University of Oxford

Lincoln College

Doctorate of Philosophy

Particle Physics

Advanced Machine Learning Applications for the Higgs and Heavy Flavour Quarks at ATLAS

Candidate

Maxence DRAGUET

Supervisor

Daniela BORTOLETTO

2020-2024

Contents

List of Abbreviations

μP Maximal Update Parametrisation
AI Artificial Intelligence
AUC Area Under the Curve
BSM Beyond the Standar Model
CERN Centre Européen pour la Recherche Nucléaire
CKM Cabibb-Kobayashi-Maskawa
CPU Core Processing Unit
DIPS Deep Impact Parameter Sets
DL Deep Learning
DL1 Deep Learner 1 Model
DL1d Deep Learner 1 Model with DIPS
DL1r Deep Learner 1 Model with RNNIP
DNN Deep Neural Network
FTAG Flavour Tagging Group
GAT Graph Attention Network
GN1 Graph Network 1 Model
GN2 Graph Network 2 Model
GNN Graph Neural Network
GPU Graphics Processing Unit
HPC High Performance Cluster
HPO Hyperparameter Optimisation
IBL Insertable B-Layer
IP Impact Parameter
ITk Inner Tracker
JVT Jet Vertex Tagger
LHC Large Hadron Collider
LSTM Long-Short Term Memory
MC Monte Carlo
ML Machine Learning
NN Neural Network
PU Pile-up
PV Primary Vertex
QCD Quantum Chromodynamics
ReLU Rectified Linear Units
RNN Recurrent Neural Network
RNNIP Recurrent Neural Network Impact Parameter
ROC Receiver Operating Characteristic
SCT Semiconductor Tracker
SF Scale Factors
SGD Stochastic Gradient Descent
SM Standard Model
SV Secondary Vertex
SV1 Secondary Vertex method 1
UFO Unified Flow Object
VR Variable Radius
WP Working Point

CHAPTER 1

Flavour Tagging

The focus of this chapter is on an essential task in the ATLAS experiment: identifying particles flying through the detector. This objective of assigning labels to reconstructed particles from measurements is referred to as tagging. An important family of particules to be tagged are quarks, and disentangling which specific flavour of the quarks should be assigned to an observed signal is called flavour tagging. Free quarks and gluons hadronise as per the rules of Quantum Chromodynamics (QCD), forming large number of particles that can themselves further decay. Such a dynamic results in a many particles radiating within a clone centred around the inital flavoured particle, a structure referred to as a jet. Consequently, this chapter introduces method to tag jets, as labelled by the flavour of the initial parton. In particular, the different algortihms and methods relevant for this task developped during the during of the author's contribution to ATLAS are reviewed, with the first trainings of DIPS, DL1d, GN1, and GN2 described in details as well as early studies of the hyperparameter optimisation of GN2.

1.1 Heavy-Flavour Jet Tagging

A fundamental ingredient in any ATLAS analysis is the ability to correctly identify particles in the aftermath of a collision, from τ-leptons, to b- and c-quarks, and gluons g. Having well-

calibrated and optimally performing b- and c-tagging tools is of primary importance in studies of the Higgs boson couplings to b- and c-quarks. It is also critical for top t-quark measurements and searches for extensions of the Standard Model (SM). As described by the theory of Quantum Chromodynamics (QCD), colour-charged objects, such as a b- or a c-quark, undergo hadronisation to form collections of colourless hadrons. These hadrons, mostly B for b-quark and D for c-quark, are quasi-stable and further decay in the volume of the detector. Such a succession of decays leaves a collection of particles within a cone oriented in the direction of the original parton, an easily recognisable pattern referred to as a *jet*. From an analysis of the complicated structure of the jet, the flavour of the initially decaying particle can be reconstructed. The labelling scheme applied to jet consist in identifying the hadron associated to the jet: a b-jet must contain at least one b-hadron, a c-jet at least one D-hadron and no b-hadron, and if none of these hadrons are found the jet is said to be a light-jet, thereby grouping u-, d-, and s-quarks with gluons g. This is the task of *flavour tagging*, and the tool to achieve this identification is called a *tagger*.

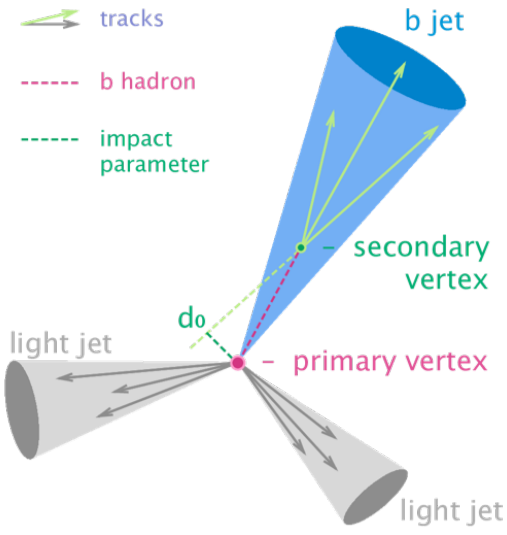
1.1.1 Decay Topology

When a b-quark is produced, such as in the aftermath of a hard scatter due to a proton-proton collision, it quickly undergoes the process of hadronisation to neutralise its free colour-charge. This process leading free quarks and gluons to a final state of hadrons and leptons is intrinsically non-perturbative and can only be described with phenomenological models of framgentation [1]. The family of b-hadrons is composed of different ensemble of a bottom quark b with one or more light quarks. These include the B-mesons, mainly $B^0 = d\bar{b}$, $B^- = \bar{u}b$, $B^+ = u\bar{b}$, and the strange and charmed B-mesons, and baryons, such as the $\Lambda_b^0 = udb$ [2]. For b-quarks, the hadronisation process is hard and most ($\sim$ 70-80%) of the quark momentum is passed to the b-hadron [1]. Tagging b-jets benefits from a particularly advantageous configuration: the b is the lightest element of the third generation of quarks and must decay through a flavour-changing process through the weak interaction. Because of the relatively small SM value of the V_{bc} Cabibb-Kobayashi-Maskawa (CKM) matrix element, decay processes involving a transition $b \to c$ are suppressed, giving b-hadrons a characteristically long proper lifetime $\tau_B \approx 1.5$ ps corresponding to a proper decay length $c\tau_B \sim 450$ μm [3]. In the laboratory frame and considering the boost of the b-hadron given by a Lorentz γ factor ($\gamma > 1$) and $\beta = v/c \approx 1$ in the high energy limit,

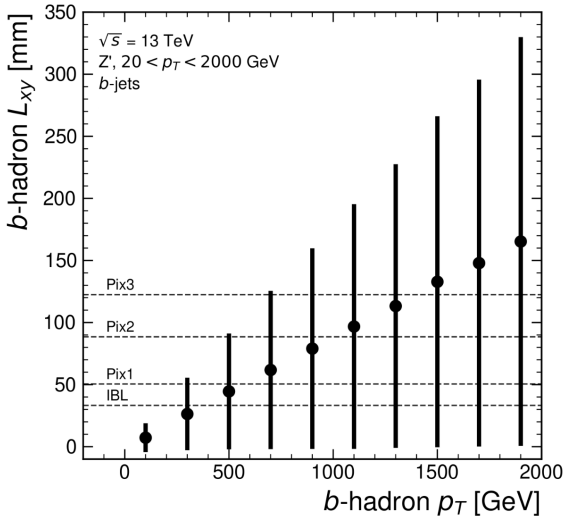
the distance travelled is given by

$$d = \gamma\beta c\tau_B \approx \gamma c\tau_B.$$

In the high-energy limit, $\gamma \approx E_B/m_B$, where the mass of B-hadrons is in the range 5-6 GeV. Consequently, a 50 GeV b-hadron decays at a distance of $d \approx 4.5$ mm from the primary vertex which can be resolved. This distance travelled, also called the b-hadron radius L_{xy}, further increases with jet p_T and at transverse momentum above ~500 GeV even overpasses the first detector layer of the Insertable B-Layer (IBL), located at a distance of roughly 33 mm from the centre of ATLAS as shown in Figure 1.1b. The location of the b-hadron decay can often be reconstructed by the ATLAS detector, and the reconstructed point of decay is called Secondary Vertex (SV) [4]. Some important variables describing the decay of b-hadrons are the Impact Parameters (IPs) d_0 and z_0 of the charged particles emanating from the SV. As shown in Figure 1.1a, d_0 and z_0 are the transversal and longitudinal distances from the primary vertex to the perigee[1] of the track associated with the charged particle. For a b-jet, the IPs can be large thanks to the long lifetime of the associated hadron. On average, a b-hadron decays weakly to four or five charged stable particles [2]. Another characteristic of b-jets is the likely presence of leptons in the jet cone, as 40% of the decays of b and c-hadrons are semi-leptonic and include either an e or a μ [3].



(a) Representation of a b-jet [5].



(b) b-hadron decay radius as a function of jet p_T reconstructed for b-jets in a Z' events with the IBL and pixel layers indicated, from [6]. Error bars show the standard deviation of L_{xy} in each p_T bin.

While b-jets benefit from an advantageous topology, tagging c-jets proves much more challenging as they are at an intermediate scale between light- (u, d, s, and gluons) and heavy flavour jets. A c-jet must contain at least one c-hadron, from either a D-meson (e.g., $D^+ = c\bar{d}$, $D^- = d\bar{c}$,

[1] The point of closest approach.

$D^0 = c\bar{u}$) or a c-baryon (e.g., $\Lambda_c^+ : udc$). The average decay length for charged (neutral) D-mesons, $c\tau_D \sim 300$ (100) μm [3], is smaller than for b-hadrons and is harder to resolve with the currently deployed tracker. The decay chain of b-hadrons often includes a c-hadron, making a clean separation of c-jets from b-jets harder. Compared to b-jets, c-jets have a lower final state charged particle multiplicity, on average 4. This lets τ-jets easily being mistaken for c-jets, as these leptons can hadronically decay into a similar number of particles to mimic a jet in the detector. For all these reasons, less effort has been historically dedicated to constructing c-taggers in ATLAS. The task is however gaining particular attention due to the focus on the challenging $H \to c\bar{c}$ search [7, 8, 9]. The focus of this chapter is on the development of novel taggers to identify b- and c-jet for the ATLAS experiment during the 2020-2024 period, overlaping with the end of Run 2 (2015-2018) analyses and the begining of Run 3 (2022-2026).

1.1.2 Flavour Tagging at ATLAS

In the ATLAS experiment, a choice was made to develop centrally a tagger to be used by the whole collaboration. It relies on a dedicated set of algorithms to perform simultaneously b- and c-tagging and is continuously improved to meet the requirements of the physics program. Currently, all adopted approaches rely on Deep Learning (DL) methods, given their vastly superior effectiveness. This area of research has been evolving rapidly in recent times due to the community adopting advanced methods from the field of Artificial Intelligence (AI). As such, various models have been introduced and the last two generations can be split into two categories:

1. The DL1 family are DL models built in a hierarchical way. These DL methods rely on high-level features reconstructed by sub-algorithms based on physics variables, such as the tracks IPs, and the reconstruction of secondary vertices [10]. The most important models in this family are those including a DL sub-model to analyse tracks with either a Recurrent Neural Network (RNN) approach for Deep Learner 1 Model with RNNIP (DL1r) [11], leveraging the Recurrent Neural Network Impact Parameter (RNNIP) sub-tagger [12], and a Deep Set approaches for Deep Learner 1 Model with DIPS (DL1d), leveraging the Deep Impact Parameter Sets (DIPS) sub-tagger [13]. This last tagger is, at the moment of writing this thesis, the state-of-the-art deployable tagger for ATLAS in the ATLAS software [14]. Algorithms from this family were developped for Run 2 of the ATLAS experiment [15], with DL1d behind developped towards the end of this data campaign.

2. The GN family of taggers are built on more advanced DL method, as they move away from the hierarchical approach of the DL1 family and directly analyse track and jet information

in a unique complex architecture. The GN family is based either on a full Graph Attention Network (Graph Attention Network (GAT)) for Graph Network 1 Model (GN1), and a Transformer encoder for Graph Network 2 Model (GN2) [16, 17, 18]. This lighter algorithm pipeline greatly simplifies the maintenance and turnaround time for modification, making the process of updating the taggers nimbler and easier to tailor to specific applications. The GN taggers greatly outperform the DL1 family and represent an exciting area of progress for future analysis requiring precise flavour jets tagging. These methods are being integrated in the ATLAS software stack with objective to be integrated in analyses of the ongoing Run 3 [14].

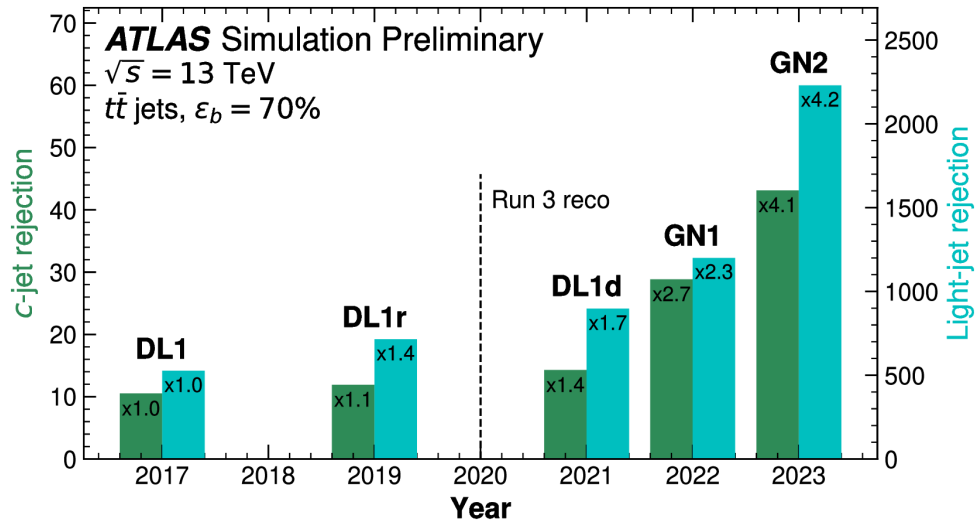


Figure 1.2: Comparison of the performance of flavour tagging models introduced through the years, from [17]. Light and c-jet rejections are plotted for different taggers at a fixed b-jet tagging efficiency of 70% on a common $t\bar{t}$ evaluation sets. The multiplicative factors in the bars are with respect to the bare DL1 model performance.

A historical perspective on the evolution of performance for the different taggers mentioned is presented in Figure 1.2, showing a remarkable continuous increase in light- and c-jet rejections at a fixed b-tagging efficiency of 70% on a $t\bar{t}$ dataset. The analysis presented in the latter part of this thesis was carried out in the span of 2021-2024, and was therefore restricted to tools and methods available to the experimental team during this period. As such, due to the need to calibrate the GN taggers as explained latter in Section 1.4 of this chapter, the second family of taggers based on graphs was not yet ready for deployement, making the first family still relevant to explore. Furthemore, some specialised use of flavour taggers and early Run 3 analyses still required the DL1 family, such as the X_{bb} tagger identifying pairs of $b\bar{b}$ and $c\bar{c}$ produced by heavy resonance decays such as a Higgs or a W, as described in Section 1.3.4. All the taggers described in this chapter have been integrated in the ATLAS software [14]. Some early studies of the future

performance of the DL1d and GN1 taggers with the updated Inner Tracker (ITk) detector at the high-luminosity Large Hadron Collider (LHC) as been performed in Ref [19].

1.1.3 Datasets

The p_T spectrum studied by ATLAS covers a wide range of energies, extending up to 7 TeV. In order to train models able to perform on this large phase space, two datasets are typically combined and described in this section. Each dataset is based from simulated proton-proton collisions at a centre of mass energy $\sqrt{s} = 13$ TeV. The lower p_T phase space is simulated with the SM top-antitop quark pair production $t\bar{t}$ process with at least one W boson produced decaying leptonically, while a Beyond the Standar Model (BSM) Z' process is used for the higher momentum region. The latter simulates a modified Z boson from the SM with an increased mass to generate a relatively flat jet p_T spectrum of up to 6 TeV. This Z' boson decays in similar proportions to a pair of b, c, and light-jets. All simulations include realistic effects present in the real data, such as multiple proton-proton interactions per bunch crossing, so-called Pile-up (PU), with an average value of $\mu = 40$. Other effects including in the simulations are the detector response from prior and posterior bunch crossing (in-time pile-up) as well the activity from the rest of the event (underlying event).

Events in the $t\bar{t}$ samples are simulated using POWHEGBOX V2 generators to next-to-leading (NLO) order in the strong coupling constant α_s [20, 21, 22, 23]. The hard scatter matrix element is computed for proton-proton collision with the NNPDF3.0NNLO set of parton distribution functions (PDF) [24], and the simulated hard scatter events are interfaced with PYTHIA 8.230 [25] using the A14 parameter tune [26] and the NNPDF2.3LO PDFs for the parton shower, hadronisation, and underlying event simulations [27]. Studies in Refs [28, 29] showed these to be the optimal choice to correctly model the top quark p_T and the number of additional jets in the event, with the h_{damp} parameter set at 1.5 the mass of the top-quark $m_{\text{top}} = 172.5$ GeV. The Z' events are fully simualted with PYTHIA 8.212, A14 tune and the NNPDF2.3LO PDFs. The decays of b- and c-hadrons are performed by EvtGEN v1.6.0 [30].

The detector reconstruction effect of ATLAS and the modelling of the interaction between long-lived hadrons and the detector material are simulated with a dedicated software [31] built on GEANT4 [32]. Jets are selected in the phase space region defined by $|\eta| < 2.5$ and $p_T > 20$ GeV, with no overlapping with prompt generator-level e or μ from the W decay allowed. Pile-up

contamination is further reduced by an additional standard selection using the Jet Vertex Tagger (JVT) algorithm at a tight working point for jets with $p_T < 60$ GeV and $|\eta| < 2.4$ [33]. Tracks are associated to jets using a ΔR association cone of width decreasing with p_T ($\Delta R \approx 0.45$ at $p_T = 20$ GeV and $\Delta R \approx 0.25$ at $p_T > 200$ GeV). Tracks within the cones of several jets are associated to jet with minimal ΔR(track, jet). The label of the jet is inferred from the presence of a truth-level hadron within the cone ΔR(hadron, jet) < 0.3 centred around the jet axis.

1.2 DL1 Family of Models: DL1r & DL1d

This family of tagger is built with a hierarchical approach, combining low-level algorithms that are independentely optimised into a final Deep Neural Network (DNN) network of a few layers to output a final prediction. Not all low-level modules are based on DL, with some instead directly implementing physics-motivated algorithms. They consist of [34, 15]:

- IP likelihood: IP2D and IP3D are likelihood-ratio templates in 2D and 3D to assign flavour-discriminating weights based, respectively, on the transversal and global impact parameters significance (corresponding to the reweighted IP variables by their respective uncertainties) S_{d_0} (35 bins) and S_{z_0} (20 bins) of the tracks, and 14 bins of track catogarisation for IP3D [11]. For the three flavours, this results in $35 \times 20 \times 14 \times 3 = 29,400$ final bins, which each probability being computed per track. The likelihood assigned to the jet assumes the tracks are independent and is therefore calculated as the product of the per-track likelihoods. A discriminant is derived from the conditional log-likelihood, e.g., $D^b_{IP3D,f} = \sum_{i \in \text{tracks}} \log \frac{p^i_b}{p^i_f}$, with $f = c$ or light [10].

- Track collection analyser: either with RNNIP [12] or DIPS [13]. These are DL approaches to extract discrimination information on the set of tracks associated with a jet. These taggers are further described later in this chapter.

- Secondary Vertex method 1 (SV1): combining a secondary vertex finder and a tagger to offer flavour discrimination information [15]. The former, based on the VKalVrt vertex reconstruction package [35], returns a list of candidate secondary vertices with measured quantities assigned to each vertex. The latter derives jet weights based on discriminative variables and computes properties of the SV, such as the mass.

- Jet Fitter: a vertexing algorithm based on a Kalman filter to reconstruct the topology and fit the decay chain Primary Vertex (PV) $\rightarrow B \rightarrow D$ with the assumption that the vertices of the weakly decaying B/D-hadrons tend to align with the PV [15, 36].

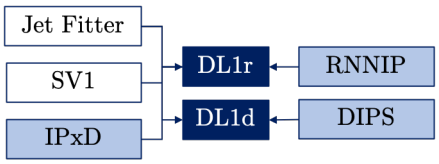


Figure 1.3: The algorithms for flavour tagging at ATLAS. High-level taggers are in dark blue, track-based taggers in light blue and vertex-related taggers in white.

The outputs of these low-level algorithms, as well as certain jet-related variables, such as p_T, are then combined as input to a high-level tagger consisting of a fully-connected Neural Network (NN) called DL1r or DL1d, respectively if RNNIP or DIPS is used. The input vector is typically made of 44-45 features. This high-level tagger outputs three probabilities p_X for the analysed jet to correspond to a b-, c-, or light–flavour (indicated with the letter u) such that $p_b + p_c + p_u = 1$. A b-tagging score D_b is then derived by computing a scaled log-likelihood ratio:

$$D_b = \log \frac{p_b}{f_c^b \times p_c + (1 - f_c^b) \times p_u}, \tag{1.1}$$

where f_c^b is the charm fraction, a parameter that can be modified to tweak the importance of each flavour. The analogous c-tagging score D_c is:

$$D_c = \log \frac{p_c}{f_b^c \times p_b + (1 - f_b^c) \times p_u}. \tag{1.2}$$

A jet is X-tagged if the D_X discriminant score is above a set threshold constant c_{wp}, defining a Working Point (WP) with a unique configuration of signal and background (mis-tag) efficiencies. In this context, the efficiency ϵ_Y^X for Y-flavour jets to be X-tagged and the corresponding rejection $\mathcal{R}_Y^X$ are respectively defined as:

$$\epsilon_Y^X = \frac{N_{Y-jets}^{X-tagged}}{N_{Y-jets}} \quad \text{and} \quad \mathcal{R}_Y^X = \frac{1}{\epsilon_Y^X}, \tag{1.3}$$

where $N_{Y-jets}^{X-tagged}$ and N_{Y-jets} are respectively the number of X-tagged Y-flavoured jets and the total number of Y-flavoured jets. The f-rejection is the inverse miss-tagged efficiency of flavour f.

These high-level models are trained on Monte Carlo (MC) simulated data samples, as mentioned in Section 1.1.3, and need to be calibrated on real data to deliver an unbiased estimate, by deriving Scale Factorss (SFs) weights correcting the predictions for each jet as described in Section 1.4. Uncertainties are derived on the predicted score and passed along to analyses using the tool. The novel algorithm of this family introduced in this work is the DL1d tagger, which relies on the DIPS sub-tagger to extract correlations between the tracks.

1.2.1 RNNIP

The RNNIP tagger runs on arbitrary-length input sequences made of track features, as ordered by the absolute transverse IP significance $|S_{d_0}|$, to extract tagging information from correlations between tracks [12]. The vector of track features, described in greater details in Table 1.1, includes the transverse and longitudinal impact parameter significances, the jet p_T fraction carried by the track, the distance between the track and the jet axis, and a learned 2D embedding of the track quality [34]. It outputs a probability p_X for the jet to belong to flavour $X \in [b, c, \text{light}, \tau]$.

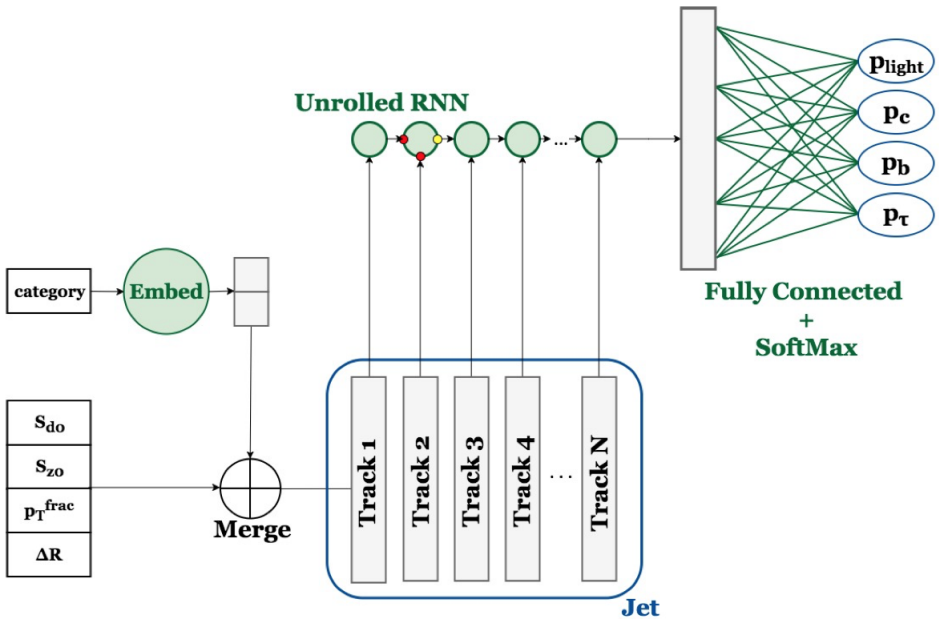


Figure 1.4: Diagram of the RNNIP tagger for flavour tagging, from [34]. The input consists in track features augmented with an embedding of track categories. Tracks are then ordered by absolute transverse IP significance and fed through an Long-Short Term Memory (LSTM) core. The unrolled sequence outputed from this LSTM is padded to a fixed size and processed by a DNN to output the per-flavour probabilities.

The architecture of RNNIP is a RNN-based model leveraging an LSTM core, as depicted in Figure 1.4. The arbitrary-length sequence fed as input is mapped by the LSTM cell with a 100-unit hidden layer into a 50-dimensional vector. This vector is then processed by a 20-unit fully-connected feed-forward neural network outputing the per-flavour probabilities by computing the softmax of the last layer's output. To avoid overfitting, a dropout value of 0.2 is applied to the LSTM cell.

RNNIP is designed to capture correlations between the tracks of a jet, a rich information explicitely missing from the IP-based discriminant of IP2D and IP3D due to the factorisation of the likelihood. Some degree of correlation is expected between tracks, as these can emerge from the same secondary or tertiary vertex of the displaced decay in b- and c-jets. It removes

Track Variables	Description
S_{d_0}	Lifetime signed transverse IP significance d_0/σ_{d_0}, with d_0 the transverse IP - the transverse distance from the PV to the point of closest approach (perigee) of the track - and σ_{d_0} the error on d_0. If the perigee is in front (behind) the PV with respect to the jet direction, the sign is positive (negative).
S_{z_0}	Lifetime signed longitudinal IP significance z_0/σ_{z_0}, with z_0 the longitudinal IP - the longitudinal distance from the PV to the perigee of the track - and σ_{z_0} the error on z_0. A sign is assigned as per the prescription of S_{d_0}.
p_T^{frac}	Fraction of the reconstructed jet p_T^{jet} carried by the track $p_T^{\text{frac}} = p_T^{\text{track}}/p_T^{\text{jet}}$.
ΔR(track, jet)	Geometrical distance in 2D angle between the track direction and jet axis $\Delta R = \sqrt{(\phi_{\text{track}} - \phi_{\text{jet}})^2 + (\eta_{\text{track}-\eta_{\text{jet}}})^2}$.
Category	2D representation of the track quality learnt by an embedding layer. The categorisation is based on the number of observed, expected and missing hits in the different layers of the tracker (silicon pixel and strip detectors) [10].

Table 1.1: Track variables passed to the initial version of the RNNIP model [12]. Later versions removed the category embedding and added the per-track hit information shown for DIPS in Table 1.2.

the cumbersome procedure to built likelihood templates, which demands large amount of data to scale to finer bin resolution and is computationally expensive due to the number of bins scaling exponentially with the number of variables. Early tests showed that RNNIP is effective at building a discriminant, delivering superior performances to the IP-based approaches with only ∼40 % of the parameters - 11,636 trainable parameters for RNNIP [34].

1.2.2 DIPS

The DIPS tagger, based on the Deep Set architecture [37] and depicted in Figure 1.5, is a Graph Neural Network (GNN)-based alternative approach to RNNIP to model the correlations between an arbitrary number of tracks [13]. As introduced in Chapter **??**, such a model is composed of two fully-connected feed-forward neural network. A first DNN called the *track network* Φ maps each individual track feature vector - similar to the input of RNNIP - to a latent space representing the nodes of a graph. The representations of each track in this latent space are then pooled by a simple summation operation - representing the unweighted edges of a fully connected graph - and given as input to a secondary DNN, called the *jet network* F, outputting the predicted probability p_X for the jet to belong to flavour $X \in [b, c, \text{light}, \tau]$. This last network represents the global attribute of the graph u, in the notation of Chapter **??**. In summary, DIPS computes the following equation on the set of track features $\{p_i\}$, with $i = 1, ..., N$ for arbitrarily-sized jets of N tracks:

$$DIPS(\{p_1, ..., p_N\}) = F\left(\sum_{i=1}^{N} \Phi(p_i)\right), \tag{1.4}$$

to output the per-flavour probabilities. The separation of computation into a per-track embedding and a per-jet processing after a size-indepent pooling performed by the sumation operator allows the model to process unordered sets of variable size. The track features used as inputs are described in Table 1.2, with only the top 15 tracks as ranked by decreasing S_{d_0} being kept.

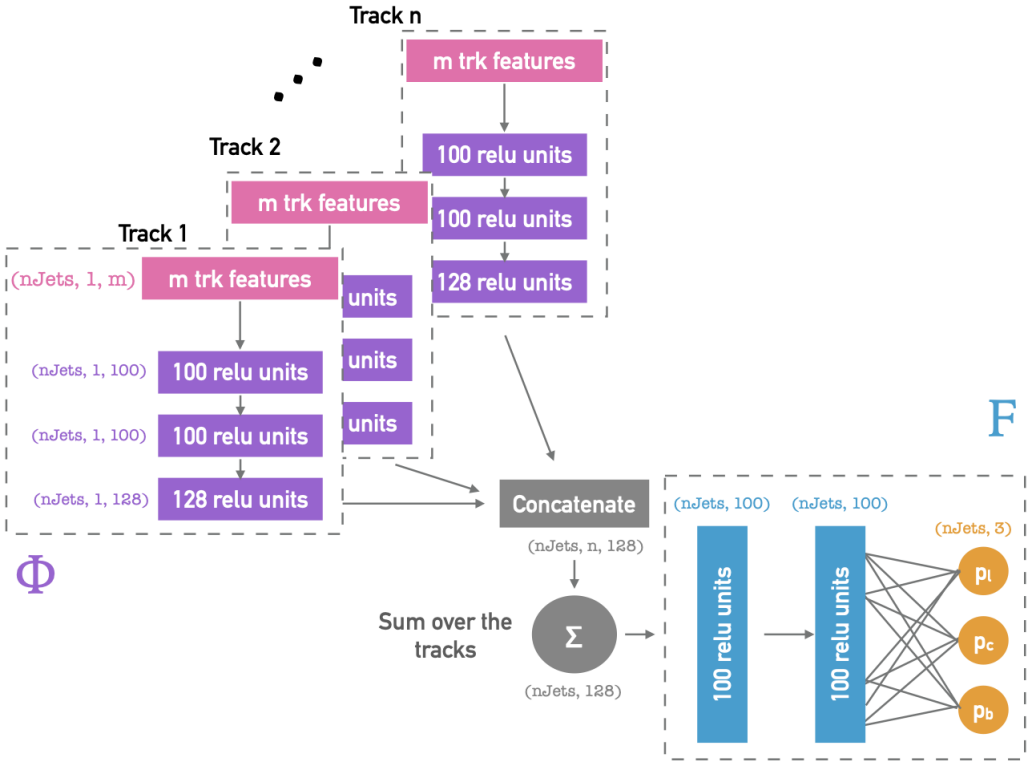


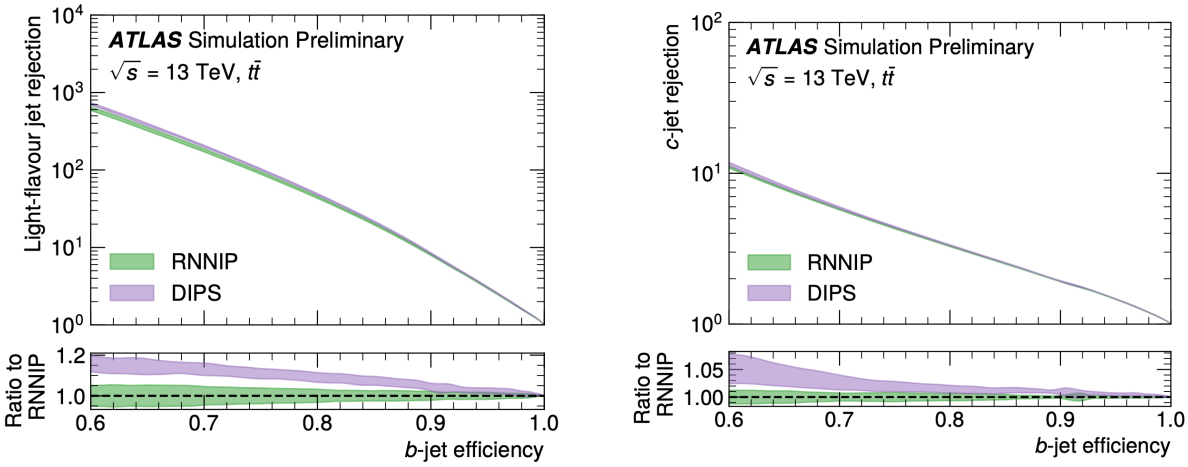
Figure 1.5: Diagram of the DIPS tagger for flavour tagging, from [13]. The input consists in a set of N tracks each represented by a feature vector. Each track is embedded by a DNN track network Φ into a fixed-dimension vector. All embedded track vectors are then pooled by summation to a fixed-size vector. The last step is to process this vector with another DNN jet network F outputing the per-flavour probabilities. The number and width of layers presented here are the nominal architecture.

