Figure 6.

Figure 7 shows the distributions of the model output split into classes. The model output consists of 3 numbers. Each of those numbers corresponds to the probability that the given jet belongs to one of the 3 classes. The better the separation of the output distributions, the better the model is. The fact that the histograms of the training set and the testing set are similar indicates that the model is not overtrained.

4 Results

Figure 8 shows the efficiency vs purity plots. Those metrics are defined as:

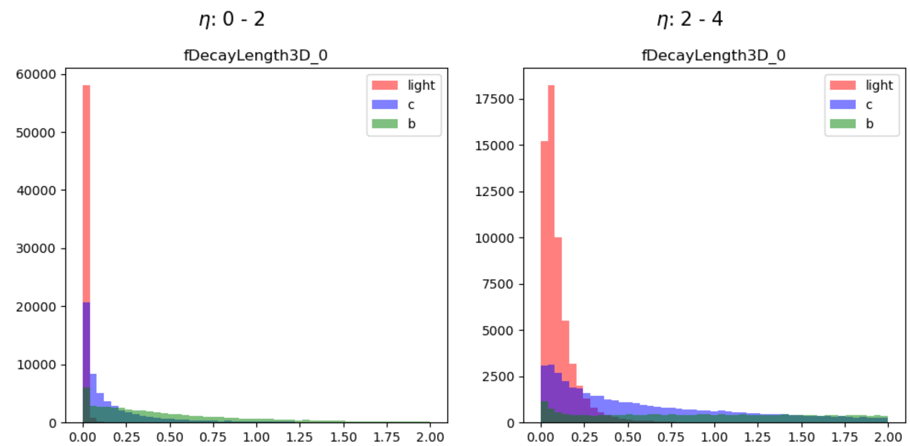

Figure 6: The decay length distribution divided into jet flavors, for two η regions. For high η the decay length is longer for every jet flavor.

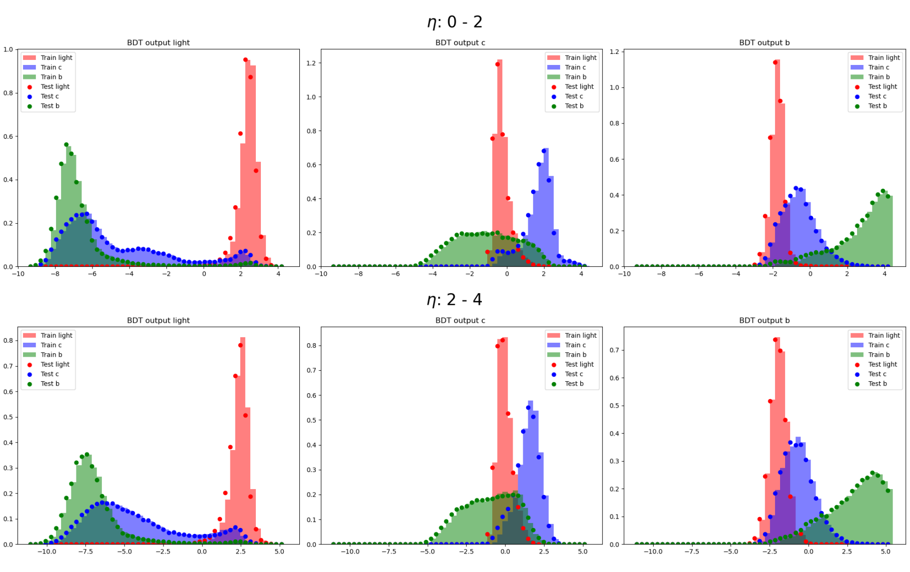

Figure 7: The model output for low and high η regions. Colored fields correspond to the training set; dots correspond to the testing set. Jet flavors are color-coded.

$$\text{Efficiency} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \tag{1}$$

$$\text{Purity} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \tag{2}$$

The performance is only slightly worse for the high η region. In both cases, efficiencies and purities around 90 % for b-tagging and around 85 % for c-tagging can be reached. Those excellent results are compared to the performance achieved by the current ALICE and ATLAS detectors in Figure 9. It is evident from this comparison that the jet-tagging performance expected with ALICE 3 significantly outperform that of the existing detectors.

4.1 Comparison of ALICE 3 Configurations

No difference in flavor tagging performance was observed when comparing the different ALICE 3 configurations listed in section 2.1. This is because the differences between configurations pertained primarily to the detector magnets, which most significantly affect the p_{T} resolution. Since, p_{T} is not utilized in flavor tagging, the improvement in p_{T} resolution did not impact the flavor-tagging performance. Different configurations of the tracker could instead potentially affect the results, as the tracker influences the reconstruction of secondary vertices and impact parameters, which are crucial for flavor tagging.

Figure 10 shows a p_{T} resolution histogram for different configurations of ALICE 3 magnets. As expected, p_{T} resolution improves when dipole magnets are added. Surprisingly, there is no improvement when the magnetic field of the solenoid is increased from 1 T to 2 T. The reason for that is not clear, and should be investigated.