This approach has several advantages over RNNIP, mainly the physically motivated permutation-invariance of the input and the improved training and evaluation time thanks to a more parallelisable architecture, as the track embedding performed by Φ can be massively parallalised on Graphics Processing Unit (GPU). These motivations are translated in an appreciable performance delivered by DIPS, which is observed to globally outpeform version of RNNIP using the same variables, while operating at a reduced computational cost [13]. The performance can be assessed from Figure 1.6, presenting the Receiver Operating Characteristic (ROC) curves for baselines trainings of DIPS and RNNIP in terms of light- and c-rejection for b-jet tagging on the same held-out $t\bar{t}$ evaluation sample.

The training times on the same GPU hardware for a 48k parameters DIPS model is estimated to take 78 ± 4 seconds per epoch - averaging over 5 seeds - while a 47k parameters RNNIP requires

Variables	Description
S_{d_0}	Lifetime signed transverse IP significance d_0/σ_{d_0}, with d_0 the transverse IP - the transverse distance from the PV to the point of closest approach (perigee) of the track - and σ_{d_0} the error on d_0. If the perigee is in front (behind) the PV with respect to the jet direction, the sign is positive (negative).
S_{z_0}	Lifetime signed longitudinal IP significance z_0/σ_{z_0}, with z_0 the longitudinal IP - the longitudinal distance from the PV to the perigee of the track - and σ_{z_0} the error on z_0. A sign is assigned as per the prescription of S_{d_0}.
$\log p_T^{\text{frac}}$	Logarithm of the fraction of the reconstructed jet p_T^{jet} carried by the track $\log p_T^{\text{frac}} = \log p_T^{\text{track}}/p_T^{\text{jet}}$.
$\log \Delta R$(track, jet)	Logarithm of the geometrical distance in 2D angle between the track direction and jet axis $\log \Delta R = \log \sqrt{(\phi_{\text{track}} - \phi_{\text{jet}})^2 + (\eta_{\text{track}-\eta_{\text{jet}}})^2}$.
IBL hits	Number of hits recorded in the IBL - 0, 1, or 2.
PIX1 hits	Number of hits in the innermost pixel layer, after the IBL - 0, 1, or 2.
Shared IBL hits	Number of hits in the IBL that are shared by more than one track.
Split IBL hits	Number of split hits in the IBL, that are created by multiple charged particles.
nPixHits	Total number of hits in all the pixel layers.
Shared pixel hits	Number of shared hits in the pixel layers.
Split pixel hits	Number of split hits in the pixel layers.
nSCTHits	Total number of hits in the Semiconductor Tracker (SCT) layers.
Shared SCT hits	Number of shared hits in the SCT layers.

Table 1.2: Track variables passed to the DIPS model and later versions of the RNNIP model [13]. Compared to the initial RNNIP variables of Table 1.1, the p_T^{frac} and ΔR are passed as log values to reduce the magnitude of the long tail observed at large values and improve the training time. Shared hits are hits used by multiple tracks without being classific as split by a dedicated cluster-splitting NN [38].



(a) Light-rejection.

(b) c-rejection.

Figure 1.6: Light- (left) and c-rejection (right) as a function of b-jet tagging efficiency for RNNIP (green) and DIPS (purple), taken from [13]. Each curve and error bands show, respectively, the mean and standard deviation of the rejections for 5 trainings per algorithm. The bottom panel shows the ratio with respect to RNNIP, showing a clear performance gain for DIPS at all b-jet efficiency considered.

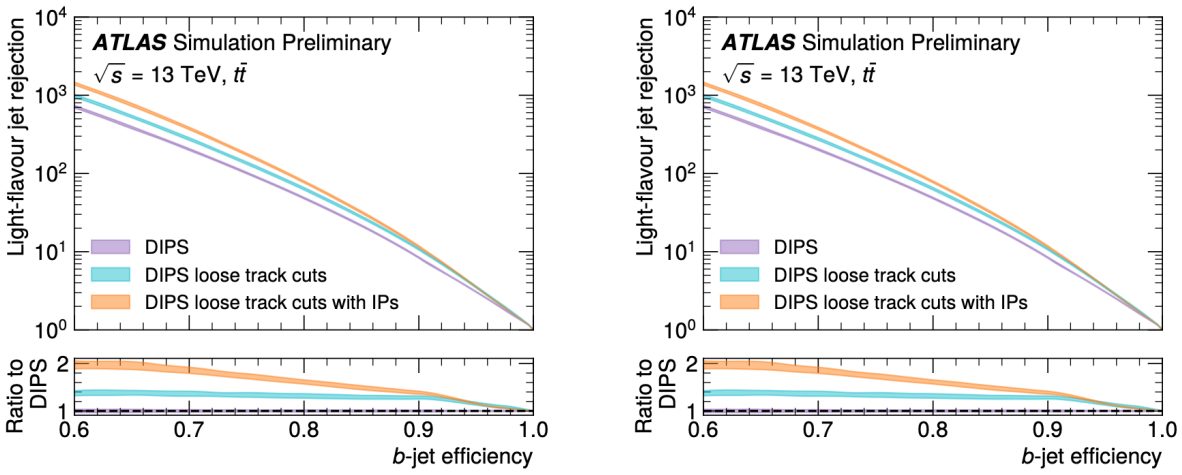
roughly thrice as much, 241 ± 14 seconds per epoch [13]. The faster training time allowed the Collaboration to focus on optimisation studies of the hyperparameters. The optimisation campaign focused on three aspects of the DIPS network: the architecture of the two NN, the track selection, and the set of trac features used as input in addition to those of Table 1.2. Regarding architecture, a grid search over various possible values for the number of layers in Φ and F, number of nodes, and the dimension of the track embedding space showed no significant performance change. The selected architecture is:

- Track network Φ: three layers of 100, 100, and 128 units applied to each track.
- Jet network F: four layers of size 100, 100, 100, 30 before the final output of size dictated by the number of flavour to identify.

To regularise and avoid overfitting, both batch normalisation and dropout were tested with the former observed to give better results.

The second optimisation step however uncovered that a variation to the track selection does offer opportunities for improved performance. Jets are reconstructed with the anti-kT algorithm with a radius of R = 0.4. For RNNIP, IP2D, and IP3D, the selected tracks must pass the following quality selection: ≥ 7 hits in the silicon layers, ≤ 2 missing hits in the silicon layers, ≥ 1 hit in the pixel detector, ≤ 1 hit shared by multiple tracks, $p_T > 1$ GeV, $|d_0| < 1$ mm, and $|z_0 \sin(\theta)| < 1.5$ mm. For DIPS, a looser track selection increasing the acceptance of the last three cuts was studied, modifying the nominal selection in the following way: $p_T > 0.5$ GeV, $|d_0| < 3.5$ mm, and $|z_0 \sin(\theta)| < 5$ mm [13]. Loosening the selection and keeping the top 25 tracks as ranked by decreasing S_{d0} to capture more tracks from heavy flavour decays was observed to lead to a significant improvement in performance for jets with a $p_T < 250$ GeV for DIPS. From an Machine Learning (ML) viewpoint, a larger set of input information with more noise can still prove beneficial if the underlying model is complex enough to capture useful features in the noisy data, that would otherwise be erased by a more stringent selection. The performance gain from this loosened selection-trained DIPS is displayed in the ROC curves of Figure 1.7.

Furthemore, clear benefits are obtained when adding additional track features as input on top of the looser selection, as shown by the orange curve of Figure 1.7, which plots the performance of a loose track selection DIPS trained with the per-track IP parameters d_0 and z_0 in addition to the features of Table 1.2.



(a) Light-rejection.

(b) c-rejection.

Figure 1.7: Light- (left) and c-rejection (right) as a function of b-jet tagging efficiency for different DIPS model, with the baseline (nominal) DIPS in purple, the loosened track selection in blue, and the fully optimised DIPS in orange, from [13]. The curve and error bands show, respectively, the mean and standard deviation of the rejections for 5 trainings per algorithm with different initial random seed. The bottom panel shows the ratio with respect to the baseline DIPS, showing a clear performance gain from the two steps optimisation procedure at all b-jet efficiency considered.

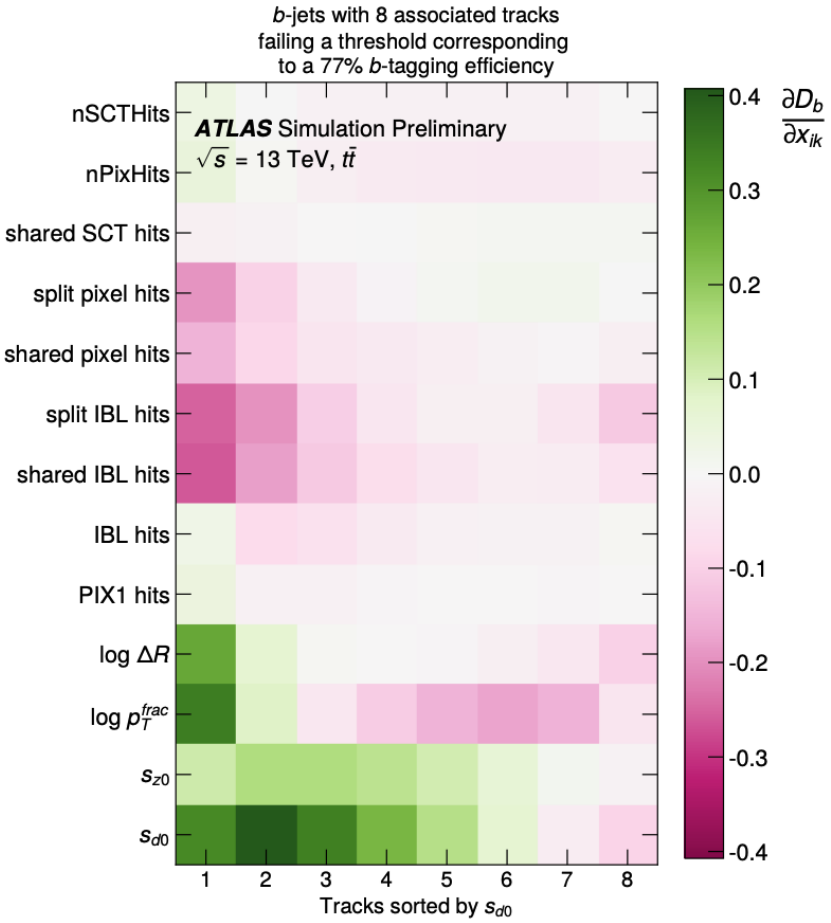


Figure 1.8: Saliency map for b-tagging with 8 tracks sorted by $|S_{d_0}|$, showing the gradient of the discrimant D_b with respect to the track features x_{ik} displayed on the y-axis [13].

How does DIPS work? Interpretability of machine learning models is an active area of research. Several effective approaches exist to gauge the importance of the input on the prediction, such as Shapley values. Figure 1.8 presents an alternative technique called *saliency maps* [39].

Using the b-tagging discriminant D_b of Equation 1.1 at a fixed efficiency of 77%, the average importance of each feature in the track inputs is assessed by averaging the gradient of the discriminant with respect to the track feature over a set of N jets with strictly 8 associated tracks failing the threshold:

$$\frac{\partial D_b}{\partial x_{ik}} = \frac{1}{N} \sum_{j=1}^{N} \frac{\partial D_b^j}{\partial x_{ik}^j}, \tag{1.5}$$

where i indexes the 8 tracks, j indexes the jet in the sample of size N, x_{ik} is the k^{th} feature of the i^{th} jet [13]. This process effectively indicates the linear sensitivity of the discriminant on the track features. Using the saliency map, one can infer what features to modify to correct the failed tagged assigned to the b-jets sample. The most sensitive parameters are measured to be the IP significances of the first five tracks, and the logarithm of the p_T^{frac} and ΔR of the track with largest $|s_{d_0}|$. This observation is physically motivated by the dynamic of the harder fragmentation of b-quarks, compared to light- and c-quarks. Negative gradients are measured for shared and split hits observables, translating into a further incorrect discriminant under linear change of these features. This is also physically motivated, as higher count typically trace back to denser environment where random combinations of hits to form tracks are more likely. However, total hit counts in the different tracker layers have a small positive impact, as these correlate with the reconstruction of the IP parameters.

1.2.3 Training of DIPS with Variable Radius Jets for Run 3

The physics program of the ATLAS Collaboration covers a wide range of analyses, targeting different topologies and processes at different energies. With respect to flavour tagging, a particularly relevant aspect is the energy or tranverse momenta of the jets to label. Indeed, flavour tagger are extremely sensitive to the dynamic of the underlying events. At higher energies, corresponding to higher momenta of the hadronised quark or gluon, the jet constituents emenating from the decaying parton tend to be more collimated in the same direction, as they have to share a high initial energy between themselves. This topology confends tracks and blends the rich internal jet dynamics in the measured signature, making tracks separation and secondary or tertiary vertex identification more difficult. Analyses targeting jets from hadronic or semileptonic decays of heavy particles, such as the top t-quark, Higgs, or the gauge bosons W/Z, can easily produce such highly energetic or *boosted* jets.

So far in this chapter, mentions of "jets" were always referring to the object as reconstructed

by the anti-k_T algorithm with a fixed radius $R = 0.4$ applied to PFLow objects, as introduced in Chapter **??**. This reconstruction method proves robust in the hadron collider setting as it both leads to suitably-shaped jet structure and Pile-up (PU)-removing properties. This fixed radius however becomes a hurdle for boosted jet, as their average radius decreases with energy due to the collimation of the jet components. Indeed, the angular separation $\Delta R = \sqrt(\Delta\eta^2 + \Delta\phi^2)$ between the products of a decaying particle X of large mass m_X scales inversely to the transverse momentum [40]:

$$\Delta R \approx \frac{2m_X}{p_T^X}. \tag{1.6}$$

At low p_T^X, the individually produced particle from the decay are sufficiently separated to be reconstructed as individual objects, hence the *resolved* regime label [41]. For example, a non-boosted Higgs decaying to a $b\bar{b}$ pair can be reconstructed as two b-jets with small R. At higher momentum however, the content of the decay is collimated and overlaps: this is the *boosted* regime. The decaying particle X in such a regime is typically reconstructed as a single large radius jets, to catch the different underlying jets, for example with the anti-k_T method with radius $R = 1.0$. Using such a fixed large radius overestimates the size of boosted jets which are easily contaminated by the PU, as well as the underlying event and initial-state radiations.

A novel approach to model jets from boosted object decays is to reconstruct them with Variable Radius (VR) jet algorithm [42], as introduced in Chapter **??**. VR jets have a size that scales with the inverse of the reconstructed jet momentum, thus correctly following the expected dynamic of Equation 1.6. Such a significant change to the jet reconstruction is bound to have an impact on algorithms learning structure from the jet contents, as is the case of all deep learning-based taggers presented in this chapter. These models have therefore to be fine-tuned seperately to this new jet-type for optimal performance, which is the focus of this section.

For the VR-training, the dataset is composed of three samples simulating proton-proton collisions at $\sqrt{s} = 13$ with the following fractions:

1. 85 % of jets are sampled from the $t\bar{t}$ with a maximal p_T of 400 GeV. At least one of the W-boson from the t-quark is required to decay leptonically.

2. 7.5% are sampled from Z' events, where an exotic boson Z' decays as $Z' \to q\bar{q}$ or $\tau\bar{\tau}$, with a variable Z' mass to generate a flat p_T spectrum extending the p_T-range of the jets studied up to 4 TeV. These jets are required to have a $p_T > 150$ GeV.

3. 7.5% are sampled from a simulated graviton process to also increase the range towards higher momenta. These jets are required to have a $p_T > 150$ GeV.

The simulation process is similar to that introduced in Section 1.1.3. Figure 1.9 displays the jet p_T and $|\eta|$ distributions for the hybrid sample as well as the individual samples it is based upon, for a total of 40×10^6 jets per flavour in $\{b, c, \text{light}\}$. In order to reach such high statistics, importance sampling is used to over-sample the limited amount of c-jets while using all available b- and downsampling light-jets. A particularity of the processing is the requirement for the p_T and $|\eta|$ spectra to be equally-distributed for all jet flavours, so that these features arising from inherent physics effects in the specific processes simulated cannot be used by the model to discriminate between flavours. The technique implemented is importance sampling with replacement. It selects jets of different flavours to match a target distribution. The importance sampling weights are derived by first deriving the ratio of the target 2D distribution to the per flavour one. Weights above 1 indicate jets in the i, j bin have to be oversampled, while values lower than 1 indicate the typical downsampling requirement. Jets are then iteratively sampled until the sampled distribution of each flavour individually matches the target distribution. As displayed in Figure 1.9a for which the target were b-jets, the thus constructed distribution as the same p_T and $|\eta|$ distributions for all flavours. This work introduced to the first implementation of the importance sampling method, now widely used to develop flavour tagging tools.

The optimised DIPS model with 62,167 learnable parameters from the previous section was trained for 200 epochs on 4 Quadro RTX 8000 GPU. The learning rate started at 0.001 and was reduced by a factor 0.8 on plateaus of 3 epochs, with a batch size of 15k jets, batch normalisation, and a dropout rate of 0.1 for the F network. Training proved stable with no signs of overtraining. The model at the epoch giving the smallest loss on a heldout validation set of 300k jets as well as the best light- and c-rejections at a fixed 77% b-tagging efficiency was selected for further comparison. Figure 1.10 shows the ROC curves for b- and c-tagging of the best DIPS model on VR-jets (blue), as well as some comparison to the DIPS model trained on PFlow jets (orange) and RNNIP trained on VR-jets from the previous software release R21 (green).

Training DIPS on a dedicated set of VR-jets clearly improves performance over relying on the PFlow-trained version, as observed by comparing the blue (VR-trained DIPS) to orange curves (PFlow-trained DIPS). At a b-tagging efficiency of 77%, the light-rejection is PFlow-trained DIPS is indeed roughly 40% lower. However, the c-rejection does not benefit as much, being either on par or even lower for the VR-trained DIPS on the $t\bar{t}$ samples. This difference in performance indicates an inappropriate choice of f_c value for the b-tagging discriminant of the VR-trained

DIPS. A so-called *flavour fraction scans*, displaying the rejections at a fixed tagging efficiency for different value of the flavour fraction, can lead to a better choice for a balanced improvement in both background jet rejections. However, DIPS probabilities are not meant to be used directly as discrimant but rather passed on to the high level algorithm DL1d, hence this optimisation is reserved for the final model as presented in Chapter 1.2.5. Figures 1.10d to 1.10f lead to similar conclusions for c-tagging.

1.2.4 Training of DL1d & DL1r with PFlow for Run 3

The ATLAS Collaboration continuously updates its software, updating specific methods to adopt new techniques, maintaining its many tools and adding capabilities. In preparation for the current Run 3 of the LHC that started in 2022, ATLAS improved its reconstruction software from release 21 (R21) to release 22 (R22). As such, important elements used by flavour tagging methods have changed, requiring to retrain all taggers to ensure optimal performance under the new conditions. This work presents the first ATLAS study of the retraining of DL1r on the new release R22 and the first training of DL1d, including the DIPS sub-tagger in the high-level flavour tagging tool. Other important novelties of this work are the possible inclusion of τ-jets in the Deep Learner 1 Model (DL1) model's predictions and a new technique to efficiently process the training data into high statistics dataset using importance sampling, as mentioned in the previous section. The interest of including τ stems from their tendency to be miss-classified as c-jets when hadronically decaying, as both particles commonly leave three particles in the detector. The resulting taggers are observed to efficiently identify τ-jets thereby providing a new way to perform τ-identification and improving c-jet tagging. However, due to the widespread use of the Flavour Tagging Group (FTAG) algorithms and the difficulties arising in calibrating a tagger with excellent rejection against τ-jets, these are not included in the default version of the tagger nor in the results shown here, but are actively under study for the new generation of tagger in the GN family.

Two samples, the $t\bar{t}$ and Z' from proton-proton collisions at $\sqrt{s} = 13$, are simulated and combined in the datasets, as described in Section 1.1.3. For both samples, PFlow jets are reconstructed using the anti-k_T algorithm with radius $R = 0.4$. These two samples are combined into a single *hybrid* sample to train the taggers, with 70% of the total number of jets coming from $t\bar{t}$ and the remaining from the Z'. The $t\bar{t}$ and Z' samples cover, respectively, a low- and high-p_T region based on a reconstructed b-hadron p_T separation threshold of 250 GeV for b-jets and a jet p_T of 250 GeV for non-b-jets. They are re-sampled to have the same $p_T - |\eta|$ distributions, as described

in the next paragraph. The relative proportion of each sample was chosen to avoid any discontinuity in the p_T spectrum at their junction, as evidenced in Figure 1.11. The final evaluation of the performance of a trained tagger is performed on separated test sets of both processes and unfolded over the flavours.

ATLAS flavour tagging tools are widely used across the Collaboration. It is therefore essential for the taggers not to learn specific features of the processes simulated but to focus on the inherent differences between the studied flavours in order to generalise to other processes. An effective way to limit the importance of the simulated processes is to downsample the hybrid sample in $[p_T - \eta]$ bins to have the same number of b-, c-, and light-jets in each 2D bin. This removes the distinction of kinematic phase space between each flavour due to the process specific physics. To avoid biasing the output of the tagger towards the most likely flavours in the process, each jet-flavour is also required to be equally likely in the training set, a requirement satisfied by having the same yield of b-, c-, and light-jets. Applying this technique, the total statistics available for the R22 training is of 25×10^6 jets per flavour for training. The $t\bar{t}$ and Z' samples for validation and testing are each made of 1 million jets and are not downsampled to have the same $[p_T - \eta]$ distribution nor the same yield of different flavours: they represent a realistic distribution of the underlying processes. The main limitator when downsampling are c-jets, as all c-jets from the $t\bar{t}$ process are selected which limits the amount of b- and light-jet that can be taken. This process is extremely wasteful, using only 17% (11%) of all available b-jets (light-jets) in the $t\bar{t}$ sample.

Training is done with the Umami framework [43] based on TensorFlow [44] for 300 epochs with a variable learning rate schedule and the default network structure adopted in the previously released DL1r (R21): 8 fully connected NN of smoothly-decreasing sizes in [256, 128, 60, 48, 36, 24, 12, 6] with Rectified Linear Units (ReLU) activation leading to a final softmax layer producing the predicted probabilities for each flavour. While the DIPS probabilities used as inputs to DL1d come from a model trained on the new release, the RNNIP probabilities are still from a model trained on the previous one (R21) [12, 13]. Indeed, due to its significantly lower performance, RNNIP is no longer supported in the new release and is included for sake of comparability to the previous techniques. The models at an epoch offering the best combined results in terms of b-tagging efficiency and rejection from b-jets on the validation set are selected for further analysis. Importantly, every training converged to a fixed set of performance values, with no overtraining occurring.

Several modifications to the model architecture, list of input variables, and preprocessing and training procedures have been explored, with no significant gain observed:

- The preprocessing steps were revised to reduce the size of the evaluation sets for the benefit of the training one. A dual approach, downsampling light-jets and upsampling c-jets to the b-jets $[p_T - \eta]$, has also been implemented. As previously described, this approach uses importance sampling with replacement to obtain the same fraction of the different flavours and the same p_T and $|\eta|$ distribution. While the performance of the majority classes was observed to improve, the efficiency at tagging the upsampled minority class (c-jets) was slightly lower. This trade-off can be compensated by modifying the flavour fractions and thus does not result in any significant performance changed. This is likely due to the model saturating its performance given the large dataset already available. Other models, such as those from the GN family that have more parameters, have however been observed to make gains from the importance sampling approach.

- Several modifications to the list of input features have been attempted, with no clear advantage uncovered. Adding pile-up information (the actual number of interactions per crossing and the number of primary vertices were tested) was not observed to have an impact on the tagging efficiency. Adding other variables from SV1 or JetFitter was also not observed to improve performance. However, a positive observation is that the IP2D and IP3D taggers can both be safely removed without changes to the performance, as the information they add is in all likelihood now covered by the DIPS sub-tagger, thereby reducing the list of sub-taggers to maintain and simplifying the architecture.

- The structure of the network and its training procedure, leveraging transfer learning. Using samples produced with an older release of the ATLAS software (R21) to pre-train the model was not observed to deliver a boost in performance when later training on the new release (R22). Changing the size of the network and the batch size was also not observed to have a positive effect.

The conclusion driven by the lack of improvements from these three attempts is that models built on this simple DNN structure with large dataset are already likely saturating their performance from the set of inputs. The performance of the retrained DL1r tagger on the new release was found to be in good agreement with the at-the-time released DL1r, despite using the same training of RNNIP on the previous release. In order to establish a meaningful benchmark

for the newly trained taggers, the performance of the then released DL1r tagger, trained and evaluated on an analogous set of samples from the previous release (R21), is included in the following results as benchmark under the label *Recom. DL1r*. A first look at the new family of taggers is also advertised by plotting the performance of a pre-release GN1 tagger, although this is discussed in further details in the next Chapter 1.3.

Figure 1.12 presents the ROC curves on the $t\bar{t}$ (left) and Z' (right) samples for b-tagging. These ROC plots show, on the x-axis, the b-tagging efficiency (ϵ_b^b) versus, on the y-axis, the rejection $\mathcal{R}_Y^b$ for $Y \in [c, \text{light}]$. The two bottom sub-plots present the ratio of the c-jet rejection and light-jet rejection curves to the blue ones. This blue curve is the recommended DL1r performance and serves as the baseline of the comparison, while the new tagger DL1d is plotted in orange. Figure 1.13 shows the same plots for c-tagging, with respect to b- and light-jet rejections. The important observation is the clear gain obtained when replacing RNNIP with DIPS. Both the b- and c-tagging performance of DL1d clearly dominate the DL1r versions, with a significant improvement in background flavour rejection for all tagging efficiency considered, as summarised in Table 1.3. The largest improvement in performance is obtained for b-tagging on the $t\bar{t}$ process, corresponding to a lower jet momentum. This latter points to a dynamical behaviour of the DIPS subtagger that can be traced back to the looser jet selection. Higher momentum jets are more likely to have a larger set of tracks and these tracks tend to be closer to each other due to relativistic boosting. The looser selection forces the DIPS model enduce to sift through a larger set of noisy tracks which brings lower performance at higher momentum, while a gain is obtained at lower momentum from the nicer geometrical separation and smaller initial set.

The light-rejection from b-jets ROC curve in Figure 1.12 traces an elbow at high b-jet efficiencies. This effect is also present in the b-rejection from c-tagging, Figure 1.13. Both correspond to a set of, respectively, light-jets and b-jets that do not overlap with the b-jets b-tagging and c-jets c-tagging discriminants distributions, as shown in Figures 1.14 and 1.15. These "background" jets are easily removed from the core set of "signal" jets due to inherent differences between the flavours and the discrete nature of some sub-taggers used.

The background rejections of the various taggers for b-tagging (c-tagging) as a function of the jet transverse momentum p_T at an inclusive b-efficiency of 70% (c-efficiency of 30%) per region displayed are shown in Figure 1.16 (Figure 1.17). Throughout the p_T range considered,

DL1d outperforms the DL1r tagger. The low p_T b-rejection from c-jets is noticeably better for the newly trained tagger compared to DL1r. The discontinuity of the rejections between the two processes arises from the inclusive b-tagging efficiency being computed inclusively per-region and not exclusively for the whole range.

b-tagging on $t\bar{t}$		
WP	c-rejection	light-rejection
60%	+26%	+73%
70%	+19%	+56%
77%	+12%	+41%
85%	+7%	+32%

b-tagging on Z'		
WP	c-rejection	light-rejection
60%	+19%	+43%
70%	+10%	+32%
77%	+9%	+26%
85%	+6%	+19%

c-tagging on $t\bar{t}$		
WP	b-rejection	light-rejection
25%	+26%	+5%
30%	+25%	+9%
40%	+22%	+12%
50%	+18%	+15%

c-tagging on Z'		
WP	b-rejection	light-rejection
25%	+12%	+22%
30%	+11%	+19%
40%	+8%	+14%
50%	+7%	+10%

Table 1.3: The change in background flavour rejection of DL1d relative to DL1r at various tagging efficiencies, both trained on the new release. Top: b-tagging ($f_c^b = 0.018$); bottom: c-tagging ($f_b^c = 0.2$); left: $t\bar{t}$; right: Z'.

In Figures 1.12 and 1.13, a GN-like tagger trained on 20 million jets from the new family base on GNN that was in development at the time is introduced: GN1 [16]. This model is based on a graph attention network (GAT) directly processing low-level inputs, thereby diverging from the traditional ATLAS flavour tagging philosophy of combining several low-level sub-taggers into a high-level one, such as in DL1d. As examplified in this plot, the method offersa significant boost in performance and is explored in further details in Chapter 1.3.

The DL1d model, integrating the Deep Set-based DIPS network in the classical DL1 hierarchical approach, was a valuable step in the development of a modern performant flavour tagger for ATLAS. Thanks to its simularities with the previous DL1r generation of tagger, built with the RNN-based RNNIP, it was smoothly integrated in the processing pipeline of the flavour tagger group. Its quick callibration lead to its rapid introduction to the Collaboration that used it in several analyses, such as di-Higgs searches decaying to $b\bar{b}$ pairs and Run 3 analyses. To exploit the full potential of the trained model and to catter to specific needs of each experience, several working points were defined and calibrated. An important parameter to control the relative importance of the jet classes to be rejected with the discriminants of Equations 1.1 and 1.2, light and c for b-tagging and light and b for c-tagging, are the flavour fractions f_c and f_b. Naturally, this is a trade-of: for b-tagging, a larger f_c-value favorised a better c-rejection at the cost of a degraded light-rejection. To measure this dependency, flavour fractions scans are performed at a fixed b-tagging (c-tagging) efficiency of 77% (30%) in Figure 1.18a (Figure 1.18b).

With regard to interpretability, it is of course challeging to outright explain the decision process underscoring the predictions of DL1d. An effective technique to measure the relative importance of the different variables is to quantify their contribution to the output using Shapley values [45]. This technique for model explanation calculates the average contribution of each input to the output [45]. Figures 1.19 and 1.20 present the outcome of applying the framework proposed in Ref. [46] to approximate the Shapley values of the inputs to the b-tagging D_b and c-tagging D_c discriminants of DL1d. These so-called *beeswarm* plots measure the impact of the evidence on the output of the model for each input feature. The plots display how each feature' Shapley value modifies the discriminant by moving from a prior background-data distribution expectation to the final model prediction using the real feature. A set of test datapoints of the targeted jet distributions are sampled and, for each, a prior expectation was randomly sampled for the initial test and the impact of using the real value was measured. Positive Shapley values

indicate variables having an increasing effect on the discriminant, thereby helping either b- or c tagging as per the plot considered. Each datapoint is coloured on a gradient scale from low-eature value in blue to high feature value in red, and the dots pile up to show density of the distribution. A feature that has a more weight of its Shapley values distribution at larger values of the feature can be expected to help the model in identifying the main flavour of jets. Conversely, if for large values of the feature the Shapley values are negative, the feature value should be lowered for the model discriminant to improve.

Inspecting Figure 1.19 reveals some interesting patterns in the DL1d network for the task of b-tagging. The most important family of features for this task are the DIPS probabilities, with higher values of p_b correctly identifying the jet as b while higher values of p_c and p_{light} (noted p_u) have the opposite effect. The number of 2-track pairs from SV1 and some JetFitter variables - namely the mass of the vertex, the energy fraction and the number of tracks at the vertex - are also highlighted as important features. These observations are in line with the physics-based reasoning about the dynamic behind the jet: b-jets are expected to have a large charged particle multiplicity and the exchange of momentum is hard, with the b-hadron taking most of the b-quark momentum. Some other interesting features to consider are the ones formatted as "algoName_isDefaults": they track whether the base-method "algoName" is activated (0 - blue) or not and thus defaulting (1 - red) for each jet. Interestingly, most of the occurences of a defaulting behaviour of SV1 and JetFitter are associated with a negative Shapley values, demonstrating the validaty of the physics-reasoning behind these methods and their active contributions to b-tagging. IPxD variables generally score low in the ranking, indicating these methods contribute little to the model predictions and can be safely removed, an observation confirmed by direct searches over the input features set. Contrasting the Shapley values for $t\bar{t}$ (left) and Z' (right), the same variables roughly rank in the same order with minimal differences explained by the change in kinematic phase space between the two samples.

The same analysis can be carried out for c-tagging, with the results displayed in Figure 1.20. As discussed for b-tagging, the most important features are again the DIPS probabilities with p_c ranking first and contributing the most to D_c. Interestingly, the ranking of features is roughly the same as for D_b, with most features that had a positive impact on D_b when taking larger values now having a negative impact on D_c. This is the case of most of the JetFitter and SV1 variables. Defaulting behaviour of these algorithms, occuring when the conditions of a jet do not pass certain requirements, often has a positive effect on D_c as expected. Again, the IPxD family

of features score low, indicating the limited importance of their contributions to the output.

1.2.5 Training of DL1d on Variable Radius Jets for Run 3

As for DIPS, changing the jet definition from PFlow to VR-jets is expected to have a large impact on the performance of the methods described here. Building on from the VR-trained DIPS model introduced in Section 1.2.3, this section presents the training of DL1d for VR-jets. The datasets are similar to those of Section 1.2.3. The VR-trained DL1d was trained for 300 epochs with no signs of overtraining. Its performance here is compared to the PFlow version introduced in the previous section, as well as the R21 DL1r version trained on VR-jets too and a pre-release GN1 trained on 20 million VR-jets.

A clear benefit from retraining on the dedicated VR-jet sets is observed on the ROC curves, with the VR-DL1d outperforming the PFlow version for all b- and c-tagging efficiencies considered. Introducing DIPS in the DL1 architecture has a significant impact on the performance of the tagger and greatly overmatches the RNNIP contribution. This is further highlighted by Table 1.4 reporting the rejections obtained at different WP of typical interest in analyses.

b-tagging						
	$t\bar{t}$		Z'		Graviton	
WP	c-rej	light-rej	c-rej	light-rej	c-rej	light-rej
60%	+20%	+6%	+14%	+83%	+19%	+72%
70%	+18%	+9%	+14%	+65%	+16%	+57%
77%	+13%	+15%	+13%	+56%	+14%	+51%
85%	+1%	+25%	+11%	+45%	+12%	+40%

c-tagging						
	$t\bar{t}$		Z'		Graviton	
WP	b-rej	light-rej	b-rej	light-rej	b-rej	light-rej
25%	-20%	+137%	-17%	+90%	-17%	+80%
30%	-25%	+114%	-21%	+73%	-19%	+66%
40%	-29%	+99%	-23%	+53%	-22%	+48%
50%	-29%	+80%	-24%	+39%	-22%	+35%

Table 1.4: The change in background flavour rejection of VR-trained DL1d relative to the PFlow trained DL1d at various tagging efficiencies, both trained on the new release. Top: b-tagging ($f_c^b = 0.1$ and 0.018 for the VR and PFlow trainijng); bottom: c-tagging ($f_b^c = 0.2$); left: $t\bar{t}$; centre: Z', left: graviton.

As shown in Table 1.4, the new VR-trained DL1d is found to outperform the PFlow version with the flavour fraction parameter for b-tagging f_c^b changed from 0.018 (for PFlow) to 0.1. For c-tagging, a clear gain in light-rejection comes at a cost of b-rejection which can also be corrected by an appropriate change of the flavour fraction parameter for c-tagging f_b^c, currently set at 0.2.

As concluded in Figure A.1 of Appendix A.1, which displays flavour fractions scans for b- and c-tagging, this choice of f_b^c is not optimal for the 30% WP.

While this physics-motivated architecture optimisation moving from an RNN-based to a Deep Set-based track analyser improves the efficiency of the hieararchical model, a clear gain in performance is accessible throught a more radical modification of the architecture as is done with the GN1 model. This is a classical observation in the world of machine learning: vast amount of low-level noisy data can be better exploited by sophisticated architecture than by using a simple model fed a few highly engineered and reconstructed features, even when these are physically motivated. GN1 is not based on any physics principles. As will be shown in the next section, tracks themselves contain enough of the rich physics signature required to unlock the label of the jet they compose.

1.3 The Graph Neural Network family of Tagger

As previously advertised in the PFlow- and VR-trained DL1d, the new generation of classifiers developped for flavour tagging at ATLAS introduce a fundamental shift in design. It moves away from the hieararchical approach, using low-level specialised methods based on physics-inspired algorithm or neural network as input to a high-level neural network. Instead, a single large neural network operates on a rich set of track information as well as some jet features to directly output the per-flavour probabilities. As suggested in Figure 1.22, this change to the flavour tagging software stacks greatly simplifies the maintenance and development effort, as all attention can be focused on a single network. A new framework called Salt [47] built on PyTorch [48] is introduced to simplify the definition and training of multitask multimodal models with multiple GPU. This large network is built on a far more powerful and rich architecture with advanced expressive powers, thanks to a modified graph attention network (GAT) [49, 50] for GN1 or a transformer encoder for GN2 [51].

GN1 uses the information associated with charged tracks in a jet to directly output the flavour-tag probabilities, which are then combined into analogous discriminants to Equations 1.1 and 1.2. This constitutes the primary goal of the network and the real point of developping this network. Alongside predicting the flavour of the jet flavour, auxiliary objectives can also be optimised to aid and guide the training. This so-called *multitask* framework is a common way to input expert knowledge in the design of a ML method, focusing the attention of the network on spelled out metrics. In this case, two side tasks are passed along due to the physically meaning

they highlight:

1. Track origin prediction: a classification task aiming to assign a physical process from which the track arises as described in Table 1.5. The flavour of a jet is strongly correlated to the origin of the tracks. This task brings the attention of the network to this important element as a form of supervised attention [52].

2. Vertex prediction: a classification task predicting whether two tracks come from the same vertex. The decays of b- and c-hadrons include secondary and tertiary vertices inside a jet. Highlighting the compatibility of two tracks to share a vertex allows the model to infer the presence of such vertices. On the truth side, vertices with a distance < 0.1 mm are merged, and tracks labelled as Pileup or Fake are forced not have any shared vertex.

These complementary objectives use truth information from the simulation and cannot therefore be predicted at inference time on real data. They improve performance during the training by providing useful information on the content of the jets. A modified approach, pre-training on the auxiliary objectives and then fine-tuning on the primary objective, was not observed to lead to a gain in performance.

Truth Origin	Description
Pileup	From a pp collision other than the primary interaction
Fake	Created from the hits of multiple particles
Primary	Does not originate from any secondary decay
fromB	From the decay of a b-hadron
fromBC	From a c-hadron decay, which itself is from the decay of a b-hadron
fromC	From the decay of a c-hadron
OtherSecondary	From other secondary interactions and decays

Table 1.5: Truth origins used to label the physics process leading to the produced tracks, from [16]. Charged particles and tracks are matches using the truth matching probability [38], and a value below 0.5 is taken to imply the reconstructed track parameters are mis-measured.

Being built with a GNN, the GN1 and GN2 networks are directly adapted to work with variable number of unordered inputs. The input is composed of 21 track with track features listed in Table 1.6. Each track is further decorrated with 2 jet-level features: the jet transverse momentum p_T and signed pseudorapidity η. Tracks are selected based on a selection, slightly modified from the $DIPS$ one: geq 8 hits in the silicon layers with less than 2 shared hits less than 3 holes in the silicon layers, < 2 holes in the pixel detector and tracks must have $p_T > 0.5$ GeV, $|d_0| < 3.5$ mm, and $|z_0 \sin\theta| < 5$mm. A hole is a missing hit that was expected on a layer

Jet Input	
p_t	Jet transverse momentum
η	Signed jet pseudorapidity

Track Input	
q/p	Track charge divided by momentum (measure of curvature)
$d\eta$	Pseudorapidity of the track, relative to the jet η
$d\phi$	Azimuthal angle of the track, relative to the jet ϕ
d_0	Closest distance from the track to the PV in the longitudinal plane
$z_0 \sin\theta$	Closest distance from the track to the PV in the transverse plane
$\sigma(q/p)$	Uncertainty on q/p
$\sigma(\theta)$	Uncertainty on track polar angle θ
$\sigma(\phi)$	Uncertainty on track azimuthal angle ϕ
$\sigma(d_0)$	Lifetime signed transverse IP significance
$\sigma(z_0)$	Lifetime signed longitudinal IP significance
nPixHits	Number of pixel hits
nSCTHits	Number of SCT hits
nIBLHits	Number of IBL hits
nBLHits	Number of B-layer hits
nIBLShared	Number of shared IBL hits
nIBLSplit	Number of split IBL hits
nPixShared	Number of shared pixel hits
nPixSplit	Number of split pixel hits
nSCTShared	Number of shared SCT hits
nPixHoles	Number of pixel holes
nSCTHoles	Number of SCT holes

Table 1.6: Input features of the GN family of models, from [16].

between two recorded hits of the same track. At most the first 40 tracks associated to a jet are selected for processing as ranked by transverse IP significance s_{d_0}. The input feature list includes missing information from the track and shared hits to specifically target high p_T jets, where tracks are more collimated and their separation can be unresolvable with the deployed detector technology. The GN1 and GN2 models shared the presented properties so far. They however differ in the architecture, which is explored in further details in the next sections.

1.3.1 GN1: a Graph Attention Network for Flavour Tagging

The architecture of GN1, described in Figure 1.23, relies on a modified graph attention network [50] specifically designed for graph learning on sets, the so-called Set2Graph [53]. The composition of the network architecture was subject to a coarse optimisation of the hyperparameters. The first step takes all tracks, each represented by a vector of features composed of the 21 track features plus the two jet features, and embeds each of these track vectors into a latent space of

dimension 64 with a fully-connected feed-forward network with three hidden layers of 64 neurons. This is simular to a track neural network Φ of a DIPS model.

A fully-connected graph is built with the embedded track representation as nodes. For the purpose of this section, each of the nodes of the graph are labelled as h_i with feature vector of dimension 64, and there is one such node per track. The graph network updates the defined graph $G(\mathcal{N})$ into a graph $G'(\mathcal{N}')$, with $\mathcal{N}$ and $\mathcal{N}'$ the set of edges, by aggregating the features of each node h_i and neighbouring nodes $\mathcal{N}_i$ using the operation of Ref. [50]. In more details, the following 4 steps are applied for a single graph update [16]:

1. Each node feature vector is passed through a fully connected layer W producing an updated representation $W\,h_i$ of size 64.

2. Pairwise scalar edge scores are computed for each pair of nodes i and $j \in \mathcal{N}$:

$$e\,(h_i, h_j) = V^T \theta[Wh_i, Wh_j], \tag{1.7}$$

   where V is a second fully-connected feed-forward layer of size 128, θ is the ReLU activation function, and $[,]$ stands for the concatenation operation.

3. Attention weights are derived from the pairwise edge scores, using a softmax over all j per node h_i:

$$a_{i,j} = \text{softmax}_j\left(e(h_i, h_j)\right). \tag{1.8}$$

4. The final step is to aggregate the information to update each node h_i into h'_i by computing the attention weighted sum over each node representation:

$$h'_i = \sum_j a_{i,j}\,.\,Wh_j. \tag{1.9}$$

For GN1, 2 heads attention with 3 such graph network layers applied in succession were found to deliver optimal performance and no overtraining. The output of the graph network is said to be a conditional track representation as it combines each track representation and its neighbours. The ordering of the conditional tracks is kept similar to that of the original track to support matching of track to their truth information. Furthermore, a global reperesentation is derived by combinijng the conditional track representation with learnable attention weights. This rich conditional and global representations can then be used as inputs for the three objectives,

implemented with three distrinct feed-forward neural networks [16]:

1. Jet flavour prediction: performed by a graph classification network fed the global representation only. The primary objective of predicting the jet flavour is done by this network, composed of 4 hidden layers with 128, 64, 32, and 16 neurons respectively, finishing on an output of size 3 with softmax for b-, c-, and light-jet probabilities (size 4 if τ are included).

2. Track origin prediction: performed by a node classifier processing each conditional track representation with the global representation. This network is built with three layers of reducing size 128, 64, and 32 to finish on the output layers of size 7 with sotmax, matching the 7 classes corresponding to the different truth origins.

3. Vertex prediction: performed by a nodes pairs binary classifier that receives every possible combinations of representational tracks as well as the global representation. This network is also made of 3 layers of size 128, 64, and 32 for a final output of size 1 with sigmoid, stating whether the pair of tracks have a common vertex.

The architecture of GN1 is an enhanced version of DIPS, with the track initialiser and graph classifiers corresponding to Φ and F. Added elements are the powerful GNN layers and conditional representation pooling layer with attention, as well as the auxiliary objectives.

Training GN1 involves minimising the combining objective $\mathcal{L}_{\text{total}}$ of Equation 1.10 [16]. $\mathcal{L}_{\text{flavour}}$ is the categorial cross entropy loss, as defined in Equation **??**, over the different jet flavours to output the per flavour probabilities. $\mathcal{L}_{\text{track}}$ is the categorial cross entropy loss for the track origin prediction averaged over all tracks in a batch. Due to intrinsic differences in the relative frequency of track origins, the contribution of each origin is weighted by their inverse frequency of occurence. Finally, $\mathcal{L}_{\text{vertex}}$ is the binary cross-entropy of the track-pair compatibility averaged over all track-pairs in a batch. The importance of matching tracks from b- and c-hadron is artificially increased by giving them double the weight compared to other track-pairs.

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{flavour}} + \alpha\,\mathcal{L}_{\text{track}} + \beta\,\mathcal{L}_{\text{vertex}}. \tag{1.10}$$

In Equation 1.10, weights are applied to combine the different tasks that are represented by distinct values, reflecting their specific loss functions and difficulties. Weights of $\alpha = 0.5$ and $\beta = 1.5$ [16] where found to lead the auxiliary objectives to converge to similar values, giving them equal weighting in $\mathcal{L}_{\text{total}}$. The proposed choice for these parameters also let the primary objective $\mathcal{L}_{\text{flavour}}$ dominate the global loss, and small variations of α and β were not found to

significantly impact the performance. The results presented here come from Ref [16], where a GN1 models were trained for 100 epochs with a 30 million jets sample made of 60% $t\bar{t}$ and 40% Z', as previously described in this chapter. The validation loss on a statistically independent sample of 500k jets is monitiored, with the epoch minimising it selected for further analysis. The optimiser is based on Adam [54] with a learning rate of $1e-3$ and a batch size of 4000 jets spread across 4 GPU.

The results of the training are presented in Figures 1.24 and 1.26 for b- and c-tagging respectively, where a DL1r model retrained on similar inputs to the GN1 with 75 million jets is presented as reference, with a significant caveat being the lack of retraining of the input RNNIP sub-tagger. The ROC curves of a GN1 model given an additional track input to those of Table 1.6 indicating if a track was used in the reconstruction of an electron or a muon is also included as GN1 Lep. At the time of deriving these results, the DL1d tagger was yet not officially released and is thus not included. Its performance can be estimated at roughly 20% to 50% above DL1r, far from the observed gains made by the GN1 models - as was also displayed in Figures 1.20 and 1.19.

Most of the improvement in rejections made by GN1 models can be found a lower tagging efficiencies. At the typical working point of 70% on the low p_T region defined by $t\bar{t}$, the c-jet (light-jet) rejection is 110% (80%) above that of DL1r. Gains are observed across the considered p_T spectrum, with a gain of 180% (500%) at a working point of 30% - the 30% working point on Z' corresponds to using the 70% working point on $t\bar{t}$. The GN1 version with lepton information further improves the performance, to a c-rejection (light-rejection) of 180% (150%) at the 70% WP on $t\bar{t}$ and 180% (600%) on the Z' at the 30% WP. Part of the measured performance increase with GN1 is due to the looser track selections leveraged by GN1 and to a more sophisticated exploitation of the noisy low-level track information. The GN1 and DL1r discriminants for b-tagging are presented in Figure 1.25. The distributions for GN1 moves the b-jet distribution to higher values of the discriminants, indicating a higher confidence on the associated p_b.

The c-tagging performance is presented in Figures 1.26 and 1.27, displaying the ROC curves and c-tagging discriminant distributions D_c. GN1 significantly outperforms DL1r for c-tagging: both background rejections are doubled on the $t\bar{t}$ sampled at a c-tagging WP of 25 %, with a more modest increase on the Z' sample of 60% for b-rejection and 100% for light-rejection at the same c-jet WP.

As previously explained, the tagging performance is strongly dependent on the jet energy considered, explaining the observed rejections differences between the $t\bar{t}$ and Z' samples. Higher energies correlate with higher transverse momentum p_T. More energy in the system introduces a higher multiplicity of fragmentation particles challenging the reconstruction process. The direction of emission of the particles is more collimated and approaches the resolution power of the tracking detector of fixed granularity: different tracks are no longer individually resolvable and their hits merge. Due to relativistic effect, at higher p_T the time of flight of heavy-hadrons increase, delaying their decay further into the detector. Traces left by the heavy-hadrons paths and fragmentation particles introduce inaccuriacies in the reconstructed track parameters [38]. This degradation of the track quality impacts the jet tagging performance significantly, as displayed in Figure 1.28 showing the b-tagging efficiency as a function of jet p_T for a fixed light-jet rejection of 100 in each bin. GN1 outperforms DL1r across the studied p_T range, with a very significant b-efficiency improvement of a factor ~ 2 at high values of p_T above 2.5 TeV.

To conclude this chapter on GN1, the importance of the auxiliary tasks is discussed by presenting ablations studies removing them iteratively from the full GN1 model. For this purpose, three variants of GN1 were trained equivalently to the full GN1 but without:

- any auxiliary objectives, leading to a model label "GN1 No Aux" only optimising the jet classification objective,
- the vertexing objective but not the track classification one, for the model labelled "GN1 Vert",
- the track classification objective without vertex, referred to as "GN1 TC".

Figure 1.29 displays the ROC curves of theses modified model with respect to the previously introduced DL1r and the full GN1. Removing either or any of the auxiliary objectives is seen to have a large impact on performance. The GN1 No Aux model is effectively similar to a DL1d model, having similar performance gains with respect to DL1r. Remarkably, this performance is obtained from a single network processing tracks without any of the sub-tagger nor methods used by the DL1 family, effectively underlying the powerful representation power of GAT. Adding either of the auxiliary task has the same beneficial impact on performance, as GN1 TC and GN1 Vert performs similarly and each is enough to outmatch DL1r. The real gain is obtained by adding both auxiliary taks, which further boosts the effectiveness of the model.

So far, the performance of GN1 on the primary objective of jet flavour classification has been discussed. The performance on the auxiliary objectives was not initially intended to be lever-

aged on real data but only to distill information for the primary goal. The track-pairs vertexing performance can be assessed by leveraging the information to perform track finding: grouping sets of tracks that are found to share a vertex with one another into a single vertex. The result is compared to the truth vertex label available in the simulations. Vertices identified by GN1 as containing tracks coming from a b-hadron decay are grouped together, and the same procedure is applied to the truth information. To measure performance, the reconstructed and true vertices are compared as well as the number of tracks correctly assigned. A vertex is correctly identified when it contains at least 65% of the correct tracks with a purity of at least 50%. The comparison is only carried out for reconstructed tracks, meaning a 100% GN1 efficiency corresponds to correctly identify all possible secondary vertices within the limit of the track reconstruction efficiency. An inclusive reconstruction efficiency in b-jets of $\sim$80% is measured for GN1, effectively proving that the model is able to identify b-hadron decay vertices. An important caveat is the current restriction is only on finding such vertices, not on a reconstructing them. In order to implement a fully-fledged secondary vertex fitter as an auxiliary objective, the fitting of the vertex must be produced by a differentiable algorithm to allow for backpropagation. This is a promising area of research, given the global interest in accessing this much-used SV information. Recent work has been carried out to introduce this task using the differentiable single vertex fitting algorithm of Ref. [55].

Concerning the track origin classification performance, Figure 1.30 presents the traditional ROC curves, comparing the false positive rate (tracks wrongly assigned a label) versus the true positive rate (correctly assigned the label), for the different track origin classes of Table 1.5. Some classes are combined with weights dictated by the class relative abundance: this is the case of the FromB, FromBC, and FromC classes that are combined as Heavy Flavour, and the Primary and OtherSecondary labels. The Area Under the Curve (AUC) of the ROC of all groups is above 90%, indicating good classification performance. The most challenging categories are the Heavy Flavour, Primary, and OtherSecondary tracks, while the Fake and Pileup tracks were most easily identified. The global mean (weighted) AUC was of 92% (95%) on $t\bar{t}$ and 94% (96%) on Z' [16]. This performance is in line with a physics-based intuition, and the p_T effect can be noted by the reduction in AUC for the Heavy Flavour tracks on the Z' sample.

GN1 proved to be an exciting direction of development for flavour tagging at ATLAS. It showed clear benefit from the previous mentality of building network by combining several algo-

rithms and methods with physics meaning. Embrassing modern advanced machine learning, it relies on a single network built around an advanced graph attention network. While the functioning of the model is less intepretable than the previous DL1 family of tagger, expert knowledge is still instilled in the model thanks to the multitask paradigm. Building on from this success, an upgrade architecture was developped to accelerate the speed of training and continue pushing the performance of the method ever higher: GN2.

1.3.2 GN2: a Transformer Encoder for Flavour Tagging

GN2 is not a radical change on the architecture of GN1. Rather, it is a fine-tuned modified model aiming to reproduce the same conceptual processing chain as GN1, only with an easier to train and simpler to scale design. The main modifications with respect to GN1 is the replacement of the computationally complex and expensive graph attention operator by a now famous architecture in machine learning: the transformer [51]. As described in Chapter **??**, the transformer is a remarkably effective and expressive design, both able to extract fine-grained correlations between ordered and unordered tokens in a sequence through the mechanism of attention and to scale to very large network size without suffering from overtraining. By design, transformers combine rich attention computing and regularisation inducing steps which let such network scale significantly in number of parameters while guaranteeing effective parallelisable training on GPU hardware.

In the case of GN2, given the design only requires building a global represenation of the sets of tracks composing a jet, only the encoder part introduced in Ref [51] and modified in Ref [56] is deployed instead of the GNN component of Figure 1.22. A dense summary of the modifications applied when moving from GN1 to GN2 is presented in Table 1.7. The reference to GN1 corresponds to the last version of the model that was developped which explains some small modifications with the model described in the previous chapter. Similarly, the GN2 model described here corresponds to the first publically released model, and this generation is also being refined and improved at the time of writing this thesis.

Some significant changes adopted for GN2 are a learning rate scheduler, a larger embedding space dimension giving a wider and deeper - thanks to the doubling of the number of layers - of the core units (the GNN for GN1 and the transformer for GN2), and the introduction of regularising effects from layer normalisation and dropout [57]. The learning rate scheduler is based on the one-cycle scheduler of Ref [58], with some important parameters described in

Modification	Parameter	GN1	GN2
Hyperparameter	Trainable parameters	0.8M	1.5M
Hyperparameter	Learning rate	Fixed 1e-3	One-cycle scheduler
Hyperparameter	Core unit layers	3	6
Hyperparameter	Attention heads	2	8
Hyperparameter	Embedding dimension	128	192
Architecture	Attention Type	GATv2	Scaled dot product
Architecture	Dense update	No	Yes (dim 256)
Architecture	Separate value projection	No	Yes
Architecture	LayerNorm + Dropout	No	Yes
Inputs	Number of training jets	30M	192M

Table 1.7: Main modifications between the last generation of GN1 and the first generation of GN2, taken from [17].

Table 1.8. This scheduler speeds up the training by initially growing the learning rate to large values, corresponding to large steps in the parameters' optimisation landscape, before annealing progressively the learning rate to small values, helping to converge on a specific minimum [59]. The attention implemented in the transformer allows similar physics performance at a reduced memory footprint and training time [18]. The improved computational performance of GN2 made it possible to scale the network up in terms of number of parameters, GN2 having roughly twice as many parameters as GN1, but also in terms of training dataset. GN2 can indeed be trained on roughly $\times 6$ more jets than GN1 with the same computing resources. The datasets for the GN2 training presented here was derived in the same fashions as those of DL1d and GN1, using importance sampling to fully utilise the b- and light-jets statistics.

Parameter	Description
LR initial	Initial value of the learning rate
LR maximal	Maximal value of the learning rate, reached at the end of warm up
LR final	Value of the learning rate reached at peak epoch
Warm Up	Period covering the increase from initial to maximal
Peak epoch	Epoch at which LR maximal should be reached

Table 1.8: The five parameters of the one-cycle scheduler.

The attention mechanism in the transformer is subtly different from the GAT, and corresponds to the multihead self-attention process described in Chapter **??**. The nodes are updated in a two steps approach: first attention is computed and applied, then a dense layer updates the set of nodes. In more details, the transformer implements the following update on the set of nodes $h_i \in \mathcal{N}$ defining the fully connected graph $G(\mathcal{N})$:

1. Layer normalisation is applied to the input set of nodes $\mathcal{N}$

2. For each attention head, three individual mapping represented by layers W_q, W_k, and W_v map each node $h_i \in \mathcal{N}$ to three independent representations $W_q h_i$, $W_k h_i$, and $W_v h_i$.

3. For each node $h_i \in \mathcal{N}$, edge scores are computed with all nodes h_j using the scaled dot product attention

$$e\left(h_i, h_j\right) = \frac{W_q h_i \,.\, W_k h_j}{\sqrt{s}},$$

where the s parameter representing the scaling weight is typically taken to be the dimension of matrix W_k.

4. The edge scores are turned into attention scores for node i, by taking the softmax over all nodes:

$$a_{i,j} = \text{softmax}_j\left(e(h_i, h_j)\right)$$

5. Each node $h_i \in \mathcal{N}$ is updated into a node $h'_i \in \mathcal{N}'$ as:

$$h'_i = \sum_j a_{i,j} \,.\, W_v h_j$$

6. Using a skip connexion, the updated nodes $\mathcal{N}'$ are added the original nodes $\mathcal{N}$ values.

7. Layer normalisation is applied to the updated nodes $\mathcal{N}'$.

8. The updated nodes are passed through a DNN.

9. The output of the DNN is summed to the updated nodes by a skip connexion, given the final updated set of nodes $\mathcal{N}'$.

The GN2 model presented here combines 6 such transformer layers with 8 attention heads in total. A comparison of the global performance of this GN2 model to the DL1r, DL1d, and GN1 models is displayed in the b-tagging ROC curves of Figures 1.31. For this comparison, the GN2 and DL1d models have been retrained on the same datasets, with the DL1r and GN1 models equivalent to those presented in the previous Chapter 1.3.1. GN2 delivers yet another significant boost to performance, drasticaly surpassing the GN1 rejections at all efficiencies considered. The largest improvement is again obtained at lower b-jet efficiencies. Compared to GN1, GN2 delivers $\times 1.5$ ($\times 1.7$) the c-rejection (light-rejection) on $t\bar{t}$ at the 70% b-tagging WP and $\times 1$ ($\times 1.7$) on Z' at 30% WP. With respect to DL1d, the gains in c-rejection (light-rejection) are respectively close to $\times 3$ ($\times 2$) for $t\bar{t}$ and $\times 3$ ($\times 4$) on Z'. The c-rejection on Z' of the GNN models is essentially equivalent, although the significantly improved light-rejection of GN2 indicates its c-rejection

can be boosted by further increasing its flavour fraction f_c^b above 0.1.

Turning to c-tagging, as displayed in Figure 1.32, a similar large performance gained is obtained by the new GNN family over the DL1 one, both in terms of b- and light-rejection. GN2 again introduces a large improvement on top of GN1, although their b-rejection performance is equivalent on Z'. The gains from GN2 with respect to GN1 are of a factor $\times 1.3$ ($\times 1.3$) for b-rejection (light-rejection) on $t\bar{t}$ at the 30% WP, while they are $\times 1$ ($\times 1.2$) on Z'. The comparison to DL1d is of $\times 1.9$ ($\times 2.1$) on $t\bar{t}$ and $\times 1.3$ ($\times 1.8$) on Z' at the same WP.

Fixing the b-tagging performance at the 77% working point for both the $t\bar{t}$ and Z', Figure 1.33 scans the f_c^b flavour fractions for the different models. A clear hierarchy of performance is displayed by these four graphics: GN2 is occupies in an undisputed way the high rejections parts, followed by GN1, DL1d, and finally DL1r. For b-tagging on Z', the c-rejection could be further improved with limited impact on light-rejection by a larger f_c^b choice. However, the flavour fractions are optimised for an improved c-rejection on $t\bar{t}$, with limited change to the light-rejection across tagger generations. If desired, the light-rejection on $t\bar{t}$ of a GN2 taggers could be push upwards by lowering the f_c^b, reaching values as high as 1800 at a c-rej of 4.8. The equivalent DL1d performance is a light-rejection of 450 at a c-rejection of 4.5, equivalent 25% of the GN2 light-rejection. Similarly, GN2 can reach a c-rejection of 19.5 at a ligh-rejection of 110, compared to a maximal c-rejection of 9.7 for a light-rejection of 40.

Figure 1.34 displays the flavour fraction f_b^c scans for c-tagging at the 30% working point. The same conclusions as for b-tagging holds, underlying the overall superiority of GN2. The scans for c-tagging show a different pattern than the b-tagging ones: at large f_b^c, the b-rejection rapidly increases while for b-tagging the c-rejection was saturating. This behaviour is due to the clear identification of b-jets giving them an outlying distribution compared to the overlap of c- and light-jets.

Figure 1.35 displays the effective per bin b-tagging efficiency for inclusive b-tagging efficiency of 70% for $t\bar{t}$ and 30% for Z' in each p_T region considered. The performance is visibly not uniform across p_T, with teh model accomodating specific parts of the p_T spectrum more easily. The region [100, 800] GeV overlapping the two samples is a sweet spot for performance, with more challenging results at lower and higher p_T. The performance for Z' in particular reduces

dramatically with larger momentum, due to the physics reasons previously explained. Figure A.4 in Appendix A.3 displays the same information for c-tagging, leading to the same conclusions.

To avoid biasing the analysis with this per-bin performance dependency, Figure 1.36 displays the b-tagging efficiency distribution across p_T at a fixed per bin light-rejection of 100. The superior capabilities of GN2 are clearly exhibited across the p_T spectrum. The same conclusion holds for c-tagging, as displayed in Figure A.5 of the appendix.

Inspecting the rejections at a fixed b-tagging efficiency of 70% per bin also leads to concluding the clear superiority of GN2. Figures 1.37 and 1.38 respectively display the c- and light-rejection for a 70% b-efficiency per bin, showing that most of the improvement from GN2 and GN1 is in the [100, 800] GeV p_T sweetspot. The same distribution with an inclusive 70% b-tagging efficiency, over the entire p_T regions, are displayed in Figures A.6 and A.7 of the appendix.

To conclude this section, the b- and light-rejection at the 30% c-tagging per bin working point are displayed in Figures 1.39 and 1.40 respectively. Clearly, most of the improvements unlocked by GN2 and GN1 is to be found in the [100, 800] GeV sweetspot of the p_T spectrum.