5 Summary

The project has been successfully completed. A Boosted Decision Trees model has been trained and tested on different ALICE 3 configurations. The performance of heavy flavor jet tagging in ALICE 3 has been shown to be excellent, reaching 90% of efficiency and purity for b-tagging and 85% for c-tagging. Different magnet setups have been shown to not affect jet tagging performance, as expected. The jet p_{T} resolution has improved when dipole magnets were introduced, but did not change when the magnetic field of the solenoid changed, which is unexpected.

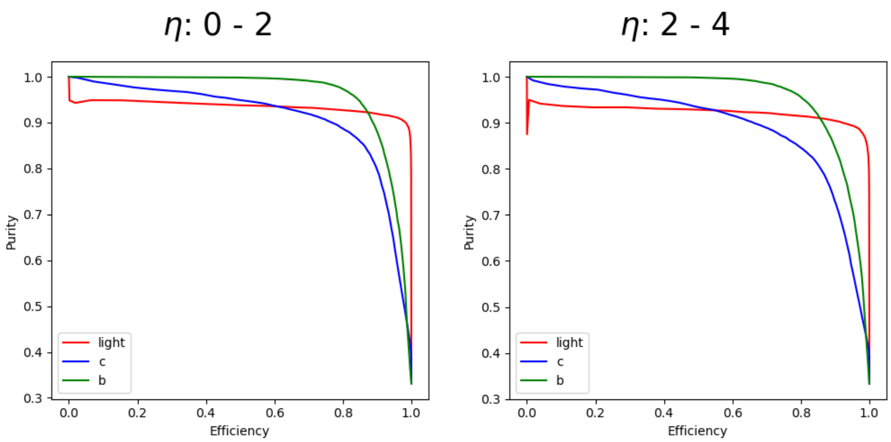

Figure 8: Efficiency vs purity plots done on a testing set, for low and high η regions. Jet flavors are color coded.

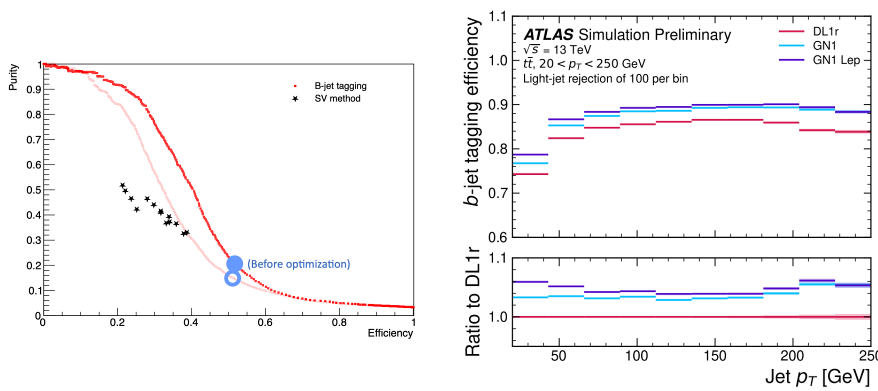

Figure 9: Tagging performance of ALICE [6] (left) and ATLAS [7] (right)

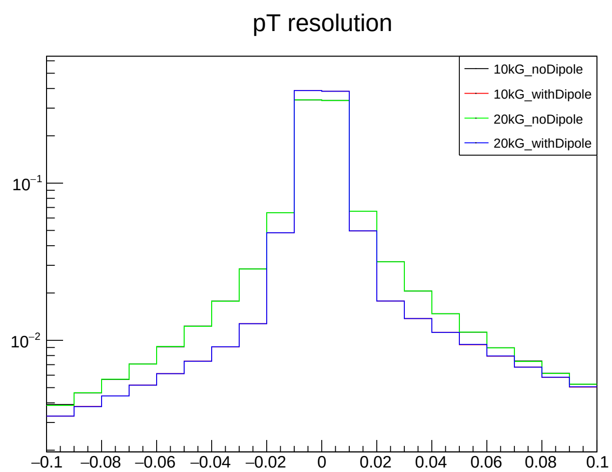

Figure 10: p_{T} resolution histogram for different configurations of the ALICE3 detector.

References

[1] URL: `https://cms.cern/news/machining-jets` (visited on 09/06/2024).

[2] Nazar Bartosik. URL: `https://commons.wikimedia.org/wiki/File:B-tagging_diagram.png` (visited on 09/06/2024).

[3] *Letter of intent for ALICE 3: A next generation heavy-ion experiment at the LHC*. Tech. rep. 202 pages, 103 captioned figures, 19 tables. Geneva: CERN, 2022. arXiv: `2211.02491`. URL: `https://cds.cern.ch/record/2803563`.

[4] URL: `https://www.smartdraw.com/decision-tree/` (visited on 09/06/2024).

[5] Christian Bierlich et al. "A comprehensive guide to the physics and usage of PYTHIA 8.3". In: *SciPost Physics Codebases* (2022), p. 008.

[6] Changhwan Choi. URL: `https://indico.cern.ch/event/1424090/contributions/6049091/` (visited on 09/06/2024).

[7] *Graph Neural Network Jet Flavour Tagging with the ATLAS Detector*. Tech. rep. Geneva: CERN, 2022. URL: `https://cds.cern.ch/record/2811135`.