These results, ableit intermediary as the developpment of the new tagger is still underway at the time of writing, are highly suggestive of the promised performance unleashed by the state-of-the-art GN2 model. Leveraging a simpler design and a more parallelisable architecture, GN2 can effectively grow to larger amount of parameters processing ever larger datasets effectively, with no significant overtraining occuring. The story of modern flavour tagging is a story of refining and ever more expressive machine learning. RNNIP and DIPS required 50-60k parameters, which when used in the high-level algorithm to form DL1r and DL1d gives rise to models with ∼130k parameters. GN1 revolutionises the approach by adopting a single powerful architecture with ∼800k parameters. GN2 modifies this radical new design to adopt a highly efficient, regularised, and parallelisable model that easily scales the number of parameters to ∼1200k, being the first flavour tagger to cross the million parameters threshold. The latest design of GN2 uses 2.6M parameters, and further tests raised this number up to ∼70M parameters. Expert knowledge is passed to the model using supervised attention, framing the intuition as learnable tasks enforced during training.

1.3.3 Optimising GN2

The state-of-the-art flavour tagger at ATLAS is, at the time of writing, built on the GN2 architecture. Naturally, fine-tuning the model is required to further push the performance higher.

Many studies are currently being carried out to deliver yet a stronger tagger than the GN2 version presented in this thesis. A non-exhaustive lists of ongoing research directions include:

- Optimising the track selection and the jet reconstruction type. Moving towards yet a looser selection and letting the network sift through a larger set of background tracks could deliver further performance. Assessing the effect of modelling uncertainties is however of particular importance for these modiciations.

- The inclusion of neutral constituent information. Tracks are reconstructed from hits in semiconductor-based detectors. Such hits are only recorded for charged particles flying through the active regions of the sensors. This entirely misses neutral particles, from neutrons, neutral pions and kaons, and neutrinos. All but the latters leave energy in the calorimeters that is measurable and accessible. Studies are ongoing to add this information to the set of tracks.

- The inclusion of leptonic information. 40% of b-hadrons include either an e or a μ in the jet cone [3]. As seen with GN1, the inclusion of a leptonic information in the set of tracks leads to a significant performance increase. Studies are ongoing to build a finer lepton-information analyser within GN2.

- Hadronic decays of τ are a major source of backgrounf for analysis focusing on c-jet tagging, due to similar signatures. Including theses leptonic decays in the classification objective has been seen to deliver promising results in initial studies.

- Finer output classes categorisation. Currently, the simple labelling scheme requires combining topologies with significant differences. For example, purely hadronic and semi-leptonic decays of b-jets are both labelled b-jets. Adopting a greater flexibility in the definition of classes allows the model to fully utilise the unique signature of each process.

- Integrating further expert information into the design is known to deliver a great boost to performance. Studies are ongoing to upgrade the set of auxiliary tasks, in particular for secondary vertex fitting and reconstruction. A GN2 model able to reliably reconstruct this information would have a use case in the ATLAS experiment beyond heavy-flavour jet tagging, while benefitting from improved performance for this essential task.

Larger design considerations as studied in the above list are paramount to a well-functioning tagger. An equally essential endeavour is to fine-tunie an architecture to extract the best performance from a chosen data processing strategy. This section focuses on some initial studies to

perform Hyperparameter Optimisation (HPO) and network architecture search for GN2. The essential challenge is that a test of higher a change to a hyperparameter or to the model architecture requires to fully train a glsgn2 model with the new choices. This is a costly process, as a single epoch of the GN2 takes roughly ∼28 min for 2 NVIDIA A100 GPU each fed data by 20 Core Processing Unit (CPU) to perform 1 epoch on a 30M jets dataset with batchsize 2k split on the GPU. GN2 has many hyperparameters that should be optimised to deliver optimal performance, among which the most relevant are: initial lr, maximal lr, end lr, the weights of the 2 auxiliary tasks, the amount of weight decay, the batchisze, and the float precision. Architecture-level changes are the embedding dimension (output of the initialiser and as input and output of each transformer encoder), the depth of the initialiser, the number of layers and heads in the transformer encoder, the size of the transformer output, the auxiliary tasks DNN, the activation functions, and the specific loss functions and their class-weights used.

Unfortunately, access to GPU was, at the time of writing, limited for members of the Collaboration. Most of the computing power leveraged to train advanced ML models such as GN2 is accessed on high-performance cluster of institutes members belong to. In this respect, a promising area of development is being pursued by Centre Européen pour la Recherche Nucléaire (CERN), with the introduction of a Kubeflow-backed served hosted on *ml.cern.ch* [60]. Kubeflow, created by Google and now backed by the Cloud Native Computing Foundation, is an open-source framework built on Kubernetes to perform machine learning operations such as training, inference, deployment, and hyperparameter optimisation. The project aims to centralise the GPU resources of the different CERN experiments into a single centralised cluster with datastorage, efficient I/O reading capabilities, and dedicated GPU nodes. Katib, Kubeflow's dedicated HPO workload, is a promising approach to perform effective hyperparameter optimisation with state-of-the-art autoML techniques, which automate and refine the strategy to test and converge on the best hyperparameters [61]. At the time of writing, the server was still in a testing phase with little hardware accessible, thereby removing it from consideration as a possible solution to carry out the full HPO of GN2. However, the salt framework GN2 is trained with was adapted to run on Kubeflow platform and tests showed promising possibilites for the Collaboration. Being accessible to any member of the ATLAS Collaboration, it greatly "democratises" access to projects requiring powerful computing power for insitutes lacking a High Performance Cluster (HPC).

Large NN, for example large language models, develop in the future at ATLAS will require

cluster designed for machine learning, with many GPU accessible on dedicated nodes for splitting the process. This paradigm of computing is markedly different from the typical grid-base distributed computing deployed in particle physics experiments. While MC-based samples and sub-sampled datasets can be effectivly processed in parallel by autonomous parallel jobs, Machine learning however requires communication between the different jobs to keep the weights of the model being updated during training synchronised on the different GPU. A fast connexion between these GPU is essential, as is having fast read access to the full datasets due to the need to loop over the whole data for each epoch during training. Distributing the computation across different HPC geographically distant, as is common with the current CERN computing grid, is not effective for this purpose. The CERN Kubeflow server is an exciting area of development for future computational needs of ATLAS. Furthermore, having a framework compatible with Kubeflow allows operating on multiple platforms, giving the flexibility to scale resource access for computationally demanding tasks, such as HPO. For example, Google Cloud is Kubeflow-compatible and host a larger amount of GPU, but so are other large cloud providers such as Amazon Web Service and Microsoft's Azure. Salt can be effectivly trained on one of these cloud prodividers or the CERN's Kubeflow with no noticeable distinction for the user.

While leveraging a large amount of computing power is the natural solution to the challenging task of HPO of a "large", by ATLAS standards, neural network models, a more refined technique can be exploited in the present case. Recent works suggest that the optimal hyperparameters of a nominal model can be estimated from a smaller model [62]. Here smaller refers to either the depth - the number of layers - or the width - the number of neurons per layer or, in the case of a transformer, the number of heads in the multihead attention - of the neural network. Ref [63] establishes the mathematical foundation backing this surprising behaviour of deep neural network: the Maximal Update Parametrisation (μP). The rest of this section is dedicated to introducing and defining the maximal update parametrisation before establishing its relevance for HPO.

Maximal Update Parametrisation

The maximal update parametrisation is first and foremost a *parametrisation*. In this context, the parametrisation of a neural network refers to the definition of the weights of each individual neurons, the way they are initialised, and how they are updated from a given optimisation

algorithms, such as Adam or Stochastic Gradient Descent (SGD) [54]. In the presented context, the default or *standard* parametrisation (SP) refers to a parametrisation of the weights following the so-called LeCun parametrisation [64]. Such a parametrisation, routinely deployed in ML frameworks such as PyTorch [48], initialises the weights by sampling them from a Gaussian with mean 0 and standard deviation being the inverse of the input dimension of the layer the weight belongs to. For both Adam and SGD, a single master learning rate (LR) η is used when updating all weights. For μP, some subtle difference are applied, as summarised in Table 1.9. Mainly, the output layer weights are sampled from a Gaussian with a standard deviation being the inverse of the input dimension **squared** of the output layer. Concerning the learning rates, the hidden and output layers are scaled down by their respective input dimension for Adam. For SGD, the output layer LR is scaled similarly, but the input and the bias LR are scaled up by the output dimension of the layers.

	Initialisation Distribution		Adam LR		SGD LR	
	SP	μP	SP	μP	SP	μP
$w^{L_{\text{inp}}}$	$\sim \mathcal{N}\left(0, \frac{1}{d^{\text{in}}_{L_{\text{inp}}}}\right)$	$\sim \mathcal{N}\left(0, \frac{1}{d^{\text{in}}_{L_{\text{inp}}}}\right)$	η	η	η	$\eta \times d^{\text{out}}_{L_{\text{inp}}}$
$w^{L_{\text{hid}}}$	$\sim \mathcal{N}\left(0, \frac{1}{d^{\text{in}}_{L_{\text{hid}}}}\right)$	$\sim \mathcal{N}\left(0, \frac{1}{d^{\text{in}}_{L_{\text{hid}}}}\right)$	η	$\eta/d^{\text{in}}_{L_{\text{hid}}}$	η	η
$w^{L_{\text{out}}}$	$\sim \mathcal{N}\left(0, \frac{1}{d^{\text{in}}_{L_{\text{out}}}}\right)$	$\sim \mathcal{N}\left(0, \frac{1}{d^{\text{in}}_{L_{\text{out}}} \times d^{\text{in}}_{L_{\text{out}}}}\right)$	η	$\eta/d^{\text{in}}_{L_{\text{out}}}$	η	$\eta/d^{\text{in}}_{L_{\text{out}}}$
$b^L \; \forall L$	0	0	η	η	η	$\eta \times d^{\text{out}}_{L}$

Table 1.9: Comparing the Standard Parametrisation (SP) to the Maximal Update Parametrisation (μP), as defined in Ref [62] based on the work of Ref [63].

This particular derivation of μP, derived in Ref [62], is equivalent to the original μP derivation spelled out in Ref [63]. μP turns out to be the unique parametrisation that maximally updates the weights of a neural network, where "*maximal update*" refers to the size of the update of a network of width tending towards infinity. For such a hypotethical model useful only for theorical considerations, the updates naturally must be independent of the width, otherwise they would become infinite leading the model to be unstable. For the specific case of the attention mechanism as present in the multihead attention of transformers, the scaling has to be modified from $\sqrt{d_k} \rightarrow d_k$ to properly scale with width, as shown in Ref [62]. Figure 1.41 shows a comparison of a GN2 model with μP parametrisation to a standardly parametrised GN2, referred to as the SP model. Each curve displays, for different embedding width in the transformer and the track initialiser, the sum of the absolute values of the weights before the activation ($L_1(\text{layer}) = \sum_{w_i \in \text{layer}} |w_i|$) for the initialiser and transformer models only. Three timesteps are displayed for each model, the

initialisation ($t = 1$) and after 1 ($t = 2$) and 2 ($t = 2$) training steps. The interesting behaviour highlighted in this figure is that for the SP model, the pre-activation weights blow up with width during training. For μP however, the L_1 of each layer stays flat with width even during the training, proving the correct parametrisation of the model and the "width-independent" scaling. This unstable behaviour of the SP parametrisation is easily highlighted thanks to the use of a large and fixed learning rate (here $lr = 10^{-2}$).

Theoretically, a μP model should deliver equal to better performance to an equivalent SP model when both have optimal hyperparameters. This behaviour is due to the maximal updating of the former, leading to optimal in-depth updates of all layers. The standard parametrisation does not implement this correct updating, with outter layers closer to the loss function having an opaque effect on the propagation of the update for the input layers proportionally to their widths. Scaling down the learning rate is not a sufficient modification to correc SP: as displayed in Figure 1.41, not all layers update incorrectly with some pre-activation sum staying flat across width. By updating all activation maximally indepentely of the width, μP should outperform SP for a tuned learning rate [63]. A significant advantage of this width independent effect is that the optimal learning rate for a μP architecture becomes width-independent. This leads to the μTransfer algorithm for HPO: one can search for the best hyperparameters on a μP model with fewer neurons per layers (smaller width) and transfer the found optimal to the full-size model at no extra cost (0-transfer) [62]. The benefit of adopting the maximal update parametrisation are:

1. Better performance of a μP model compared to an SP model for a tuned learning rate.

2. Improved hyperparameter optimisation with the μTransfer algorithm: performing the HPO scan on a smaller and easier to train model to 0-transfer the best set of hyperparameters to the full-size models.

3. Better hardware usage for HPO: a smaller model can be trained on a single GPU. This is of particular interest for the ATLAS Collaboration, as most of the GPU resource accessible is scattered on geographically distant computing tiers and not on single nodes.

4. Simplified architecture: in the μP, a wider model outperforms a smaller model if no over-training occurs. Therefore, the best learning rate hyperparameter has to be found ounce for all GN2 model of varying widths.

Hyperparameters that can be optimised with the μTransfer algorithms are said to be μTransferable. They consist of [62]:

- Learning rate and parameters of a learning rate scheduler.
- Optimiser parameters (momentum, Adam α and β).
- Initialisation parameters (initial per-layer variance).
- Multiplicative constants.

Many parameters unfortunately do not μTranfer as they combine aspects of the model and the data. They must be studied on the full size model directly. For example, the regularisation parameters (dropout, weight decay, normalisation, ...) do not scale, as a particular model size will overfit depending on the data. Finally, the last important family of hyperparameters are those defining the scale of the problem. These parameters are not found from μTransfer but rather "μTransfered along". They consist of the width (number of neurons per layer, number of attention heads in a transformer, ...), the depth, and the batchsize. Only the scaling along width is theoretically justified thanks to μP, while the others are empiraclly observed to hold [62].

Studies of the μP parametrisation and μTransfer algorithm were performed for the GN2 flavour tagger. In this architecture, the most relevant dimensions are the width and the depth of the transformer part, tasks with building a global conditional representation of the tracks from the embedded tracks processed by the initialiser network. These two dimensions are keys as most of the parameters of the GN2 model are in the transformer and the initialiser, with only few parameters set in the networks of the primary and auxiliary tasks. As such, the dimension scaled with μTransfer is the embedding width. The number of parameters in the tranformer roughly scales with the square of the embedding width due to the attention mechanism, making it the most sensitive parameter to define the complexity of GN2.

To demonstrate the effect of μP on GN2, a learning rate hyperparameter optimisation campaign targeting the initial and maximal value of the learning rate (the final value - LR end - was not modified and is kept at 10^{-5} for all test due to limited compute) is performed using the standard and maximal update parameterisation (SP vs μP). Three embedding widths are considered: the nominal 256 embedding width, defining a GN2 model with 2.3M parameters, a mid-size 128 embedding width (0.72M parameters), and a small 64 embedding width model with 0.23M parameters. Interestingly, this smaller model with an embedding 1/4 of the full model only has a 10th of the parameters. Furthermore, the small model was found to be trainable on a single GPU while the full and mid-size models required two GPU to be trained in a reasonable amount of time. All models are full GN2 models (with auxiliary tasks) trained on 30M PFlow

jets composed 60% $t\bar{t}$ and 40% Z' for 40 epochs with batchsize 1024. All parameters not mentioned are kept similar between embedding widths and parametrisation, and the epoch giving the lowest validation loss is picked for each model. Figure 1.42 displays the main result from this campaign, displaying the various LR max considered at the best LR initial found (10^{-5}). Three main observations can be drawn from analysing the result:

1. With μP, the wider GN2 model - larger embedding width - outperforms the smaller version.

2. Wider models do not always outperform smaller model with SP. In particular at large LR max the wider model becomes unstable and its performnce in terms of validation loss significantly decreases.

3. The optimal LR max (and LR init as shown in Figure 1.43) are shared across width with μP, while no such behaviour is guaranteed for SP - but is observed in the present case.

The full LR init vs LR max scans can be found in Figure 1.43 for SP and μP. Changing the LR init has little effect on the reached performance, due to the LR scheduler quickly moving away from the initial value and the common LR end value of 10^{-5} shared by all model at the end of training. The LR max however is a significant hyperparameter having a large impact on performance. All SP models with 256 embedding width are found to become unstable at large value of max LR. Note that the scan at LR initial = 10^{-5} benefitted from more test to capture the sudden rise in validation loss at larger LR max. As expected from the previous discussion, all μP models stay stable, even at larger value of the learning rate. On the contrary, SP models become unstable with large LR max. μP models share the same optimal LR parameters, although some variance impacts the precision of the method on the smallest model. Due to limited computing power available, only one seed was run per-test, introducing some unmeasurable statistical variance in the output. An essential conclusion is this respect is the computing gain from performing the HPO on the smaller width model then the full width one:

- The full width model (embedding size 256) has 2.3M parameters, taking ∼39 min per epoch on 2 A100 GPUs each fed data by 20 CPUs.

- The small width model (embedding size 64) has 0.23M parameters, taking ∼20 min per epoch on 1 A100 GPU fed data by 20 CPUs.

Essentially, a single full width model hyperparameter test is in computing terms equivalent to running 4 individual tests on the smaller model. Given a fixed computing budget, one can therefore have a far better coverage of the hyperparameter search space with μTransfer.

This optimisation study was carried out to demonstrate the benefits of μP on GN2. Interestingly, the optimal value found for both the μP and SP models is at an LR max $= 5 \times 10^{-4}$ and LR initial $= 10^{-5}$. The default values used in the prior training of GN2 were, by luck, the same LR max but a larger LR init of 10^{-7}. To quantify the effect on performance, the b-efficiency versus c- and light-rejection on $t\bar{t}$ and Z' of two μP models are displayed in Figure 1.44, with the suboptimal one being the worst performing full-width model (LR max $= 5 \times 10^{-5}$, LR init $= 10^{-7}$) and the optimal one the best performing one (LR max $= 5 \times 10^{-5}$, LR init $= 10^{-5}$). While the optimal and suboptimal models had close validation loss, respectively 0.601 and 0.591, a significant difference in background rejection at all efficiency is observed. At a b-tagging working point of 70%, the suboptimal GN2 model underperforms the optimal one on $t\bar{t}$ by 18% (14%) on c-rejection (light-rejection) and the disparity is even higher on Z', rising to 24% (26%) at a b-tagging working point of 30% - which is equivalent to the 30% WP on $t\bar{t}$.

More preliminary tests of μP were performed with GN2, showing good scaling across depth as expected from empirical results [62]. Due to limited computing power available, the study of SP versus μP only encompassed two hyperparameters: the initial and maximal learning rate. The validity of the method has been confirmed and future studies optimising all of the learning rate scheduler hyperparameter (including the warm-up and the learning rate at the end) will be carried out. Other hyperparameters that can best optimised with μTransfer are the initialisation variances of the different layers and the auxiliary objectives individual weights of Equation 1.10. In summary, the present work introduces two approaches that are combined to deliver an improved hyperparameter optimisation:

- Carrying out the HPO on Kubeflow with the Katib workload to benefit from state-of-the-art autoML algorithm.
- Leveraging the μP parametrisation to increase the performance of the tuned GN2 and benefit from the factor 4 boost in hyperparameter test coverage from μTransfer.

The full optimisation of GN2 is, at the time of writing, an ongoing effort of the ATLAS Collaboration.

1.3.4 GN2X: a GN2 variant for Boosted Higgs Bosons Decay to Heavy Flavours

A final aspect of the GN2 model presented in this thesis is an application of the architecture to a specialised objective: identifying boosted Higgs boson decaying into a pair of b- or c-quarks. Having an effective tagger to identify these boosted decays can significantly help analyses studying the decay of Higgs particles to a $c\bar{c}$ pair [66], for the precise measurement of the Higgs boson p_T spectrum [67], and for beyond the SM measurements [68]. To perform this task, a new algorithm labelled GN2X is introduced based on the design of GN2 [69]. Its main task is to discriminate jets from boosted Higgs boson decaying into a $b\bar{b}$ or a $c\bar{c}$ pair from those originating from the fully-hadronic top-quark decay and the multijet processes. While other taggers presented in this chapter relied on small-radius ($R = 0.4$) PFlow jets or VR jets, GN2X is trained on jet reconstructed with a large-radius ($R = 1.0$) with Unified Flow Object (UFO) objects to capture the majority of the decay products [70]. UFO combines PFlow [71] and Track-Calo clusters objects [72], thereby including neutral and charged components in the reconstruction. UFO large-R jets are reconstructed with the anti-k_T algorithm with a radius $R = 1.0$ [73].

To train the algorithm, Higgs produced in association with a Z boson and decaying to a pair of heavy flavour quarks ($b\bar{b}$ or a $c\bar{c}$) are simulated. To not bias the result towards a specific p_T, η, and mass distributions of the jets, the simulation are biased through sampling to have an approximately flat distribution of jet mass in the training set, while the validation set follow the SM ZH prodcution for a Higgs boson H of a mass equal to 125 GeV. Similarly, the top-quark decay with subsequent hadronic decay of the W boson in the $t \to bW$ chain is simulated for the training sample using a hypothetical Z' boson of 4 TeV mass decaying as $Z' \to t\bar{(t)}$ with approximately flat jet p_T distribution. The evaluation sample uses the SM $t\bar{t}$ decay with filters on the scalar sum of the objects p_T in the event. Finally, the multijet process is simulated in slices of particle-level jet p_T to have the same spectrum. More details on the simulated samples used can be found in Ref [69]. After resampling the samples to ahve similar p_T, η, and mass distributions, there are 62 million jets split between 15 million $H_{b\bar{b}}$, 15 million $H_{c\bar{c}}$, 10 million top, and 22 million multijets.

The previous algorithm for this task that now serves as benchmark in this study is the X_{bb} tagger, a feed-forward network combining the flavour tagging discriminants of DL1r or DL1d for up to three VR subjets associated to the large-R jet [74, 75]. The track selection is similar to that

of the GN-models (Section 1.3), and the inputs of the model are equivalent to those of Table 1.6, with the jet variables defined on the large-R jet and an additional jet variable being used: the mass of the large-R jet. At most 100 tracks associated with a jet are supplied to the network, as sorted by the decreasing transverse impact parameter significance S_{d_0}. The same auxiliary tasks as in GN2 are used with the same respective weights and neural network designs. The initialiser has an 192 embedding dimension and the transformer encoder combines 6 layers with 4 attention heads. The global representation is again obtained from a weighted sumer over the conditional tracks, with learnable attention weights. GN2X contains in total 1.5 million parameters, and is trained on 4 A100 GPUs for 40 epochs (∼1 hour per epoch) with a batchsize of 1000.

To discriminate, the model outputs four probabilities $p_{H_{b\bar{b}}}$, $p_{H_{c\bar{c}}}$, p_{top}, and p_{QCD} that are combined in a discriminant score equivalent to Equations 1.1 and 1.2:

$$D_{H_{b\bar{b}}} = \log \frac{p_{H_{b\bar{b}}}}{f_{H_{c\bar{c}}} \cdot p_{H_{c\bar{c}}} + f_{\text{top}} \cdot p_{\text{top}} + (1 - f_{H_{c\bar{c}}} - f_{\text{top}}) \cdot p_{\text{QCD}}}, \tag{1.11}$$

where the flavour fractions were chosen from dedicated performance studies to be $f_{H_{c\bar{c}}} = 0.02$ and $f_{\text{top}} = 0.25$. A discriminant for $H_{b\bar{b}}$ is similarly defined:

$$D_{H_{c\bar{c}}} = \log \frac{p_{H_{c\bar{c}}}}{f_{H_{b\bar{b}}} \cdot p_{H_{b\bar{b}}} + f_{\text{top}} \cdot p_{\text{top}} + (1 - f_{H_{b\bar{b}}} - f_{\text{top}}) \cdot p_{\text{QCD}}}, \tag{1.12}$$

with $f_{H_{b\bar{b}}} = 0.3$ and $f_{\text{top}} = 0.25$. The performance of GN2X can be assessed from the ROC curves presented in Figure 1.45. An additional performance to X_{bb} and GN2X is presented, where two individual VR subjets are b- or c-tagged by a VR-trained GN2 model. The jets used are the leading VR subjets associated to the large-R jet. Note that X_{bb} was not retrained on the specific samples.

A clear performance gained is delivered by the GN2X method above both the X_{bb} tagger and the combination of two individual tags with GN2. The latter approach does not access correlations between the subjets, explaining its lower performance at higher $H(b\bar{b})$ and $H(c\bar{c})$ efficiencies than the GN2X and X_{bb} model. At a 50% $H(b\bar{b})$ WP, GN2X improves the top rejection (multijet rejection) on X_{bb} by a factor 1.6 (2.5) [69]. For $H(b\bar{b})$ tagging, the $H(c\bar{c})$ background is negligible. GN2X also improves the performance for $H(c\bar{c})$ tagging over the approach combining two individual VR tagged-jets: at a 50% WP, GN2X improves the top rejection by a factor 3, the multijet rejection by a fator 5, and the $H(b\bar{b})$ rejection by a factor 6. This novel approach

to perform boosted object tagging is the first of its kind in ATLAS and is now integrated in the ATLAS software.

1.4 Calibration

All flavour taggers presented in this chapter have been trained on MC simulations, as described in Section 1.1.3. As such, they depend on and acquire specific features of the simulated data that might not be present in the real data collected by the ATLAS experiment. While the Collaboration aims to generate the highest-fidelity simulation possible with advance software build on GEANT4 [32] and many other specialised framework, inherent and unavoidable differences are left. To quantify the effect of using a simulation-trained network on real data, the ATLAS Collaboration performs Data-Monte Carlo agreement and calibration studies. These are performed in two steps:

- Data-MC Scale Factor (SF) are derived, comparing the output of the tagger on a simulated and real dataset with equivalent selection [4, 76, 77, 78]. The efficiencies ϵ^f for each flavour $f \in b, c,$ light are measured, both on the simulated and real dataset, with

$$\epsilon^f(p_T) = \frac{N^f_{\text{tagged}}(p_T)}{N^f_{\text{all}}(p_T)},$$

  where $N^f_{\text{tagged}}(p_T)$ is the number of jet of flavour f in the bin of p_T that are b-tagged and N^f_{all} the total number of jet of flavour f in the same bin. A scale factor is then derived for each flavour f as

$$\text{SF}^f_{\text{Data-MC}}(p_T) = \frac{\epsilon^f_{\text{Data}}(p_T)}{\epsilon^f_{\text{MC}}(p_T)},$$

  giving the ratio of the measured efficiency in data over simulation. To include dynamic effect of the taggers, the efficiencies ϵ^f and SF are derived in bins of jet p_T. Such calibration factors correct the efficiencies of tagging and misstaging and are applied to all analyses using the flavour tagger. This calibration is performed independently for each output flavour of the tagger, as it relies on selecting a portion of the ATLAS data with a large proportion of the studied flavour. The b-tagging efficiency is derived from a sample of $t\bar{t}$ with two charged leptons in the final state, as described in Ref [4]. The SF for c-jet misstagging is calibrated on a $t\bar{t}$ sample decaying to exactly one charged lepton and several jets [78]. Finally, the SF for light-jets is derived in a sample of Z bosons produced in association with jets (Z+jets) [79]. Due to the extreme rejection power of modern flavour tagger, a special

technique so-called "flip tagger" is used for this last SF in which the tagger is modified to have a reduced light-rejection.

- MC-MC SF are then derived between the chosen Monte Carlo simulator for the training and other simulators or by changing the tuning [80]. This dependency is measured by applying the same tagger to samples simulated with different generators, mainly Pythia [25], Herwig [81], and Sherpa [82] for variation to the parton shower and hadronisation and MadGraph for variation of the matrix element [83]. The decay chains of b- and c-hadrons in ATLAS is further simulated with the EvtGen package [30]. These effects are measured into SF using the same technique as the data-MC scale factors. For an alternative generator, the SF of flavour f is derived by composing the Data-MC SF with the nominal sample and the MC-MC SF as:

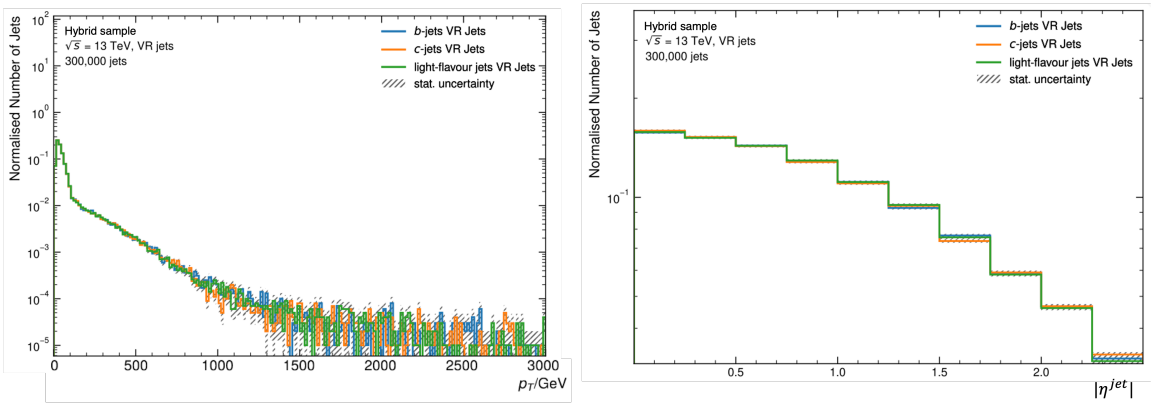
$$\mathrm{SF}^{f}_{\text{Alternative}}(p_T) = \frac{\epsilon^{f}_{\text{Data}}(p_T)}{\epsilon^{f}_{\text{Nominal MC}}(p_T)} \times \frac{\epsilon^{f}_{\text{Nominal MC}}(p_T)}{\epsilon^{f}_{\text{Alternative MC}}(p_T)} = \frac{SF^{f}_{\text{data-MC}}(p_T)}{SF^{f}_{\text{MC-MC}}(p_T)}.$$

These scale factors are applied in physics analyses as a per-jet weight. Some early studies of both scale factor types have been performed in Ref [17], showing good agreement between the data and simulated performance of DL1d and GN1. Variations due to the change of generator are also found to be at most of 8% with respect to the nominal choice.

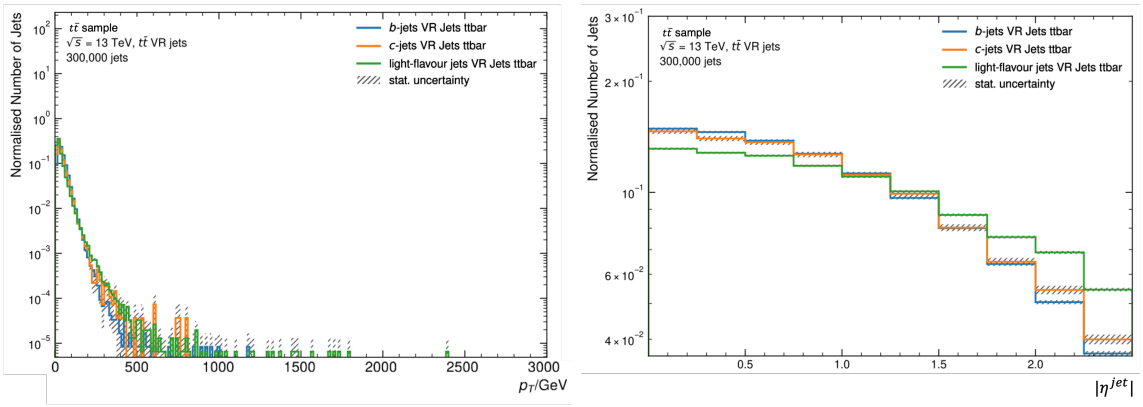
1.5 Conclusion

This chapter introduces the main machine learning models developped for heavy-flavour jets identification in ATLAS during the period covering 2020 to 2024. Work carried out in and presented in this thesis includes the first training of the DL1d model, including the DIPS sub-tagger of the first time in the ATLAS software. DL1d is found to have improved background rejections at a fixed working point for both b- and c-tagging compared to the at-the-time released tagger: DL1r. Significant changes in addition to the development of this new tagger were made to the preprocessing pipeline of the Umami framework [43] and the architecture as well as the list of features used. Finally, the new family of taggers based on graph neural network for GN1 and transformers for GN2 is introduced, with the architecture adopted fully introduced as well as a presentation and comparison of the results obtained by the new methods. A discussion on the hyperparameter optimisation of GN2 is also included, introducing the possibilities of using a new infrastructure of the Kubeflow server managed by CERN as well as the relevance of the maximal update parametrisation for improving the search for optimal hyperparameters of GN2.

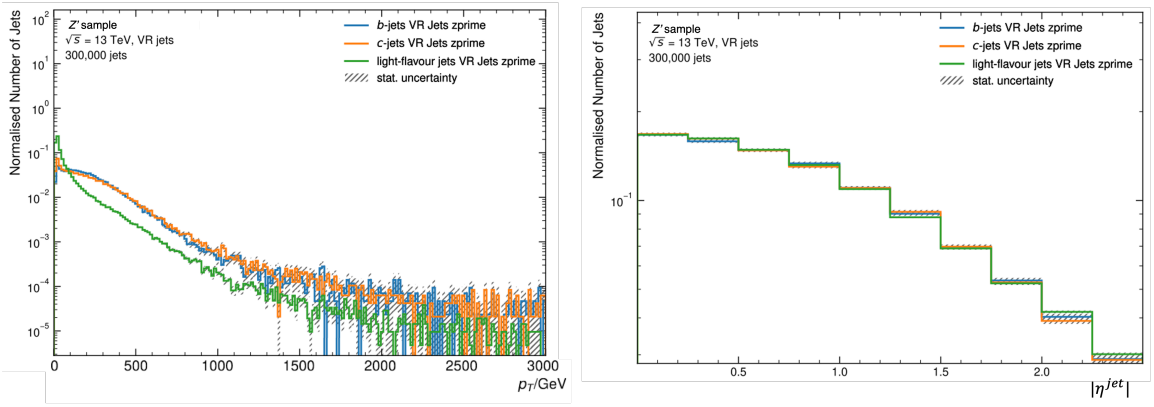
Significant contributions were made to the development of the Salt framework used to train GN-typed model were made to support these studies [47].



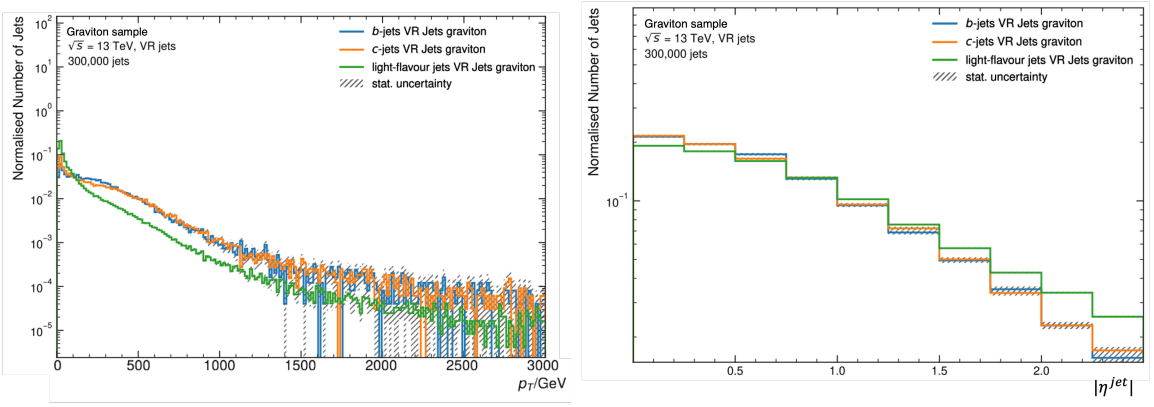
(a) Hybrid sample.



(b) $t\bar{t}$ sample.

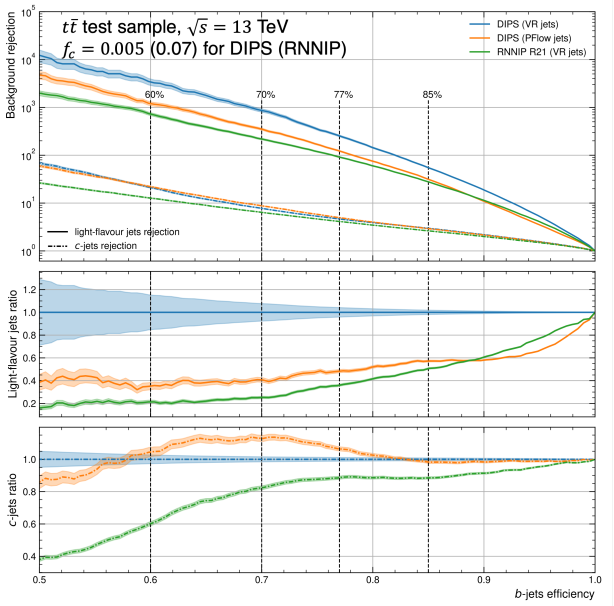


(c) Z' sample.

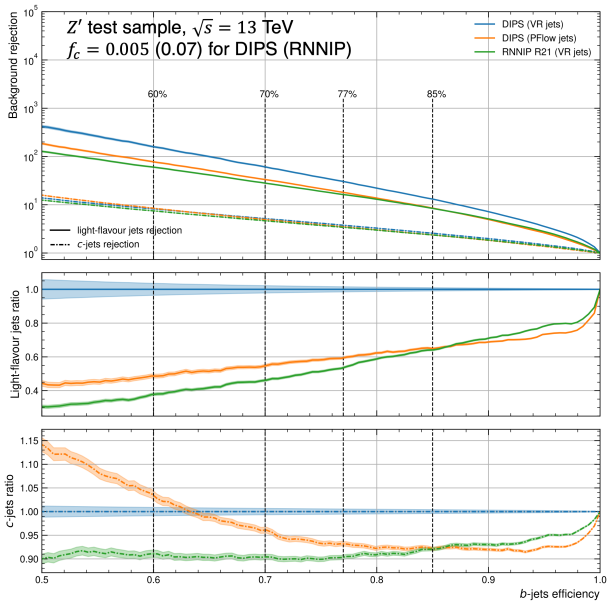


(d) Graviton sample.

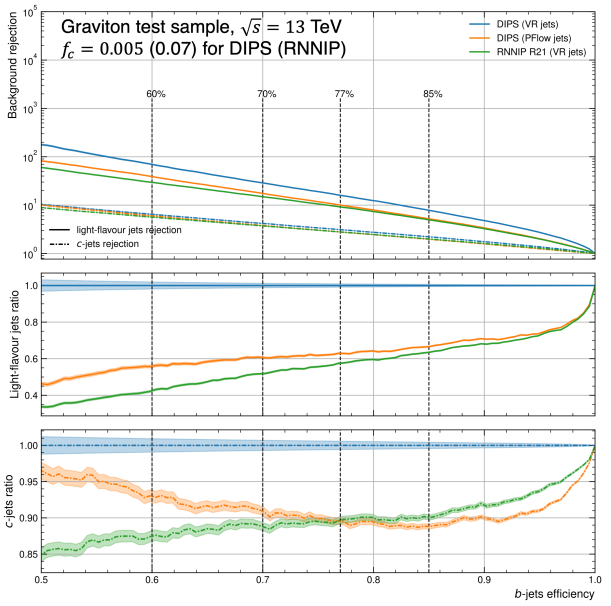
Figure 1.9: Distributions for the VR-jet training of jets p_T (left) and $|\eta|$ (right) for the hybrid combined process (top row) made from the three bottom processes, in the order $t\bar{t}$, Z', and the graviton.



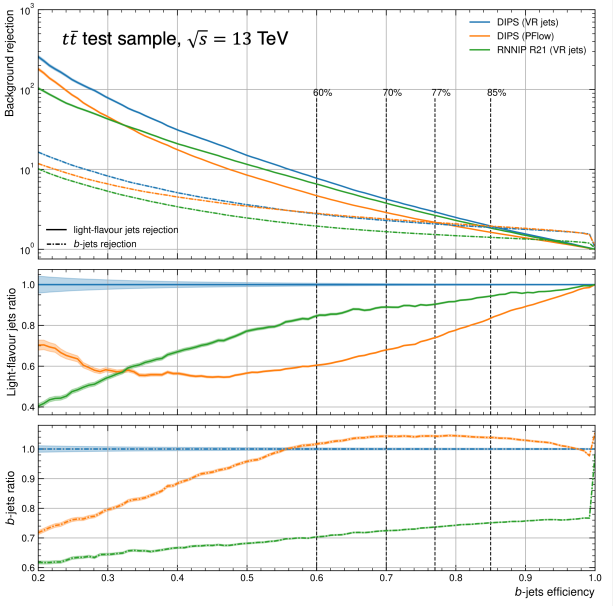
(a) $t\bar{t}$ test sample b-tagging.



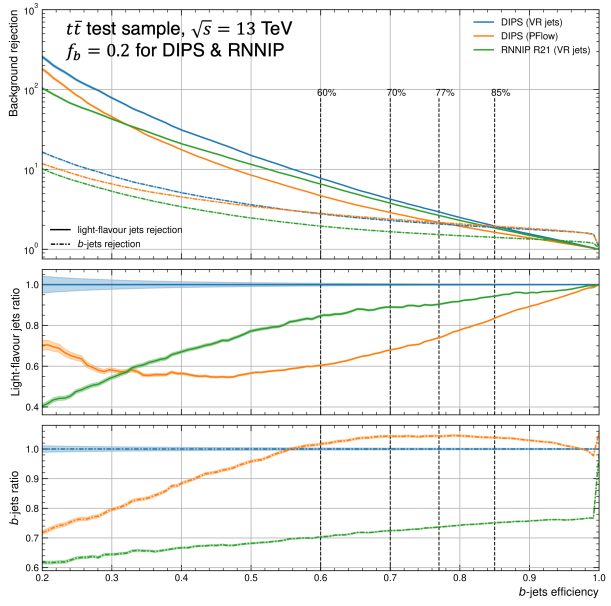
(b) Z' test sample b-tagging.



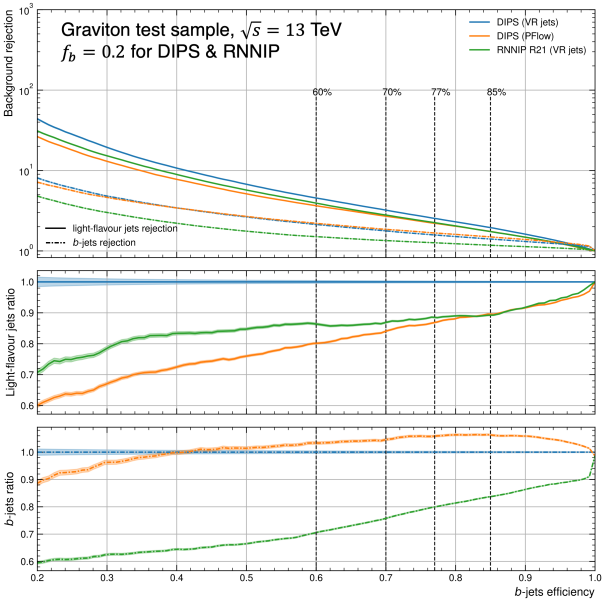
(c) Graviton test sample b-tagging.



(d) $t\bar{t}$ test sample c-tagging.



(e) Z' test sample c-tagging.



(f) Graviton test sample c-tagging.

Figure 1.10: ROC curves for b-tagging (top) and c-tagging (bottom) on test samples of 300 k jets for $t\bar{t}$ (left), Z' (centre), and the graviton process (right). Models are displayed as curves of different colours, with the VR-jets DIPS in blue, the DIPS model trained on PFlow jets in orange, and RNNIP trained on VR-jets from the previous software release R21 in green.

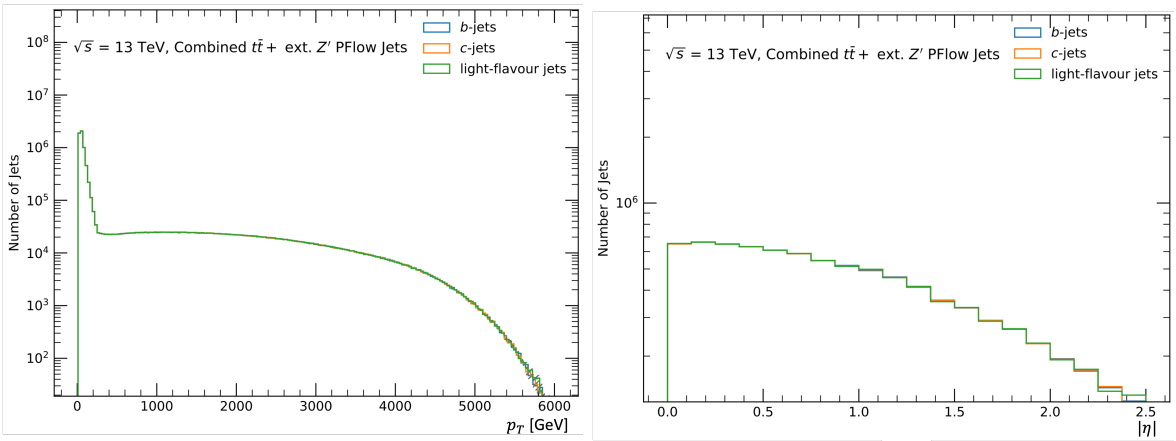


Figure 1.11: The p_T (left - in MeV) and $|\eta|$ distributions of the resampled b-, c-, and light-jets in, respectively, blue, orange, and green. The three ses are resampled to have the same $p_T - |\eta|$ 2D distributions. The flat p_T spectrum extending up to several TeV is due to the exotic Z' process generated with varying mass, starting at 150 GeV. The large peak at lower p_T is the $t\bar{t}$-process. These sets have 8.3 million jets per flavour.

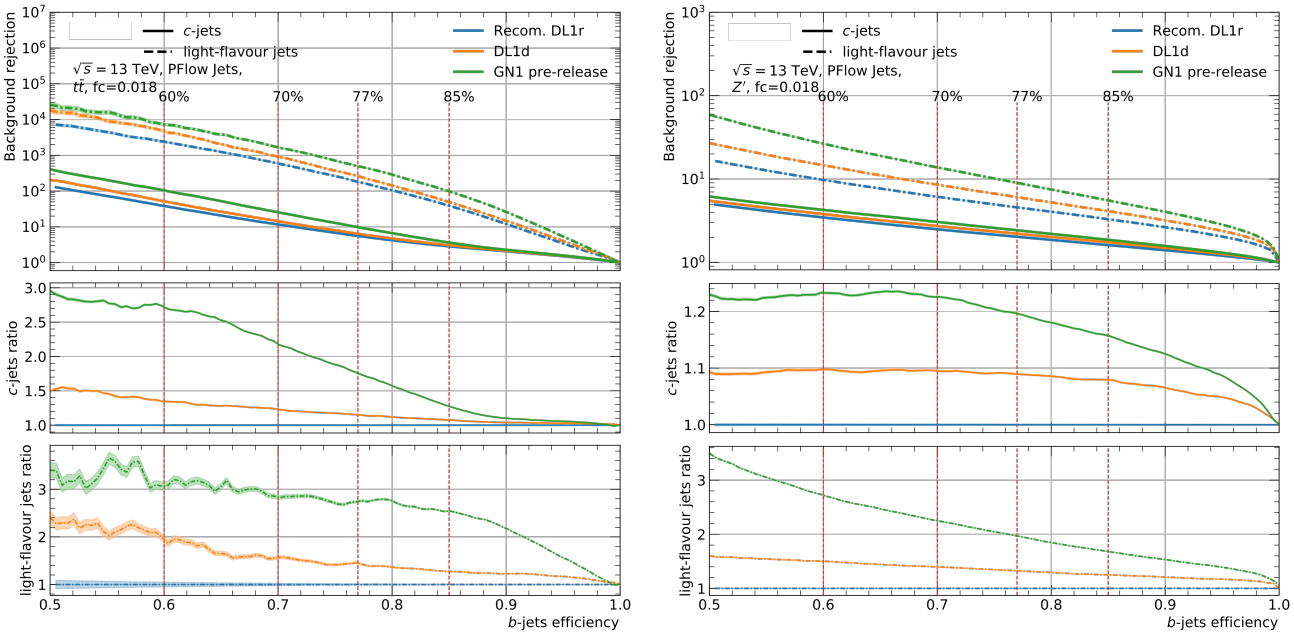


Figure 1.12: Performance for b-tagging with a flavour fraction of $f_c^b = 0.018$. Left: $t\bar{t}$; right: Z'. Top: ROC curves; centre: ratio of c-jets rejection from b-jets relative to the R22-retrained DL1r; bottom: same ratio for light-jets rejection. List of taggers: recommended DL1r from the previous release; DL1d trained on the new release; GN1 test-model trained on the new release.

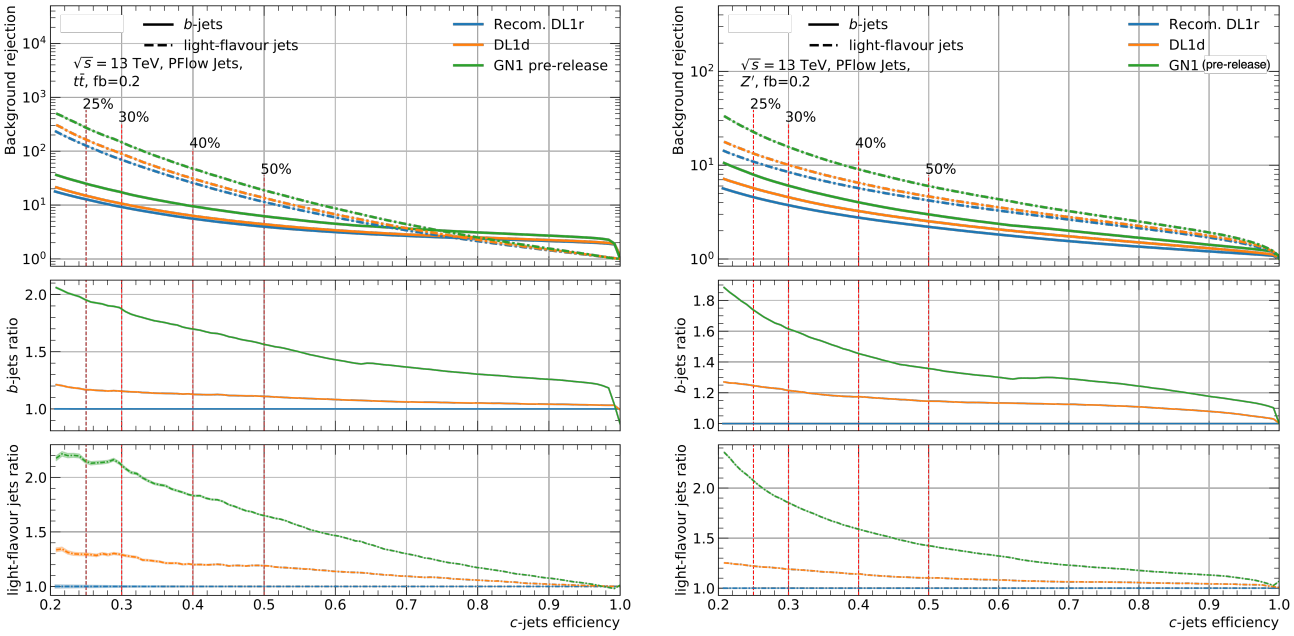


Figure 1.13: Performance for c-tagging with a flavour fraction of $f_b^c = 0.2$. Left: $t\bar{t}$; right: Z'. Top: ROC curves; centre: ratio of b-jets rejection from c-jets relative to the R22-retrained DL1r; bottom: same ratio for light-jets rejection. List of taggers: recommended DL1r from the previous release; DL1d trained on the new release; GN1 test-model trained on the new release.

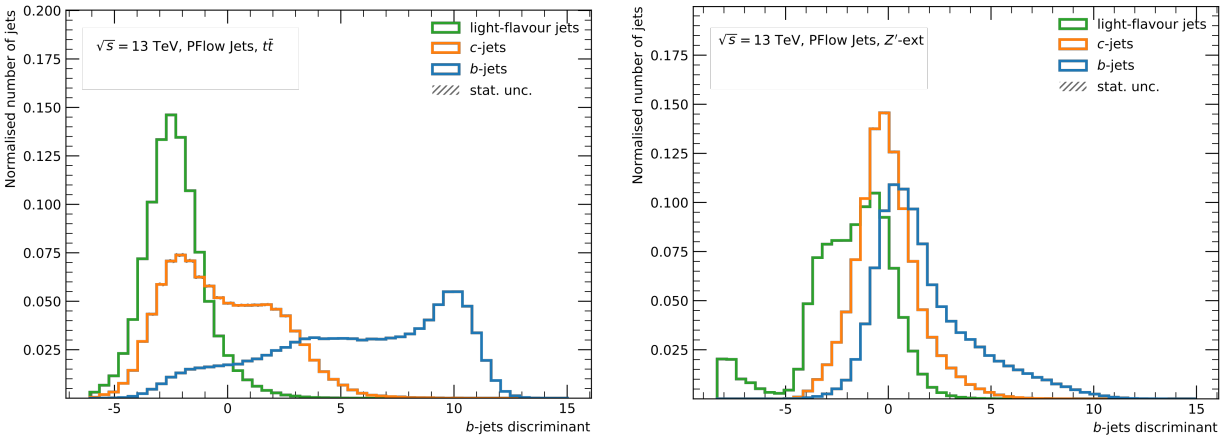


Figure 1.14: Distribution of DL1d b-tagging discriminant with $f_c = 0.018$ for the different jet flavours, evaluated on $t\bar{t}$ (left) and Z' (right).

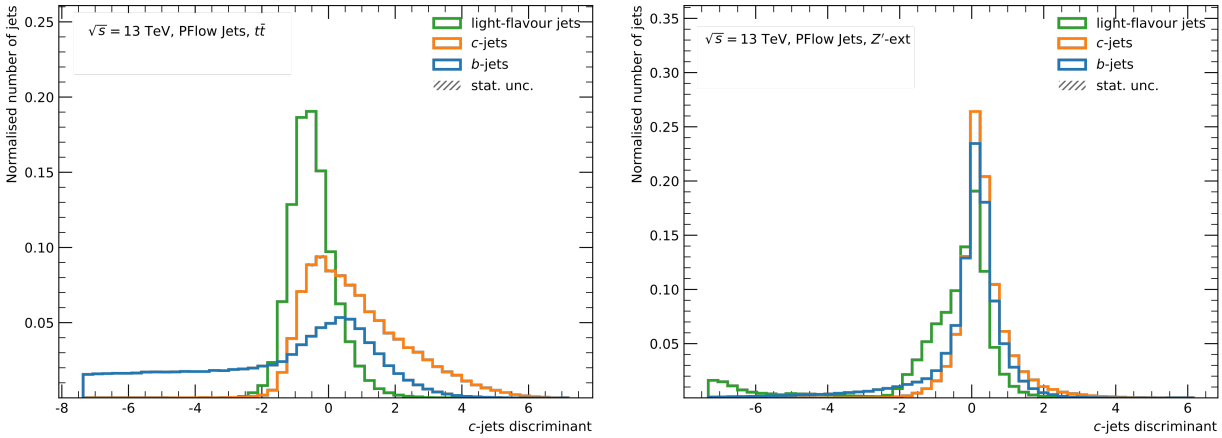


Figure 1.15: Distribution of DL1d c-tagging discriminant with $f_b = 0.2$ for the different jet flavours, evaluated on $t\bar{t}$ (left) and Z' (right).

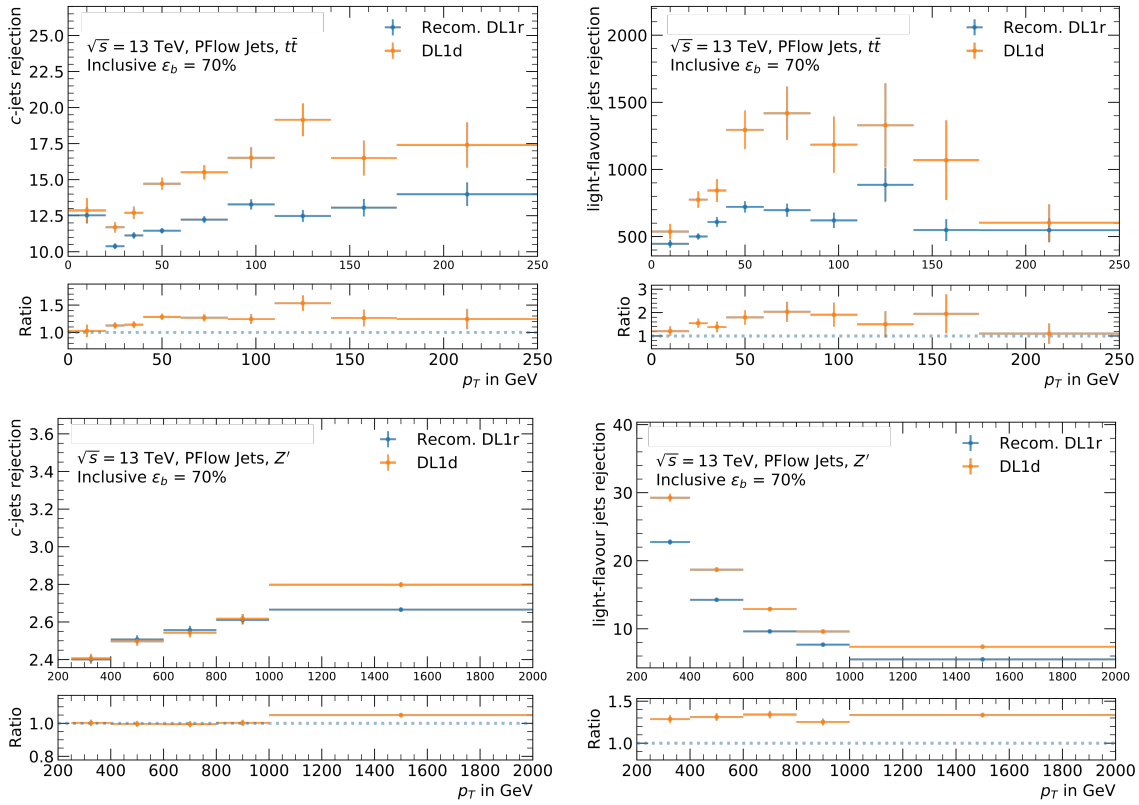


Figure 1.16: Background flavour rejections at a fixed b-tagging efficiency of 70% (per region shown) for the various taggers. Top: $t\bar{t}$; bottom: Z'; left: c-rejection; right: light-rejection. For each plot, the bottom panel presents the ratio to the recommended DL1r.

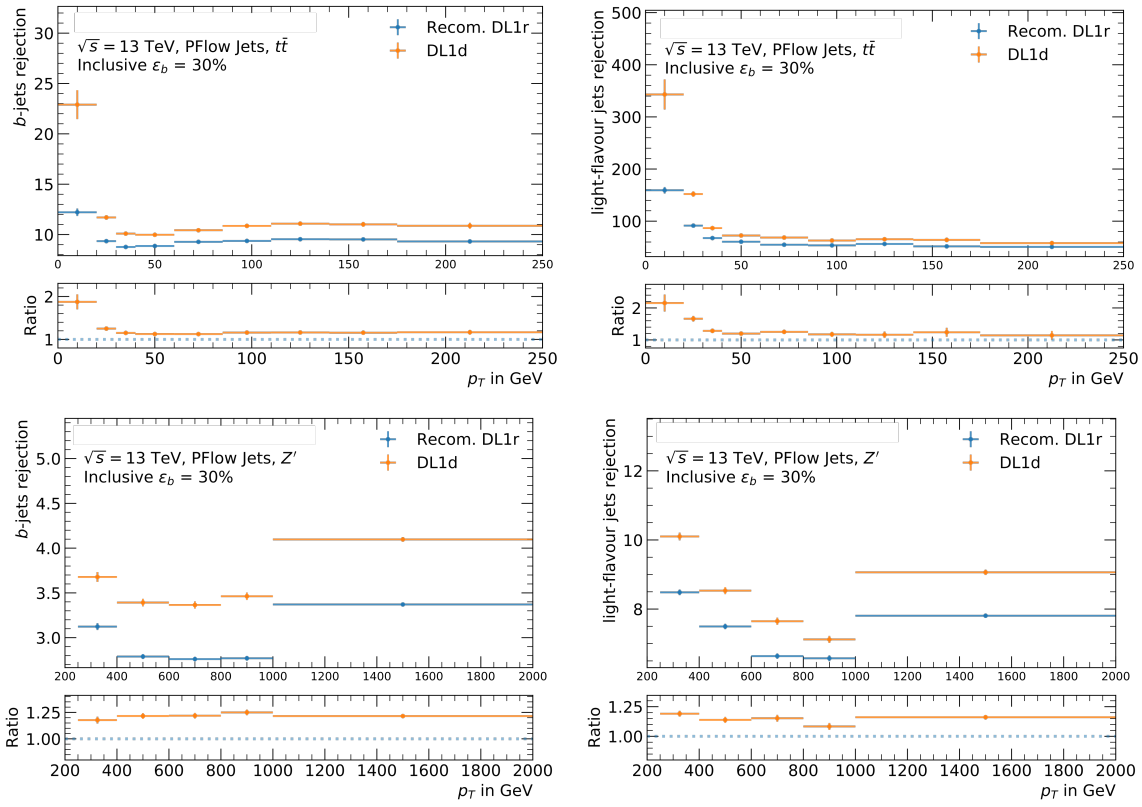
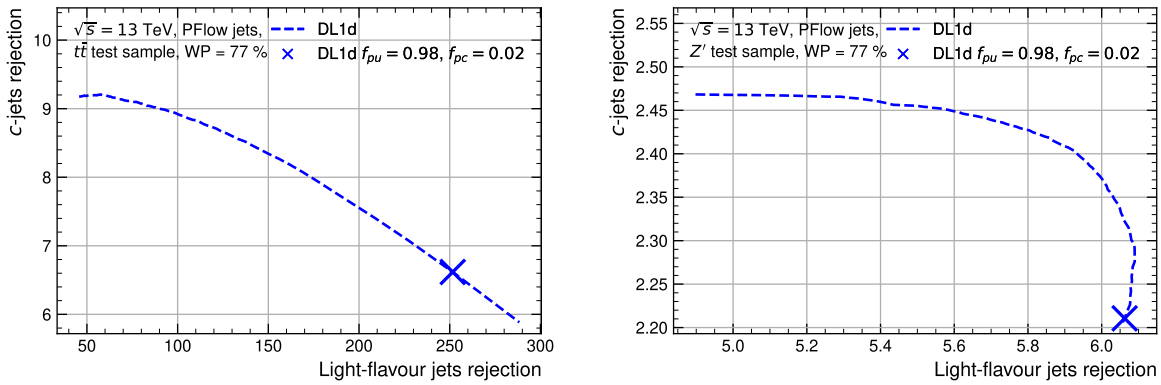
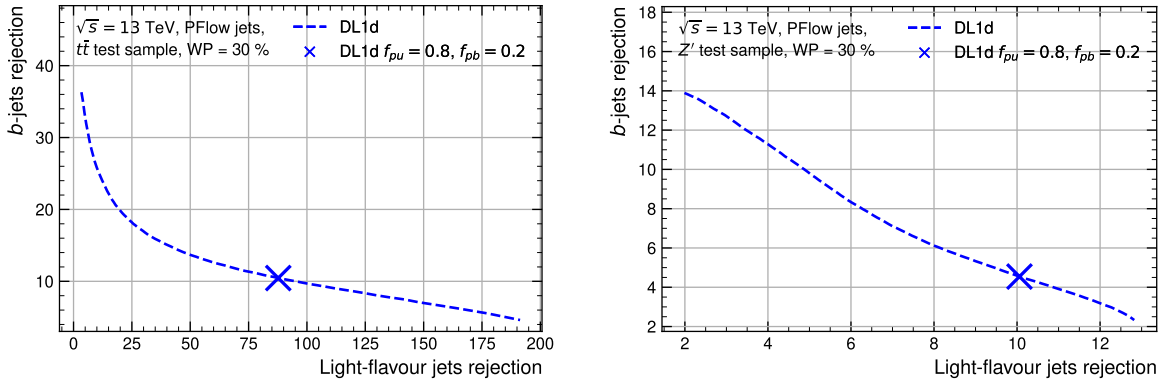


Figure 1.17: Background flavour rejections at a fixed c-tagging efficiency of 30% (per region shown) for the various taggers. Top: $t\bar{t}$; bottom: Z'; left: b-rejection; right: light-rejection. For each plot, the bottom panel presents the ratio to the recommended DL1r.



(a) Flavour fraction f_c^b for b-tagging scan: left is $t\bar{t}$ and right Z' test samples.



(b) Flavour fraction f_b^c for c-tagging scan: left is $t\bar{t}$ and right Z' test samples.

Figure 1.18: The flavour fraction scans of the DL1d model. The chosen values are marked on the curves, displaying on the y-axis the c-rejection (b-rejection) for b-tagging (c-tagging) vs the light-rejection on the x axis at a fixed working point of 77% (33%). Increasing f_c or f_b shifts the marker upwards along the curves.

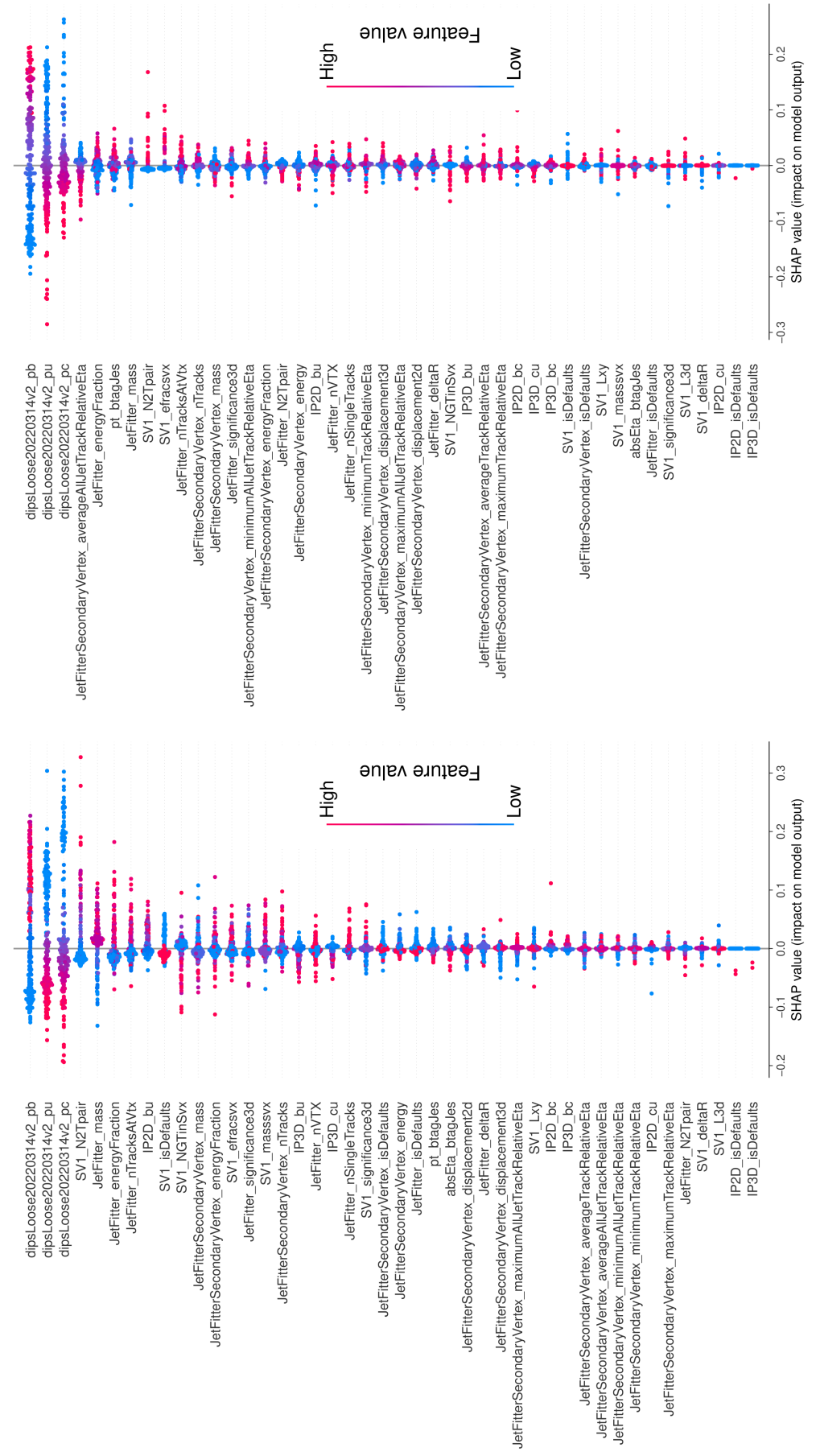


Figure 1.19: Shapley values of the different inputs variables of DL1d for b-tagging, $t\bar{t}$ on the left and Z' on the right.

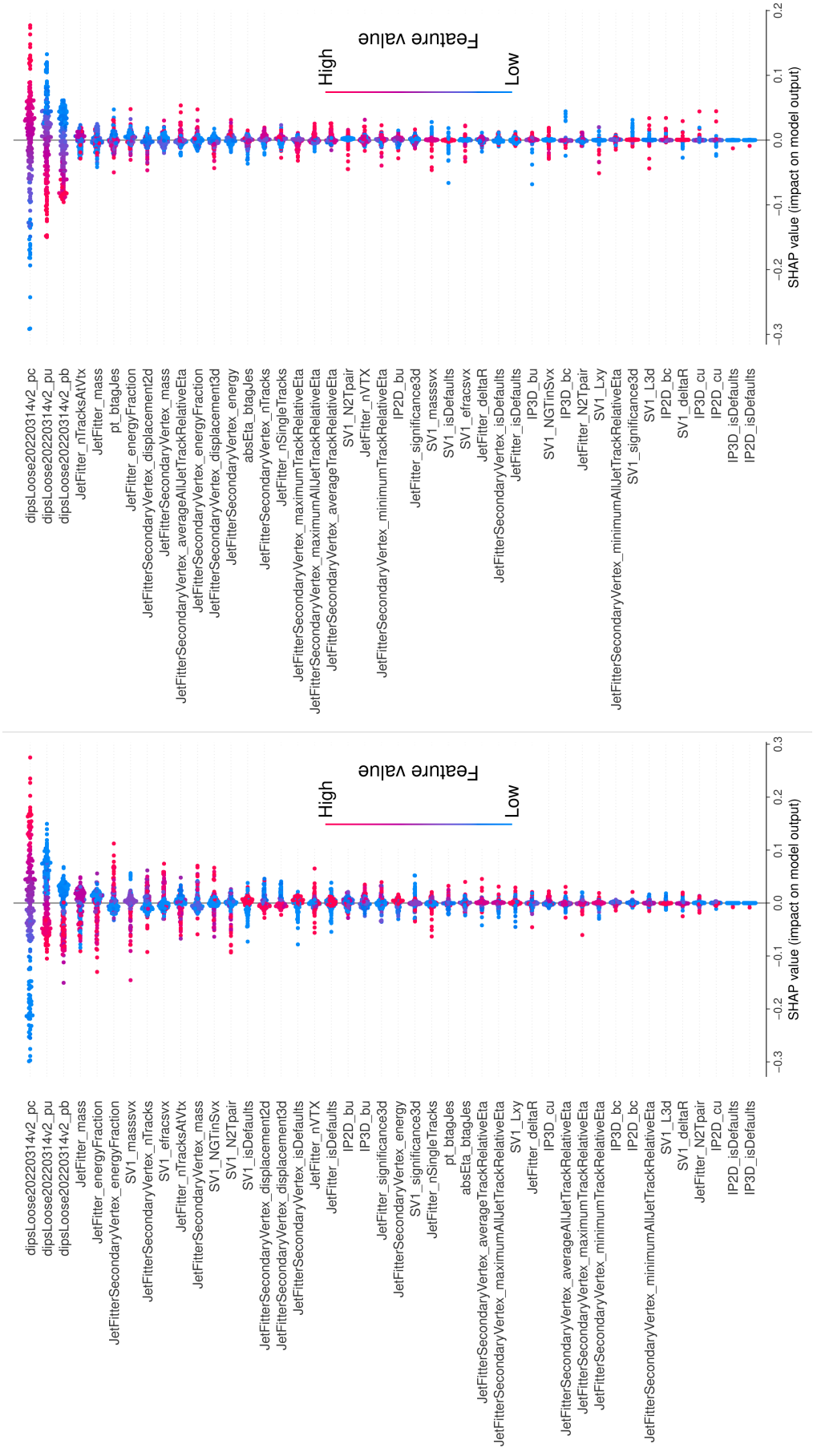
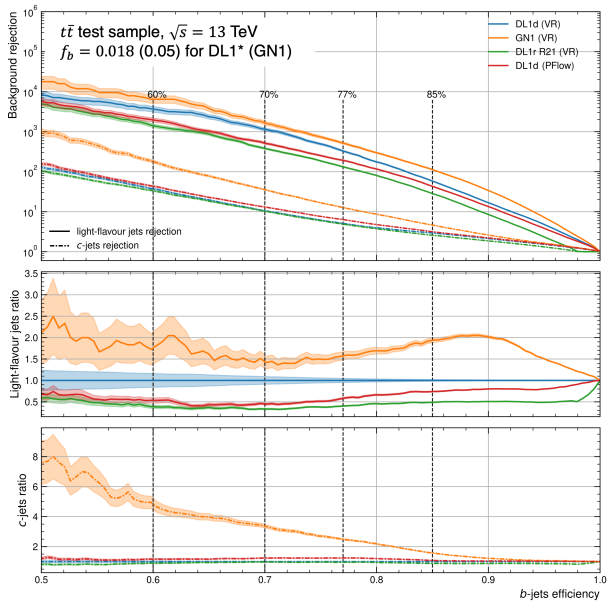
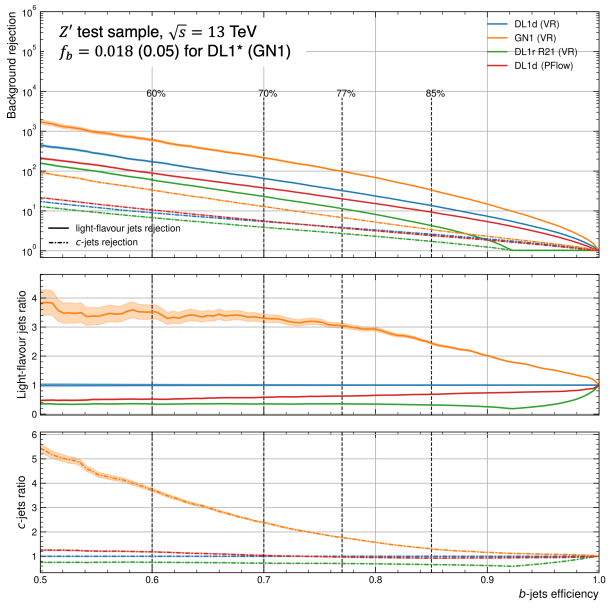


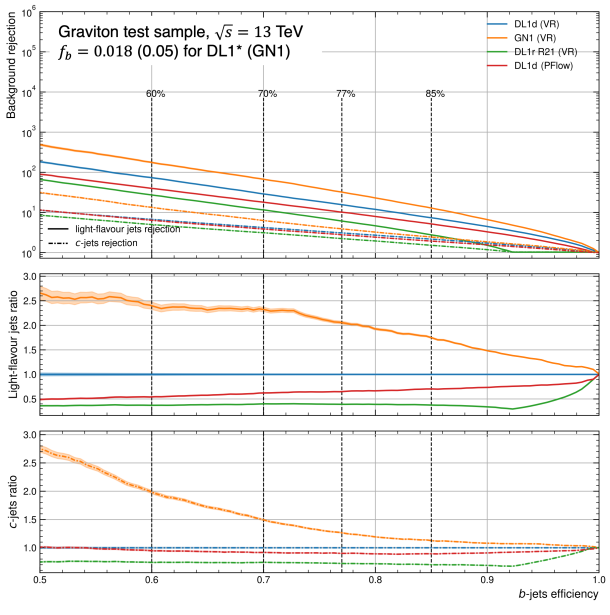
Figure 1.20: Shapley values of the different inputs variables of DL1d for c-tagging, $t\bar{t}$ on the left and Z' on the right.



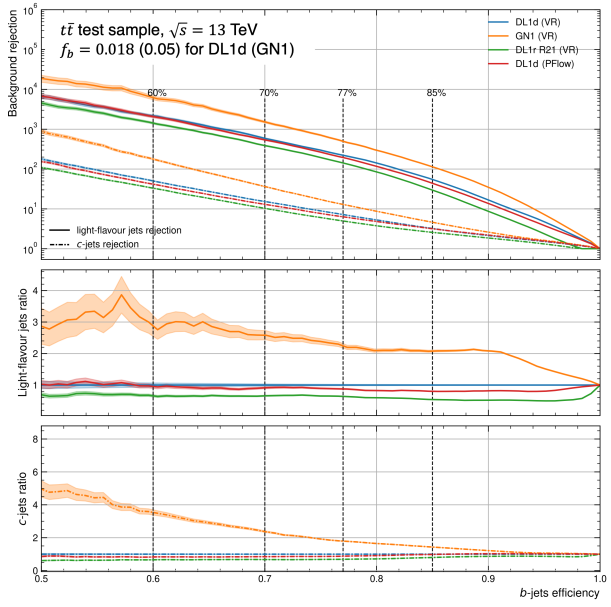
(a) $t\bar{t}$ test sample b-tagging, $f_c = 0.018$ for DL1d.



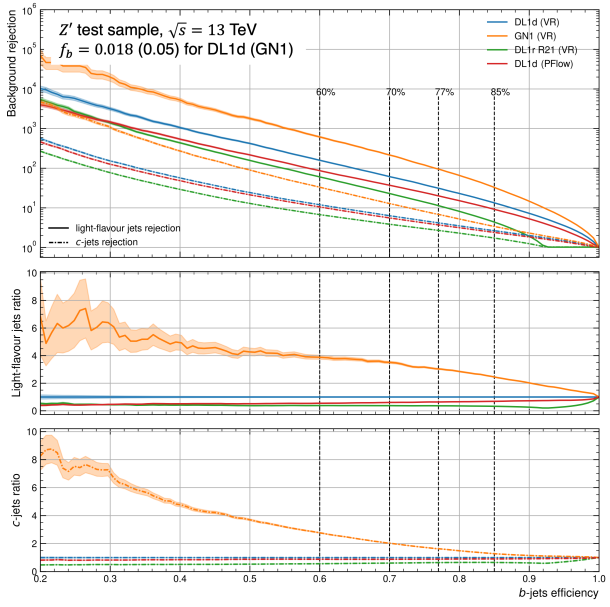
(b) Z' test sample b-tagging, $f_c = 0.018$ for DL1d.



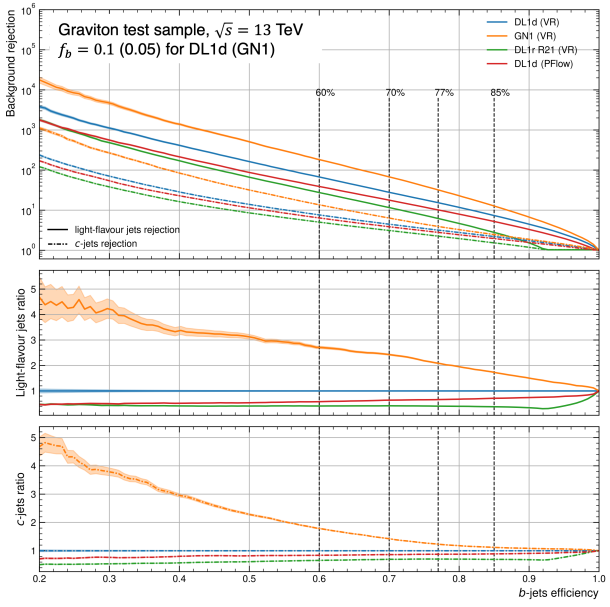
(c) Graviton sample b-tagging, $f_c = 0.018$ for DL1d.



(d) $t\bar{t}$ test sample b-tagging, $f_c = 0.1$ for DL1d.



(e) Z' test sample b-tagging, $f_c = 0.1$ for DL1d.



(f) Graviton sample b-tagging, $f_c = 0.1$ for DL1d.

Figure 1.21: ROC curves for b-tagging for $t\bar{t}$ (left), Z' (centre), and graviton (right)processes. Top row uses $f_c = 0.018$ for DL1d, while bottom row is $f_c = 0.1$ (GN1 $f_c = 0.05$ everywhere). Models are displayed as curves of different colours, with the VR-jets DL1d in blue, a pre-release VR-trained GN1 on 20 million in orange, DL1r trained on VR-jets with the previous software release R21 in green, and the PFlow trained DL1d in red.

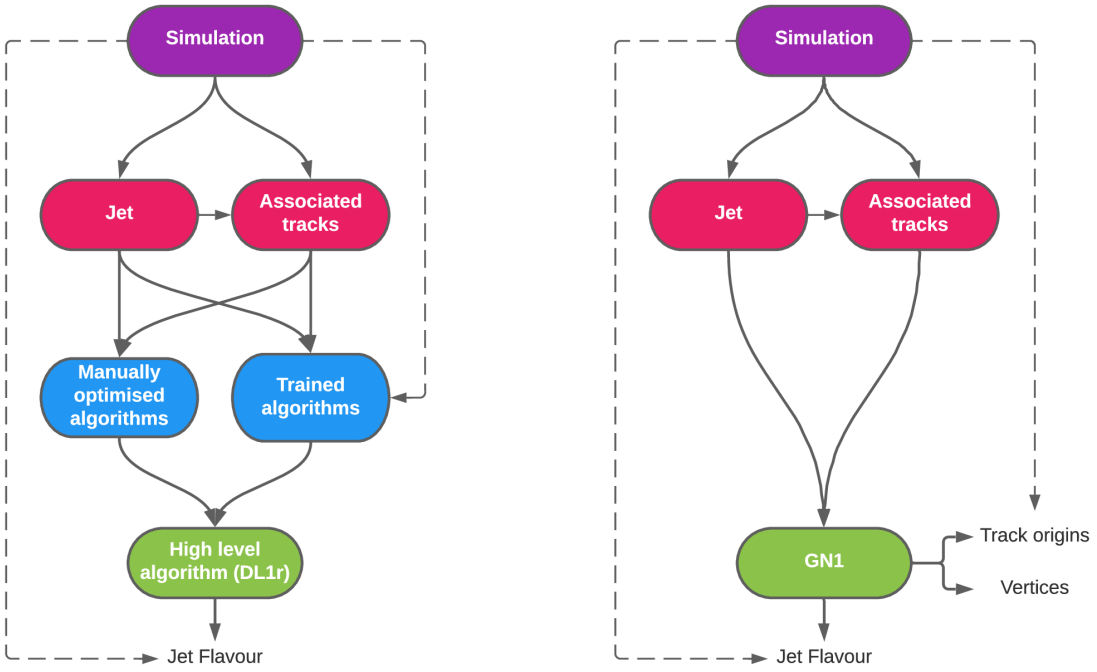


Figure 1.22: Comparison of the tagging scheme between the DL1 family (left) and the GN family (right), from [16]. Solid lines represent reconstructed information while dashed lines represent truth information only accessible from the simulations.

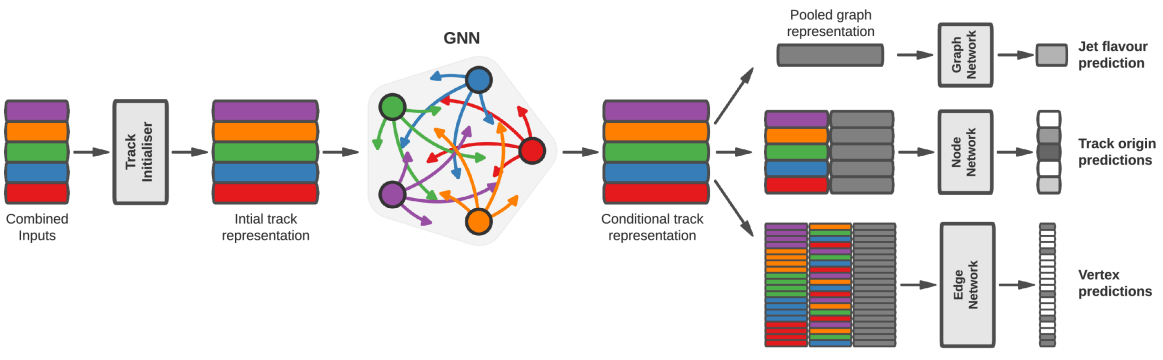
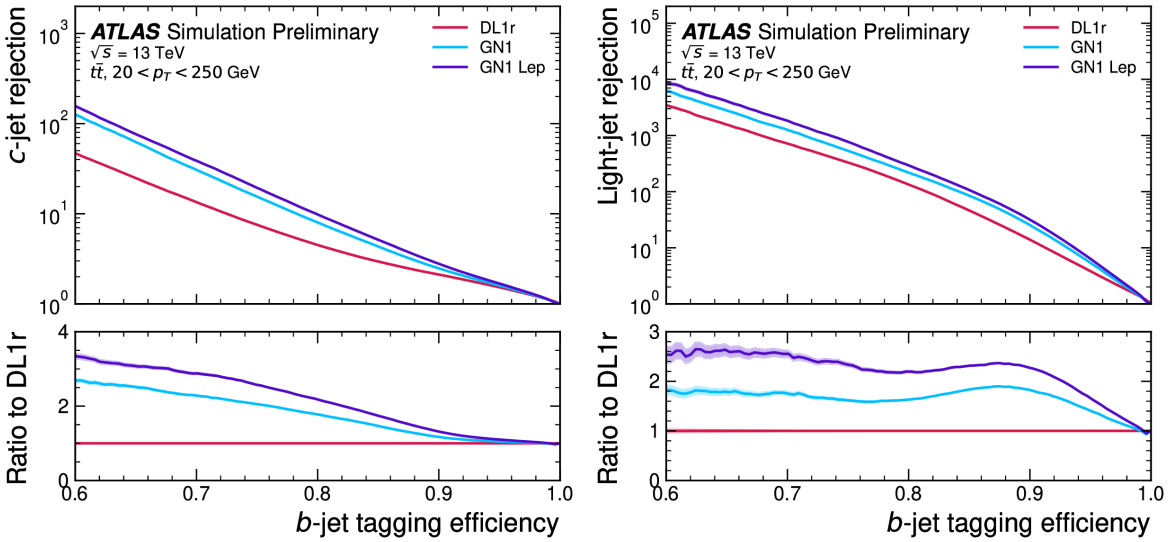
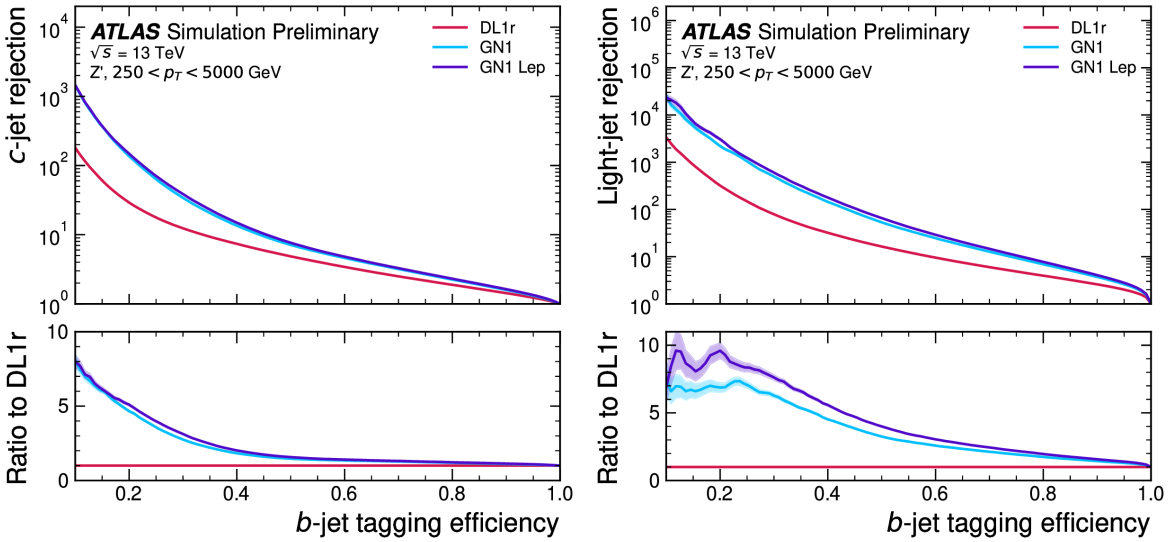


Figure 1.23: The architecture of the GN1 network, from [16]. The combined input is made of the set of tracks, each of which is given a copy of the two jet variables in addition to the track features as per Table 1.6. After a first embedding taking the input to an enriched latent representation, a fully connected graph is defined with the embedded tracks as nodes. The output of the graph is a conditional representation that is used in the three training objectives.



(a) GN1 and DL1r performance on the $t\bar{t}$ test sample, with $20 < p_T < 250$ GeV.



(b) GN1 and DL1r performance on the Z' test sample, with $250 < p_T < 5000$ GeV.

Figure 1.24: ROC curves tracing the b-tagging efficiency versus the c-jet (left) and light-jet (right) rejections for the $t\bar{t}$ (top) and Z' (bottom) test samples, from [16]. Models compared are DL1r in red, GN1 in blue, and GN1 Lep in purple. The bottom panels show the ratio with respect to DL1r. The flavour fraction is set at $f_c^b = 0.018$ for DL1r and 0.05 for GN1 and GN1 Lep. The binomial error bands are shown as shaded regions.

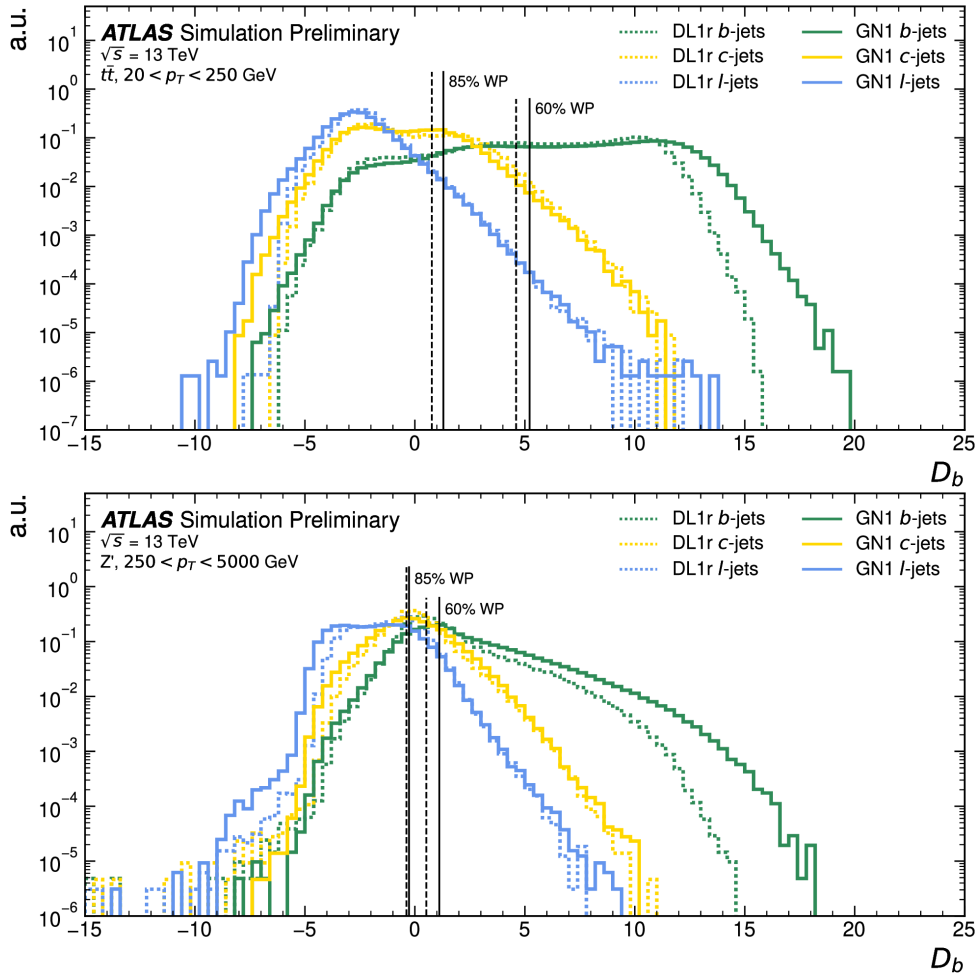
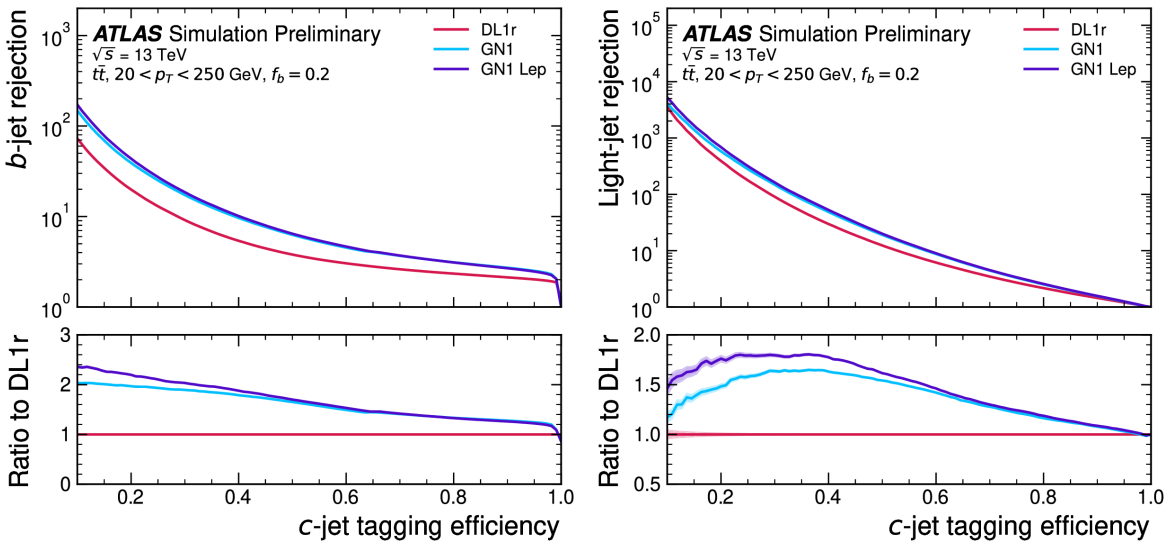
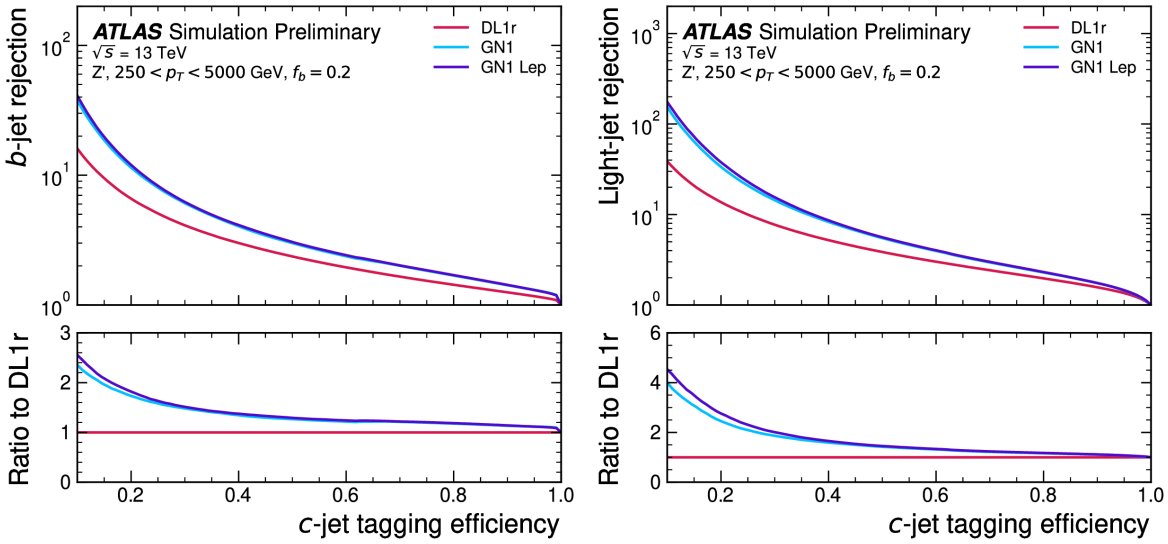


Figure 1.25: Comparing the GN1 and DL1r b-tagging discriminants D_b normalised distributions on the $t\bar{t}$ (top) and Z' (bottom) test samples, from [16]. Models compared are DL1r in dashed lines and GN1 in continuous line. Each flavour is indicated by a different colour: green for b-jets, yellow for c-jets, and blue for light-jets. The flavour fraction is set at $f_c^b = 0.018$ for DL1r and 0.05 for GN1



(a) GN1 performance on the $t\bar{t}$ test sample, with $20 < p_T < 250$ GeV.



(b) GN1 performance on the Z' test sample, with $250 < p_T < 5000$ GeV.

Figure 1.26: ROC curves tracing the c-tagging efficiency versus the b-jet (left) and light-jet (right) rejections for the $t\bar{t}$ (top) and Z' (bottom) test samples, from [16]. Models compared are DL1r in red, GN1 in blue, and GN1 Lep in purple. The bottom panels show the ratio with respect to DL1r. The flavour fraction is set at $f_b^c = 0.2$. The binomial error bands are shown as shaded regions.

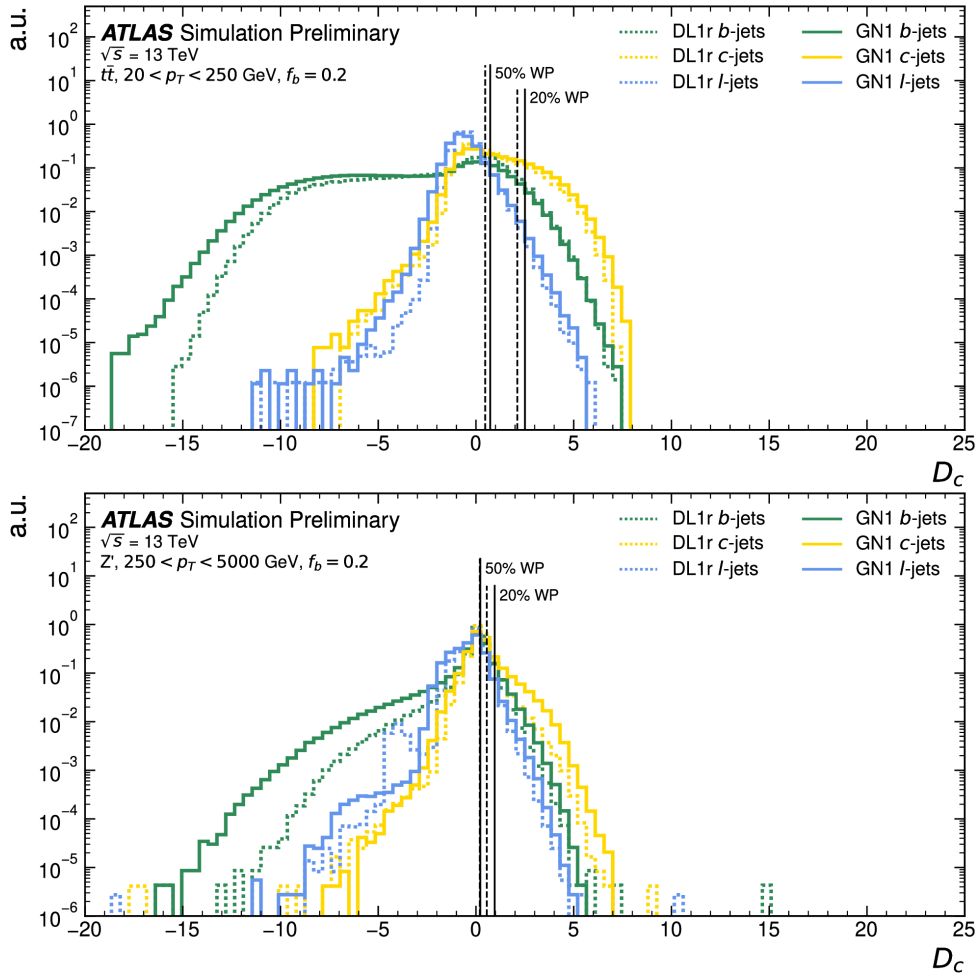


Figure 1.27: Comparing the GN1 and DL1r c-tagging discriminants D_c normalised distributions on the $t\bar{t}$ (top) and Z' (bottom) test samples, from [16]. Models compared are DL1r in dashed lines and GN1 in continuous line. Each flavour is indicated by a different colour: green for b-jets, yellow for c-jets, and blue for light-jets. The flavour fraction is set at $f_b^c = 0.2$.

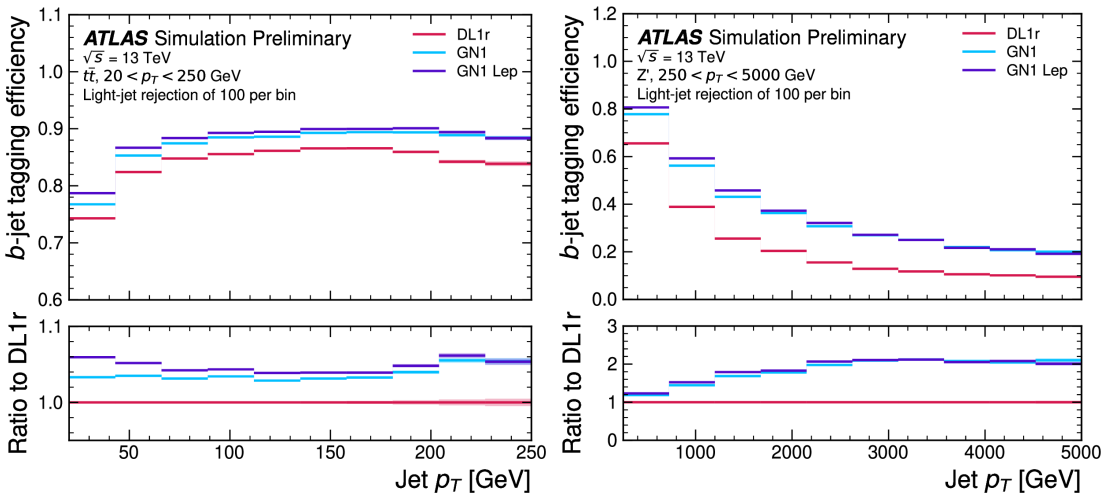


Figure 1.28: Comparing the GN1 and DL1r b-tagging efficiency as a function of jet p_T at a fixed 100 light-jet rejection in each bin on the $t\bar{t}$ (left) and Z' (right) test samples, from [16]. Models compared are DL1r in dashed lines and GN1 in continuous line. The flavour fraction is set at $f_c^b = 0.018$ for DL1r and 0.05 for GN1 and GN1 Lep.

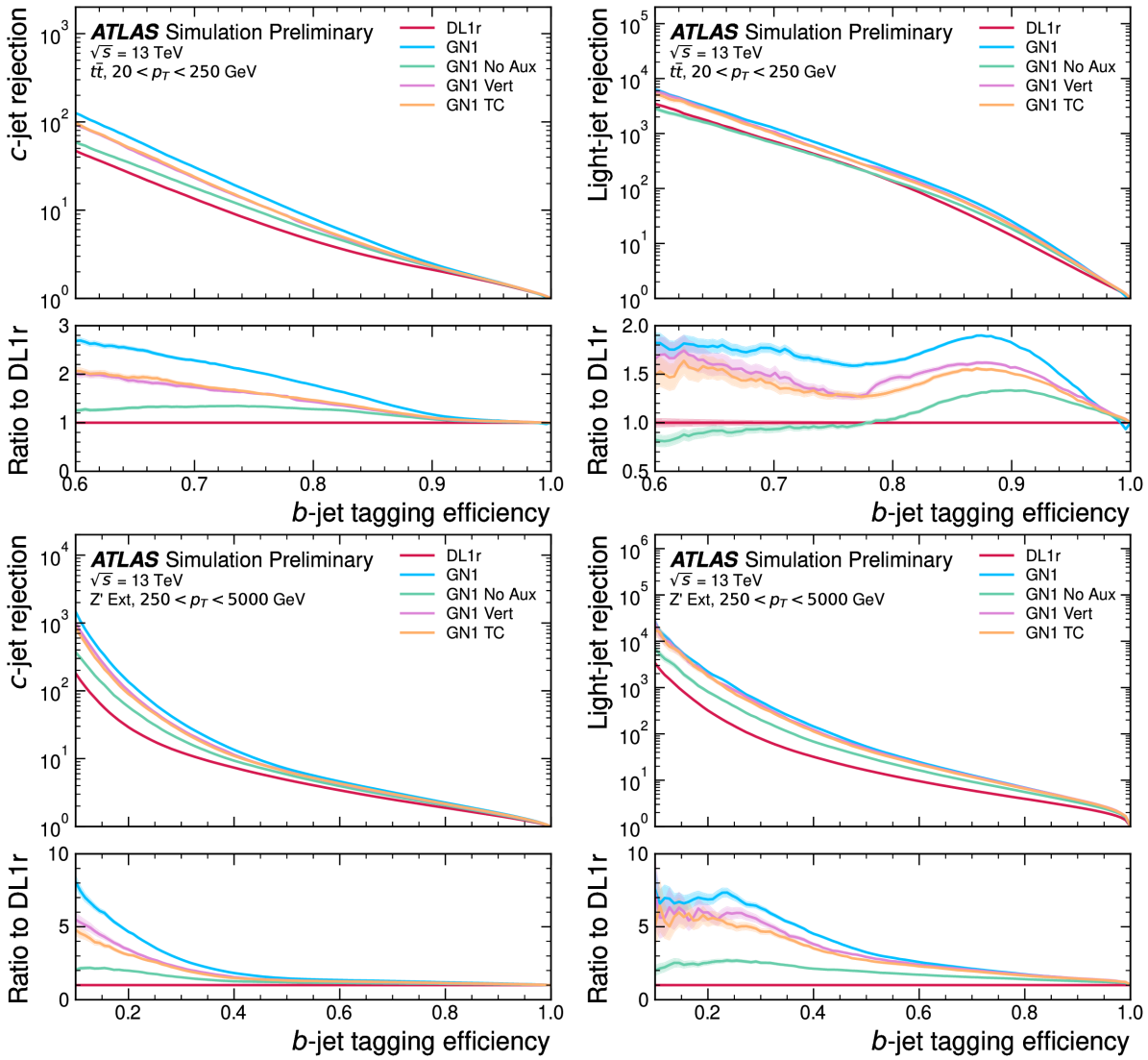


Figure 1.29: ROC curves tracing the b-tagging efficiency versus the c-rejection (left) and light-rejection (right) for the $t\bar{t}$ (top) and Z' (bottom) test samples, from [16]. Models compared are DL1r in red, GN1 in blue, and versions of GN1 with missing auxiliary tasks. GN1 No Aux in green has none of the auxiliary, GN1 Vert in purple only the vertexing task, and GN1 TC in orange only the track classification. The flavour fraction is set at $f_c^b = 0.018$ for DL1r and 0.05 for GN1. The binomial error bands are shown as shaded regions.

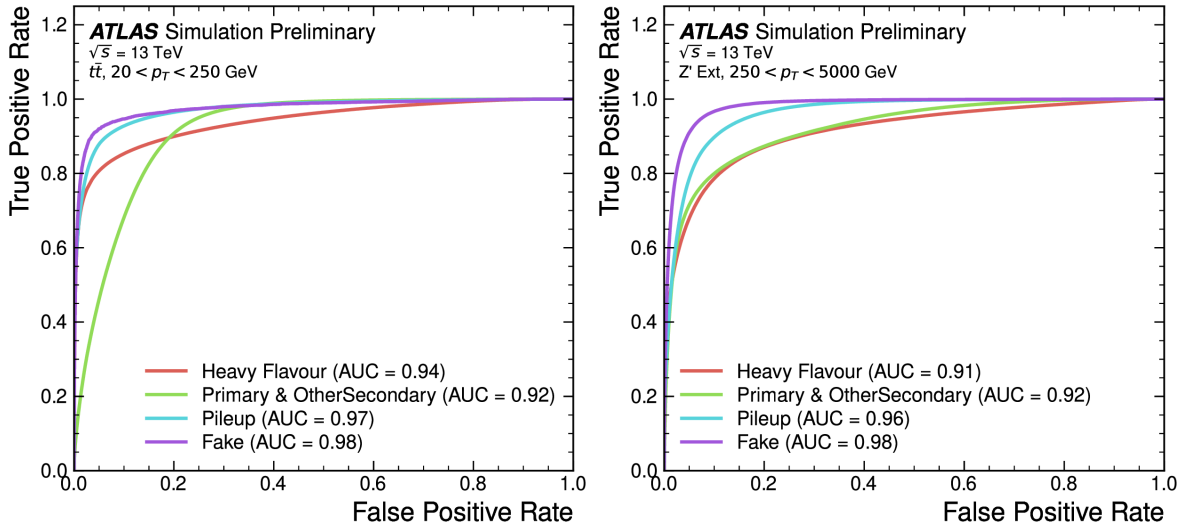


Figure 1.30: ROC curves tracing the false positive rate versus the true positive rate of the truth origin classification on the $t\bar{t}$ (left) and Z' (right) test samples, from [16]. Heavy Flavour is a weighted combination of the FromB, FromBC, and FromC by their relative abundance.

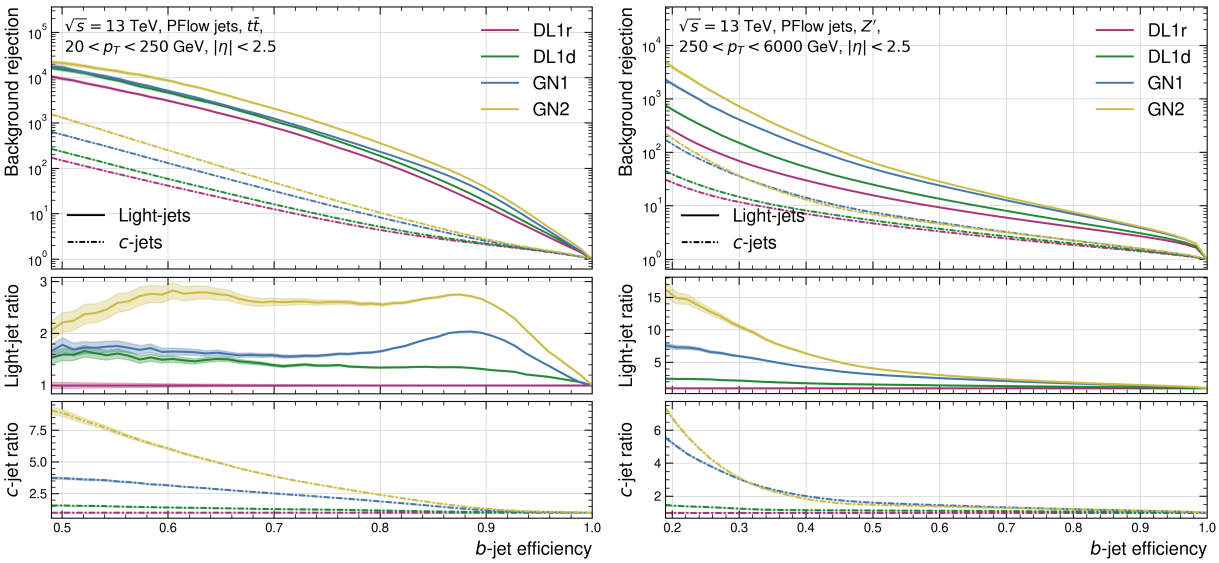


Figure 1.31: The c- and light-rejections as a function of the b-jet tagging efficiency in the $t\bar{t}$ with $20 < p_T < 250$ GeV (left) and Z' with $250 < p_T < 6000$ GeV (right) test samples. Models compared are DL1r in purple, DL1d in green, GN1 in blue, and GN2 in yellow. The bottom plots show the ratio with respect to the DL1d performance. Flavour fractions are set at $f_c^b = 0.018$ for DL1r and DL1d, 0.05 for GN1, and 0.1 for GN2. Shaded regions represent the binominal error band.

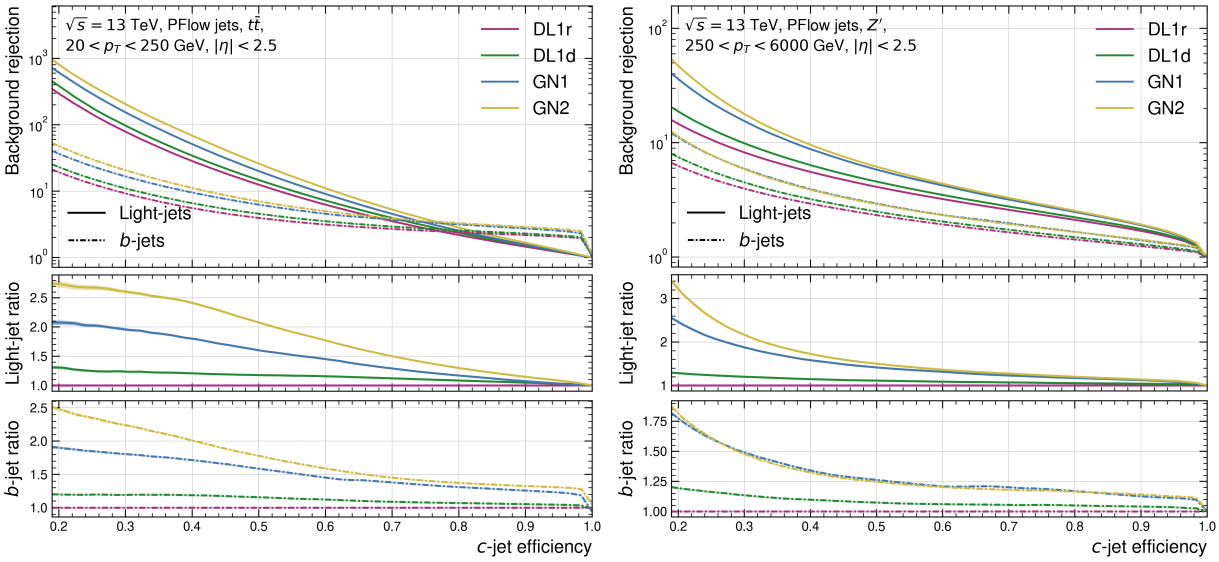


Figure 1.32: The b- and light-rejections as a function of the c-jet tagging efficiency in the $t\bar{t}$ with $20 < p_T < 250$ GeV (left) and Z' with $250 < p_T < 6000$ GeV (right) test samples. Models compared are DL1r in purple, DL1d in green, GN1 in blue, and GN2 in yellow. The bottom plots show the ratio with respect to the DL1d performance. Flavour fractions are set at $f_b^c = 0.2$ for all models. Shaded regions represent the binominal error band.

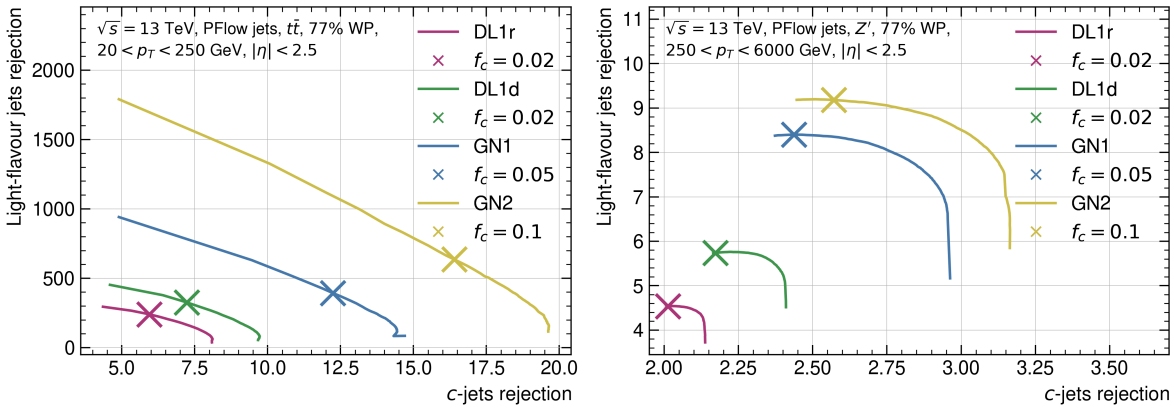


Figure 1.33: The flavour fraction f_c^b scans for b-tagging at a fixed working point of 77% of the different models considered evaluated on the $t\bar{t}$ (left) and Z' (right). The chosen values are marked on the curves, displaying on the x-axis the c-rejection vs the light-rejection on the y axis. Increasing f_c^b shifts the marker rightwards along the curves.

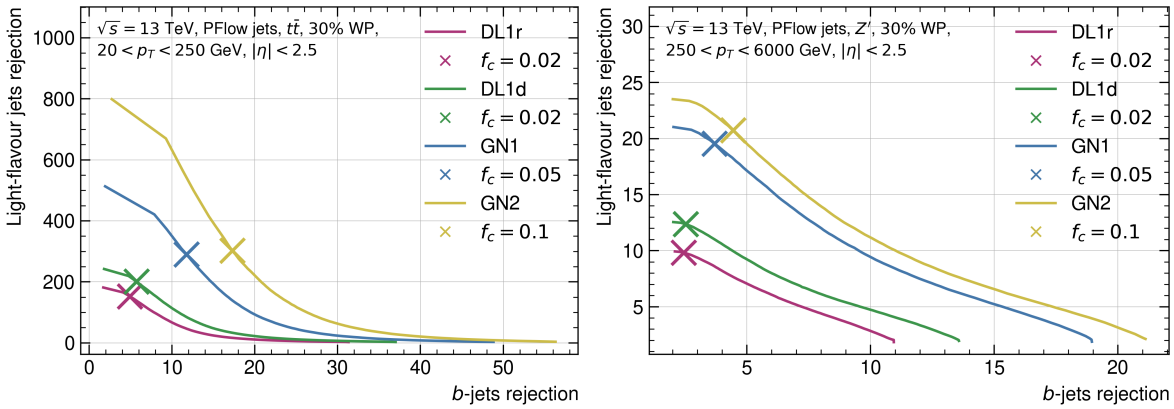


Figure 1.34: The flavour fraction f_b^c scans for c-tagging at a fixed working point of 30% of the different models considered evaluated on the $t\bar{t}$ (left) and Z' (right). The chosen values are marked on the curves, displaying on the x-axis the b-rejection vs the light-rejection on the y axis. Increasing f_b^c shifts the marker rightwards along the curves.

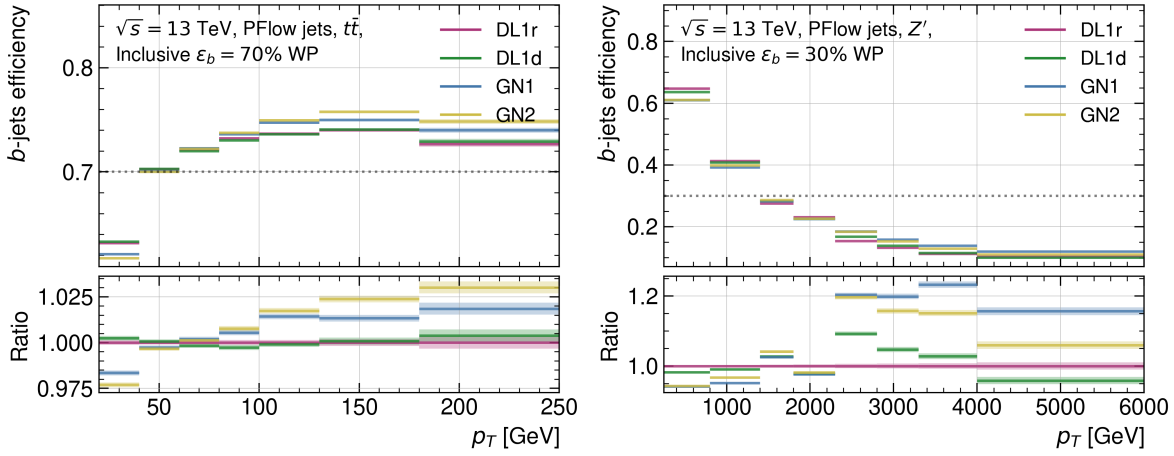


Figure 1.35: Comparing the different models b-tagging efficiency as a function of jet p_T for the inclusive b-tagging 70% working point on the $t\bar{t}$ (left) and 30% working point on Z' (right). The flavour fraction is set at $f_c^b = 0.018$ for DL1r and DL1d, 0.05 for GN1, and 0.1 for GN2.

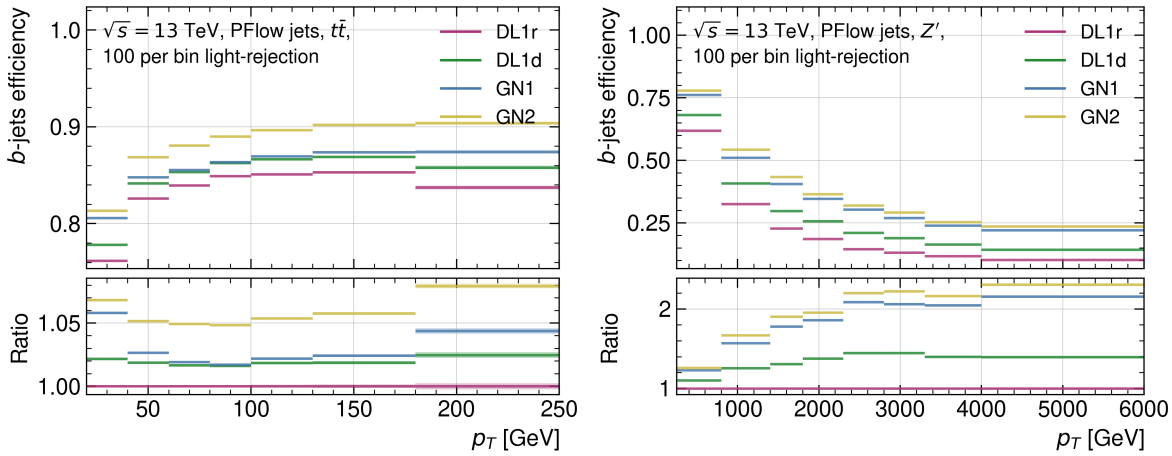


Figure 1.36: Comparing the different models b-tagging efficiency as a function of jet p_T at a fixed 100 light-jet rejection per bin on the $t\bar{t}$ (left) and Z' (right) test samples. The flavour fraction is set at $f_c^b = 0.018$ for DL1r and DL1d, 0.05 for GN1, and 0.1 for GN2.

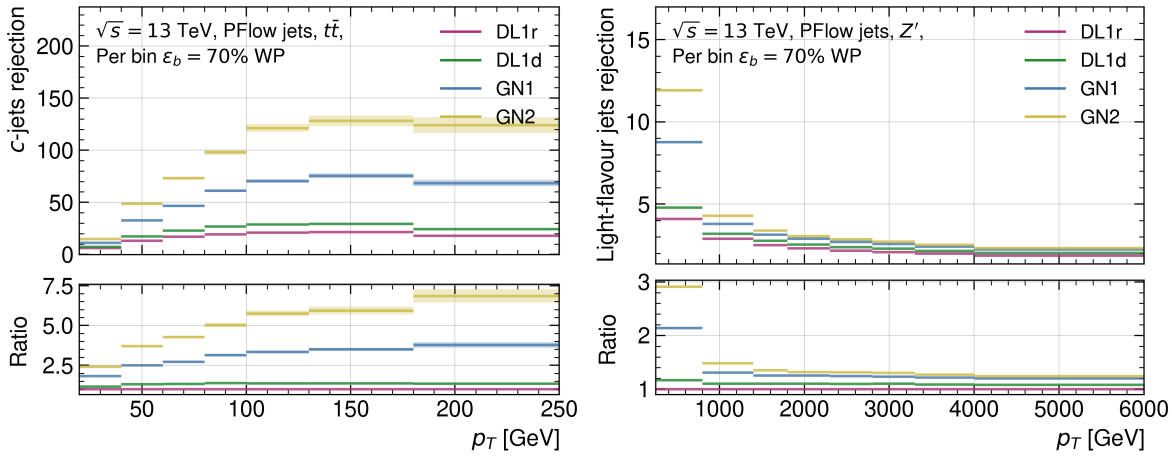


Figure 1.37: Comparing the different models c-rejection as a function of jet p_T for the b-tagging 70% working point per bin on the $t\bar{t}$ (left) and the 30% working point per bin on Z' (right). The flavour fraction is set at $f_c^b = 0.018$ for DL1r and DL1d, 0.05 for GN1, and 0.1 for GN2.

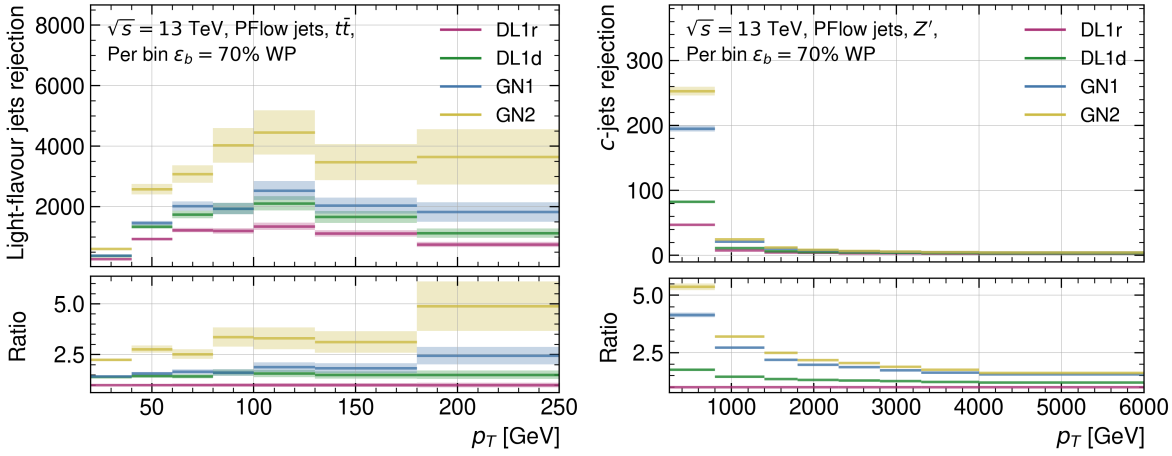


Figure 1.38: Comparing the different models light-rejection as a function of jet p_T for the b-tagging 70% working point per bin on the $t\bar{t}$ (left) and the 30% working point per bin on Z' (right). The flavour fraction is set at $f_c^b = 0.018$ for DL1r and DL1d, 0.05 for GN1, and 0.1 for GN2.

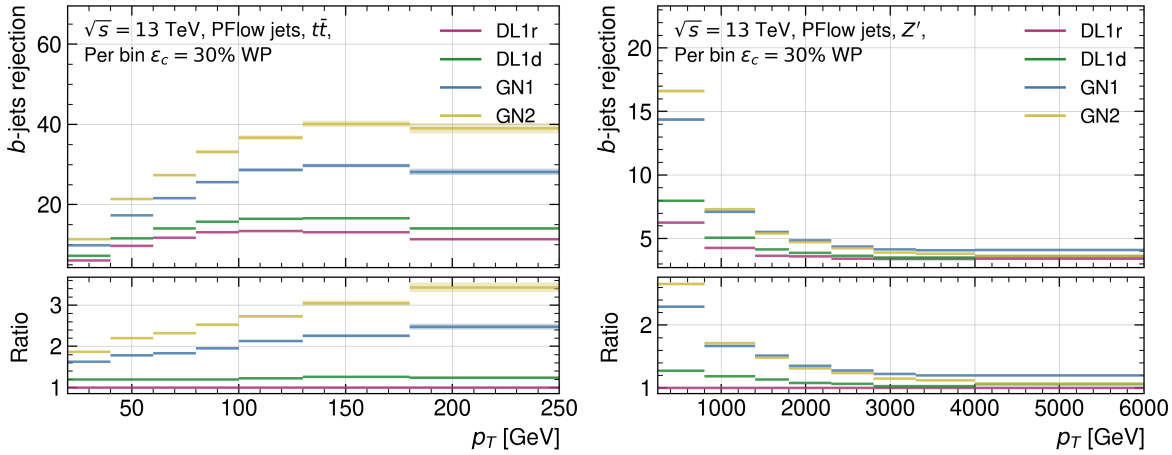


Figure 1.39: Comparing the different models b-rejection as a function of jet p_T for the c-tagging 30% working point per bin on the $t\bar{t}$ (left) and Z' (right). The flavour fraction is set at $f_b^c = 0.2$ for all taggers.

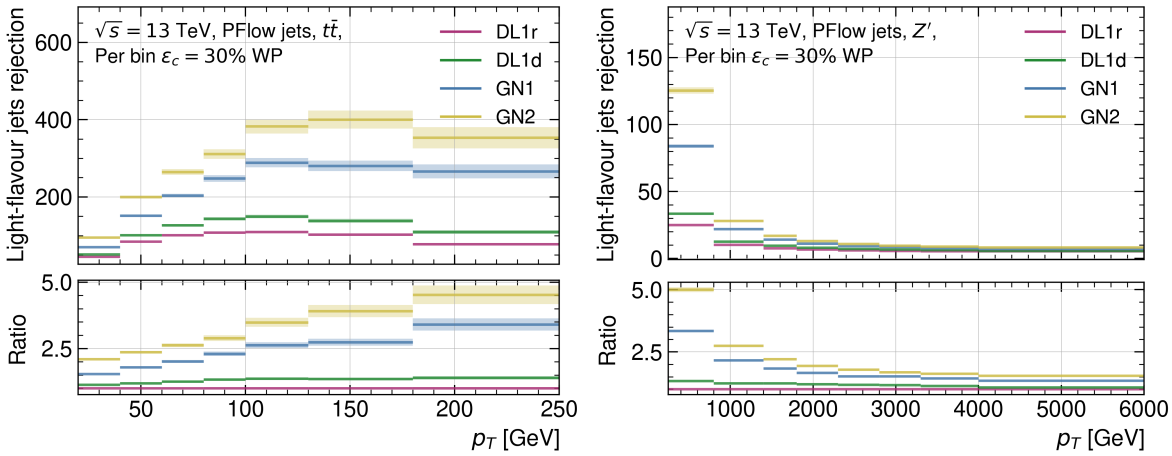


Figure 1.40: Comparing the different models light-rejection as a function of jet p_T for the c-tagging 30% working point per bin on the $t\bar{t}$ (left) and Z' (right). The flavour fraction is set at $f_b^c = 0.2$ for all taggers.

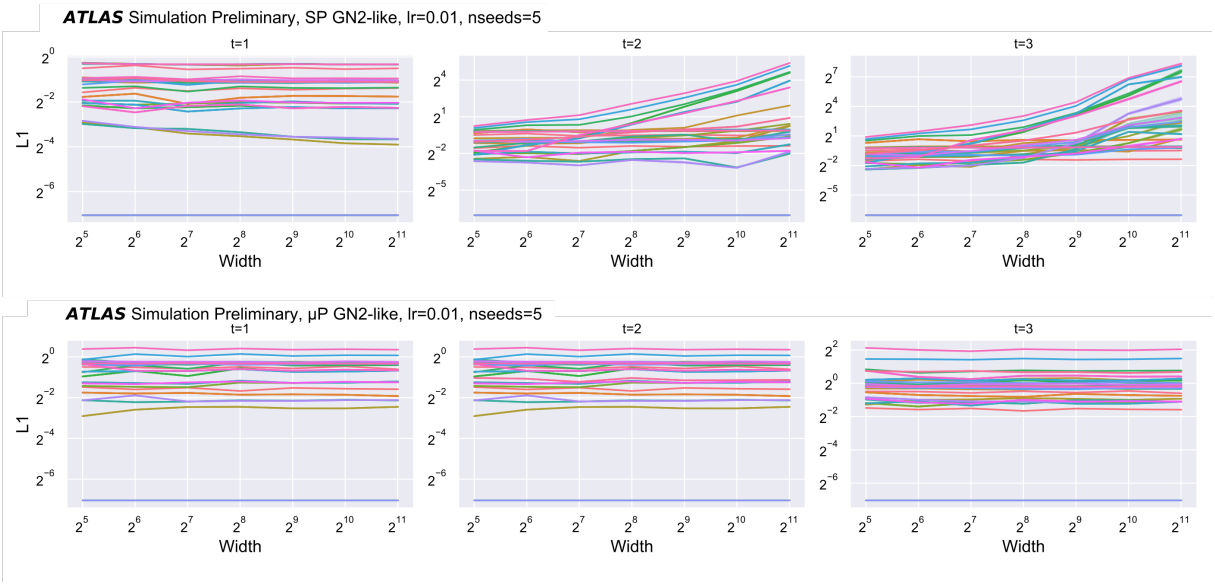


Figure 1.41: The sum of the absolute value of the pre-activation weights for the different layers in the initialiser and transformer parts of a GN2-like model in standard parametrisation (SP - top) and in μP parametrisation (bottom), at three timesteps: initialisation ($t = 1$ - left), after one training step with $lr = 10^{-2}$ ($t = 2$ - centre), and a second training step ($t = 3$). Taken from [65]. The models displayed are labelled GN2-like as they lack auxiliary tasks.

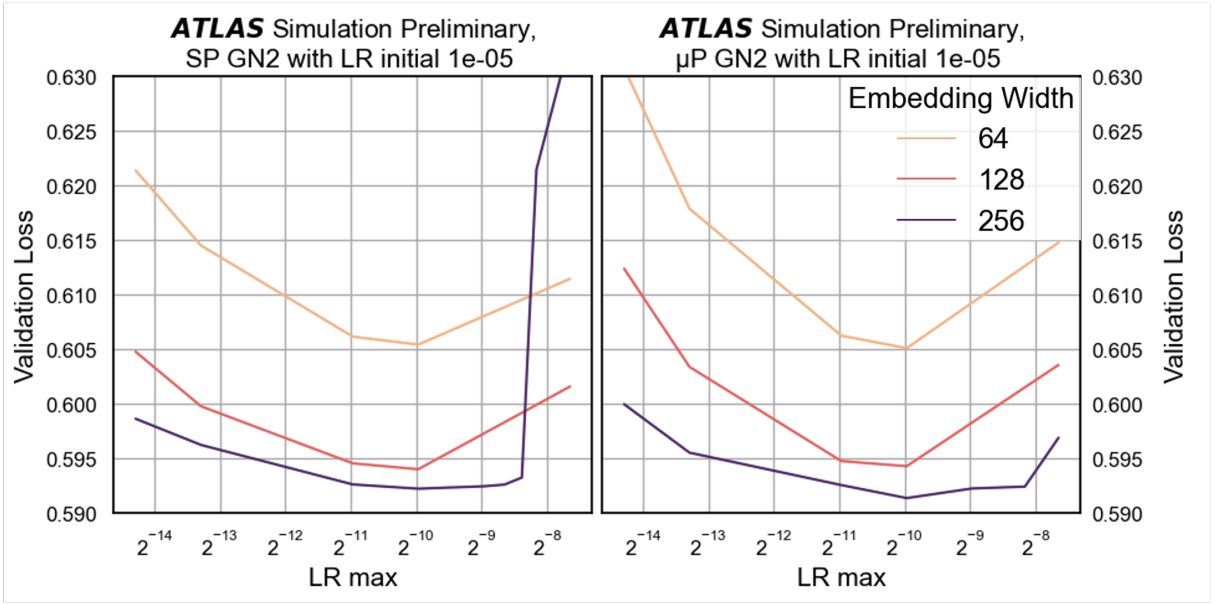


Figure 1.42: Comparison of a maximal learning rate value scan at an initial learning rate value of 10^{-5} for an SP (left) and a μP GN2 models (right) for three different embedding widths: 64 (yellow), 128 (red), and 256 (purple). The y-axis displays the validation loss attained. Taken from [65].

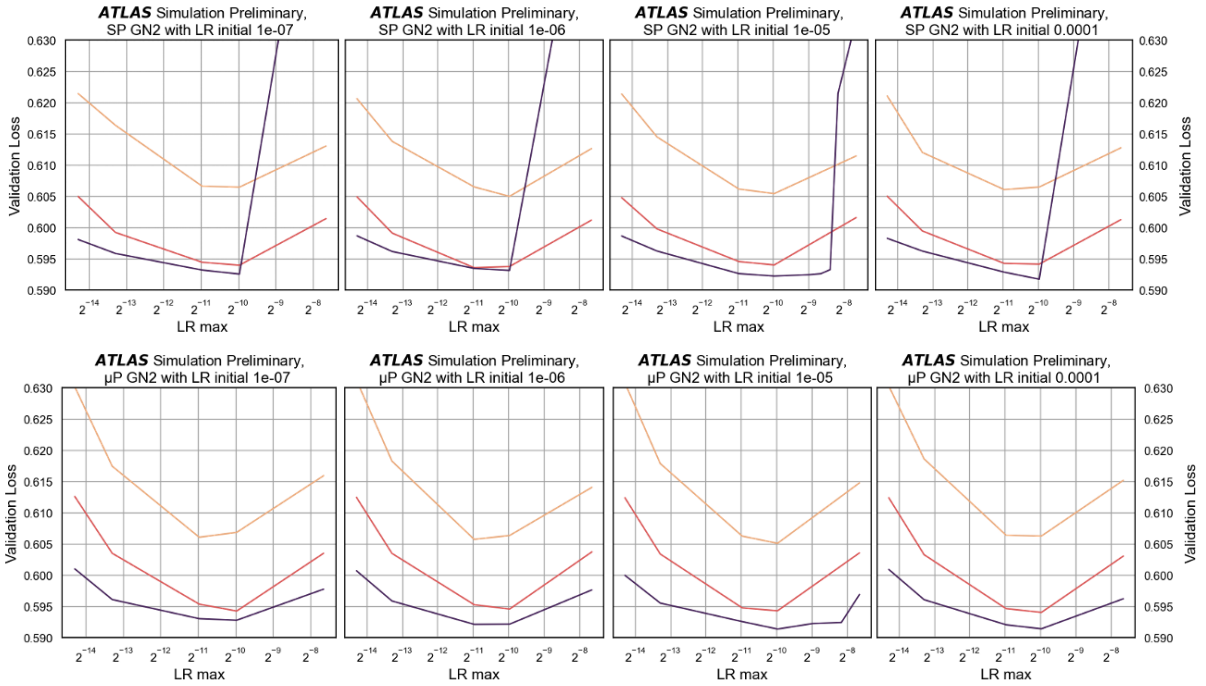


Figure 1.43: Scan of the maximal learning rate (x-axis) versus initial learning rate (individual column) as measured by the validation loss (y-axis) of SP models (top) and the μP model (bottom) with three different embedding widths: 64 (yellow), 128 (red), and 256 (purple). Taken from [65]. The scan at LR initial $= 10^{-5}$ benefitted from more test to capture the sudden rise in validation loss at larger LR max for SP.

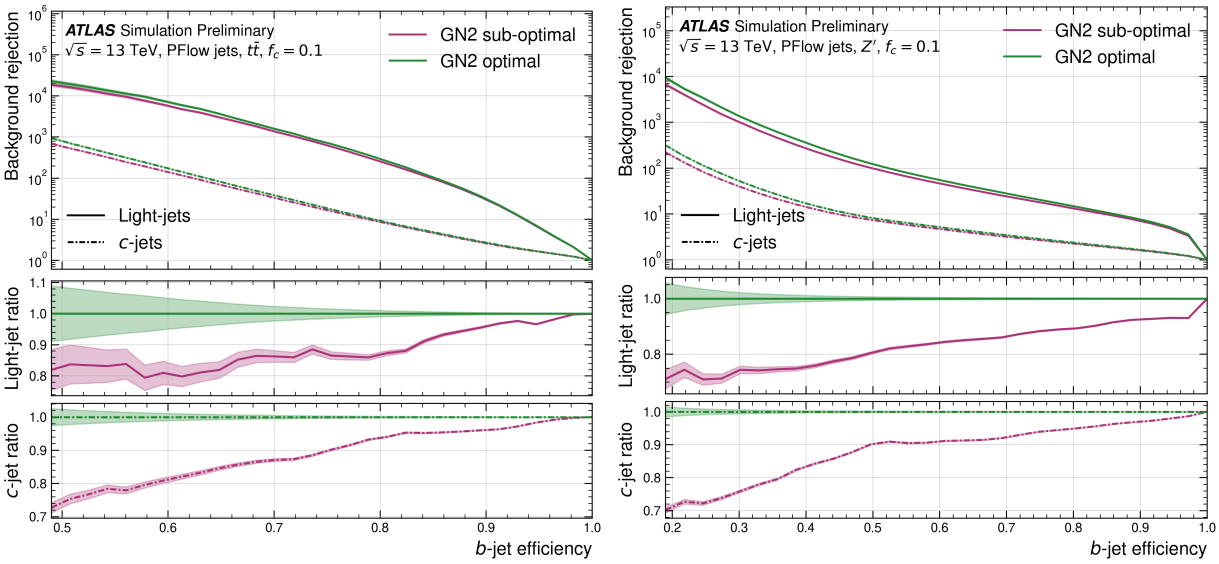


Figure 1.44: The c- and light-rejections as a function of the b-jet tagging efficiency in the $t\bar{t}$ (left) and Z' (right) test samples, from [65]. Models compared are the optimal μP GN2 (LR max $= 5 \times 10^{-5}$, LR init $= 10^{-5}$) and the suboptimal μP GN2 (LR max $= 5 \times 10^{-5}$, LR init $= 10^{-7}$), all with 256 embedding width. Shaded regions represent the binominal error band.

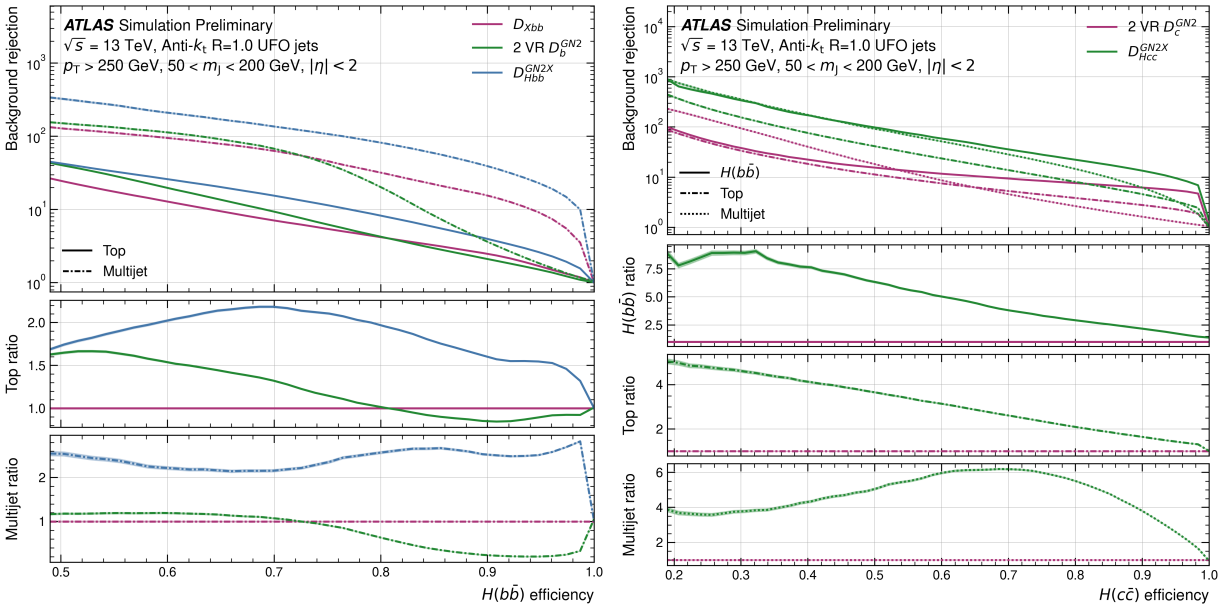


Figure 1.45: The ROC curves for $H(b\bar{b})$ (left) and $H(c\bar{c})$ tagging (right) on an SM simulated test samples, from [69]. The respective tagging efficiency is displayed versus the top and multijet rejections, for jets with a $p_T > 250$ GeV and a mass $50 < m_J < 200$ GeV. Models compared are the baseline X_{bb} tagger, using the variable-radius DL1r of at most 3 identified subjet in the large-R jet, the tag obtained by combining the tag on two variable-radius jets within the large-R jet with the single-jet GN2 tagger, and the GN2X model. The former is only available for $H(b\bar{b})$ tagging, and the $H(b\bar{b})$ rejection is displayed for $H(c\bar{c})$ tagging. The $H(c\bar{c})$ background is negligible for $H(b\bar{b})$ tagging. Shaded regions represent the binominal error band.

Bibliography

[1] B.R. Webber. "Fragmentation and Hadronization". In: *Int. J. Mod. Phys. AS* 15S1 (2000), pp. 577–606. DOI: 10.1142/S0217751X00005334. URL: %5Curl%7Bhttps://cds.cern.ch/record/419784%7D.

[2] ATLAS Collaboration. *Comparison of Monte Carlo generator predictions for bottom and charm hadrons in the decays of top quarks and the fragmentation of high pT jets*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2014-008. Geneva: CERN, 2014. URL: %5Curl%7Bhttps://cds.cern.ch/record/1709132%7D.

[3] Particle Data Group et al. "Review of Particle Physics". In: *Progress of Theoretical and Experimental Physics* 2020.8 (Aug. 2020), p. 083C01. ISSN: 2050-3911. DOI: 10.1093/ptep/ptaa104. eprint: https://academic.oup.com/ptep/article-pdf/2020/8/083C01/34673722/ptaa104.pdf. URL: https://doi.org/10.1093/ptep/ptaa104.

[4] Georges Aad et al. "ATLAS b-jet identification performance and efficiency measurement with $t\bar{t}$ events in pp collisions at $\sqrt{s} = 13$ TeV". In: *Eur. Phys. J. C* 79.11 (2019), p. 970. DOI: 10.1140/epjc/s10052-019-7450-8. arXiv: 1907.05120.

[5] Nazar Bartosik. *Diagram showing the common principle of identification of jets initiated by b-hadron decays*. https://en.m.wikipedia.org/wiki/File:B-tagging_diagram.png. Accessed: 2024-02-16. 2022.

[6] Samuel Van Stroud. "Graph Neural Network Flavour Tagging and Boosted Higgs Measurements at the LHC". Presented 2023. UCL, 2023. URL: %5Curl%7Bhttps://cds.cern.ch/record/2869719%7D.

[7] ATLAS Collaboration. "Search for the Decay of the Higgs Boson to Charm Quarks with the ATLAS Experiment". In: *Phys. Rev. Lett.* 120.21 (2018), p. 211802. DOI: 10.1103/PhysRevLett.120.211802. arXiv: 1802.04329.

[8] ATLAS Collaboration. *Direct constraint on the Higgs-charm coupling using Higgs boson decays to charm quarks with the ATLAS detector*. Tech. rep. Geneva: CERN, 2020. URL: %5Curl%7Bhttps://cds.cern.ch/record/2721696%7D.

[9] CMS Collaboration. *Search for Higgs boson decay to a charm quark-antiquark pair in proton-proton collisions at $\sqrt{s} = 13$ TeV*. Tech. rep. Geneva: CERN, 2022. arXiv: 2205.05550. URL: %5Curl%7Bhttps://cds.cern.ch/record/2809290%7D.

[10] ATLAS Collaboration. *Expected performance of the ATLAS b-tagging algorithms in Run-2*. Tech. rep. Geneva: CERN, 2015. URL: %5Curl%7Bhttps://cds.cern.ch/record/2037697%7D.

[11] ATLAS Collaboration. "Optimisation and performance studies of the ATLAS b-tagging algorithms for the 2017-18 LHC run". In: (July 2017).

[12] ATLAS Collaboration. *Identification of Jets Containing b-Hadrons with Recurrent Neural Networks at the ATLAS Experiment*. Tech. rep. ATL-PHYS-PUB-2017-003. CERN, 2017. URL: %5Curl%7Bhttps://cds.cern.ch/record/2255226%7D.

[13] ATLAS Collaboration. *Deep Sets based Neural Networks for Impact Parameter Flavour Tagging in ATLAS*. Tech. rep. ATL-PHYS-PUB-2020-014. CERN, 2020. URL: %5Curl%7Bhttps://cds.cern.ch/record/2718948%7D.

[14] ATLAS Collaboration. *The ATLAS Collaboration Software and Firmware*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-SOFT-PUB-2021-001. Geneva: CERN, 2021. URL: %5Curl%7Bhttps://cds.cern.ch/record/2767187%7D.

[15] ATLAS Collaboration. "ATLAS flavour-tagging algorithms for the LHC Run 2 pp collision dataset". In: *The European Physical Journal C* 83.7 (2023), p. 681. DOI: 10.1140/epjc/s10052-023-11699-1. URL: https://doi.org/10.1140/epjc/s10052-023-11699-1.

[16] *Graph Neural Network Jet Flavour Tagging with the ATLAS Detector*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2022-027. Geneva: CERN, 2022. URL: %5Curl%7Bhttps://cds.cern.ch/record/2811135%7D.

[17] ATLAS Collaboration. *Jet Flavour Tagging With GN1 and DL1d. Generator dependence, Run 2 and Run 3 data agreement studies*. Tech. rep. Geneva: CERN, 2023. URL: %5Curl%7Bhttps://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PLOTS/FTAG-2023-01/%7D.

[18] Arnaud Duperrin. *Flavour tagging with graph neural networks with the ATLAS detector*. 2023. arXiv: 2306.04415 [hep-ex].

[19] ATLAS Collaboration. *Neural Network Jet Flavour Tagging with the Upgraded ATLAS Inner Tracker Detector at the High-Luminosity LHC*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2022-047. Geneva: CERN, 2022. URL: %5Curl%7Bhttps://cds.cern.ch/record/2839913%7D.

[20] Paolo Nason. "A new method for combining NLO QCD with shower Monte Carlo algorithms". In: *Journal of High Energy Physics* 2004.11 (2004), p. 040. DOI: 10.1088/1126-6708/2004/11/040. URL: https://dx.doi.org/10.1088/1126-6708/2004/11/040.

[21] Stefano Frixione, Giovanni Ridolfi, and Paolo Nason. "A positive-weight next-to-leading-order Monte Carlo for heavy flavour hadroproduction". In: *Journal of High Energy Physics* 2007.09 (2007), p. 126. DOI: 10.1088/1126-6708/2007/09/126. URL: https://dx.doi.org/10.1088/1126-6708/2007/09/126.

[22] Stefano Frixione, Paolo Nason, and Carlo Oleari. "Matching NLO QCD computations with parton shower simulations: the POWHEG method". In: *Journal of High Energy Physics* 2007.11 (2007), p. 070. DOI: 10.1088/1126-6708/2007/11/070. URL: https://dx.doi.org/10.1088/1126-6708/2007/11/070.

[23] Simone Alioli et al. "A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX". In: *Journal of High Energy Physics* 2010.6 (2010), p. 43. DOI: 10.1007/JHEP06(2010)043. URL: https://doi.org/10.1007/JHEP06(2010)043.

[24] Richard D. Ball et al. "Parton distributions for the LHC run II". In: *Journal of High Energy Physics* 2015.4 (2015), p. 40. DOI: 10.1007/JHEP04(2015)040. URL: https://doi.org/10.1007/JHEP04(2015)040.

[25] Torbjörn Sjöstrand et al. "An introduction to PYTHIA 8.2". In: *Computer Physics Communications* 191 (2015), pp. 159–177. ISSN: 0010-4655. DOI: https://doi.org/10.1016/j.cpc.2015.01.024. URL: https://www.sciencedirect.com/science/article/pii/S0010465515000442.

[26] ATLAS Collaboration. *ATLAS Pythia 8 tunes to 7 TeV data*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2014-021. Geneva: CERN, 2014. URL: %5Curl%7Bhttps://cds.cern.ch/record/1966419%7D.

[27] Richard D. Ball et al. "Parton distributions with LHC data". In: *Nuclear Physics B* 867.2 (2013), pp. 244–289. ISSN: 0550-3213. DOI: https://doi.org/10.1016/j.nuclphysb.2012.10.003. URL: https://www.sciencedirect.com/science/article/pii/S0550321312005500.

[28] ATLAS Collaboration. *Studies on top-quark Monte Carlo modelling for Top2016*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2016-020. Geneva: CERN, 2016. URL: %5Curl%7Bhttps://cds.cern.ch/record/2216168%7D.

[29] ATLAS Collaboration. *Study of top-quark pair modelling and uncertainties using ATLAS measurements at $\sqrt{s}$=13 TeV*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2020-023. Geneva: CERN, 2020. URL: %5Curl%7Bhttps://cds.cern.ch/record/2730443%7D.

[30] David J. Lange. "The EvtGen particle decay simulation package". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 462.1 (2001). BEAUTY2000, Proceedings of the 7th Int. Conf. on B-Physics at Hadron Machines, pp. 152–155. ISSN: 0168-9002. DOI: https://doi.org/10.1016/S0168-9002(01)00089-4. URL: https://www.sciencedirect.com/science/article/pii/S0168900201000894.

[31] ATLAS Collaboration. "The ATLAS Simulation Infrastructure". In: *The European Physical Journal C* 70.3 (2010), pp. 823–874. DOI: 10.1140/epjc/s10052-010-1429-9. URL: https://doi.org/10.1140/epjc/s10052-010-1429-9.

[32] GEANT4 Collaboration. "GEANT4, A Simulation toolkit". In: *Nucl. Instrum. Methods Phys. Res., A* 506.CERN-IT-2002-003 (2002), 250–303. 54 p. URL: %5Curl%7Bhttps://cds.cern.ch/record/602040%7D.

[33] ATLAS Collaboration. *Tagging and suppression of pileup jets with the ATLAS detector*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/CONFNOTES/ATLAS-CONF-2014-018. Geneva: CERN, 2014. URL: %5Curl%7Bhttps://cds.cern.ch/record/1700870%7D.

[34] ATLAS Collaboration. *Machine Learning Algorithms for b-Jet Tagging at the ATLAS Experiment*. Tech. rep. ATL-PHYS-PROC-2017-211. CERN, 2017. DOI: 10.1088/1742-6596/1085/4/042031.

[35] V Kostyukhin. *VKalVrt - package for vertex reconstruction in ATLAS*. Tech. rep. ATL-PHYS-2003-031. CERN, 2003. URL: %5Curl%7Bhttps://cds.cern.ch/record/685551%7D.

[36] *Topological b-hadron decay reconstruction and identification of b-jets with the JetFitter package in the ATLAS experiment at the LHC*. Tech. rep. Geneva: CERN, 2018. URL: %5Curl%7Bhttps://cds.cern.ch/record/2645405%7D.

[37] Manzil Zaheer et al. "Deep Sets". In: *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/f22e4747da1aa27e363d86d40ff442fe-Paper.pdf.

[38] ATLAS Collaboration. "Performance of the ATLAS track reconstruction algorithms in dense environments in LHC Run 2". In: *The European Physical Journal C* 77.10 (2017), p. 673. DOI: 10.1140/epjc/s10052-017-5225-7. URL: https://doi.org/10.1140/epjc/s10052-017-5225-7.

[39] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps". In: *CoRR* abs/1312.6034 (2013). URL: https://api.semanticscholar.org/CorpusID:1450294.

[40] ATLAS Collaboration. "Performance of jet substructure techniques for large-R jets in proton-proton collisions at $\sqrt{s} = 7$ TeV using the ATLAS detector". In: *Journal of High Energy Physics* 2013.9 (2013), p. 76. DOI: 10.1007/JHEP09(2013)076. URL: https://doi.org/10.1007/JHEP09(2013)076.

[41] "Boosted Object Tagging with Variable-R Jets in the ATLAS Detector". In: (July 2016). URL: %5Curl%7Bhttps://cds.cern.ch/record/2199360%7D.

[42] David Krohn, Jesse Thaler, and Lian-Tao Wang. "Jets with variable R". In: *Journal of High Energy Physics* 2009.06 (2009), p. 059. DOI: 10.1088/1126-6708/2009/06/059. URL: https://dx.doi.org/10.1088/1126-6708/2009/06/059.

[43] *Umami Framework*. https://gitlab.cern.ch/atlas-flavor-tagging-tools/algorithms/umami. Accessed: 2023-04-21.

[44] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: %5Curl%7Bhttps://www.tensorflow.org/%7D.

[45] Benedek Rozemberczki et al. "The Shapley Value in Machine Learning". In: *ArXiv* abs/2202.05594 (2022). URL: https://api.semanticscholar.org/CorpusID:246822765.

[46] Scott M Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 4765–4774. URL: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

[47] *Salt Framework*. https://gitlab.cern.ch/atlas-flavor-tagging-tools/algorithms/salt. Accessed: 2024-02-20.

[48] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035. URL: %5Curl%7Bhttp://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf%7D.

[49] Petar Veličković et al. "Graph Attention Networks". In: *International Conference on Learning Representations* (2018). URL: https://openreview.net/forum?id=rJXMpikCZ.

[50] Shaked Brody, Uri Alon, and Eran Yahav. "How Attentive are Graph Attention Networks?" In: *International Conference on Learning Representations*. 2022. URL: https://openreview.net/forum?id=F72ximsx7C1.

[51] Ashish Vaswani et al. "Attention is All you Need". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[52] Dasol Hwang et al. *Self-supervised Auxiliary Learning with Meta-paths for Heterogeneous Graphs*. 2021. arXiv: 2007.08294 [cs.LG].

[53] Hadar Serviansky et al. *Set2Graph: Learning Graphs From Sets*. 2020. arXiv: 2002.08772 [cs.LG].

[54] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. 2015. URL: http://arxiv.org/abs/1412.6980.

[55] Rachel E. C. Smith et al. *Differentiable Vertex Fitting for Jet Flavour Tagging*. 2023. arXiv: 2310.12804 [hep-ex].

[56] Sam Shleifer, Jason Weston, and Myle Ott. *NormFormer: Improved Transformer Pretraining with Extra Normalization*. 2021. arXiv: 2110.09456 [cs.CL].

[57] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. *Layer Normalization*. cite arxiv:1607.06450. 2016. URL: http://arxiv.org/abs/1607.06450.

[58] Leslie N. Smith. *A disciplined approach to neural network hyperparameters: Part 1 – learning rate, batch size, momentum, and weight decay*. 2018. arXiv: 1803.09820 [cs.LG].

[59] Leslie N. Smith and Nicholay Topin. *Super-Convergence: Very Fast Training of Neural Networks Using Large Learning Rates*. 2018. arXiv: 1708.07120 [cs.LG].

[60] *ml.cern.ch*. https://ml.docs.cern.ch. Accessed: 2024-02-16.

[61] Johnu George et al. *A Scalable and Cloud-Native Hyperparameter Tuning System*. 2020. arXiv: 2006.02085 [cs.DC].

[62] Greg Yang et al. "Tuning Large Neural Networks via Zero-Shot Hyperparameter Transfer". In: *Advances in Neural Information Processing Systems*. Ed. by A. Beygelzimer et al. 2021. URL: https://openreview.net/forum?id=Bx6qKuBM2AD.

[63] Greg Yang and Edward J. Hu. "Tensor Programs IV: Feature Learning in Infinite-Width Neural Networks". In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 2021, pp. 11727–11737. URL: https://proceedings.mlr.press/v139/yang21c.html.

[64] Yann A. LeCun et al. "Efficient BackProp". In: *Neural Networks: Tricks of the Trade: Second Edition*. Ed. by Grégoire Montavon, Geneviève B. Orr, and Klaus-Robert Müller. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 9–48. ISBN: 978-3-642-35289-8. DOI: 10.1007/978-3-642-35289-8_3. URL: https://doi.org/10.1007/978-3-642-35289-8_3.

[65] Maxence Draguet. "Training and optimisation of large transformer models at CERN: an ATLAS case study on Kubeflow". In: *Sixth Inter-Experiment Machine Learning Workshop*. Geneva, 2024. URL: %7Bhttps://indico.cern.ch/event/1297159/contributions/5729198/%7D.

[66] ATLAS Collaboration. "Direct constraint on the Higgs-charm coupling from a search for Higgs boson decays into charm quarks with the ATLAS detector". In: *Eur. Phys. J. C* 82 (2022), p. 717. DOI: 10.1140/epjc/s10052-022-10588-3. arXiv: 2201.11428 [hep-ex].

[67] ATLAS Collaboration. "Constraints on Higgs boson production with large transverse momentum using $H \to b\bar{b}$ decays in the ATLAS detector". In: *Phys. Rev. D* 105 (9 2022), p. 092003. DOI: 10.1103/PhysRevD.105.092003. URL: https://link.aps.org/doi/10.1103/PhysRevD.105.092003.

[68] ATLAS Collaboration. "Anomaly detection search for new resonances decaying into a Higgs boson and a generic new particle X in hadronic final states using $\sqrt{s} = 13$ TeV pp collisions with the ATLAS detector". In: *Phys. Rev. D* 108 (2023), p. 052009. DOI: 10.1103/PhysRevD.108.052009. arXiv: 2306.03637 [hep-ex].

[69] ATLAS Collaboration. *Transformer Neural Networks for Identifying Boosted Higgs Bosons decaying into $b\bar{b}$ and $c\bar{c}$ in ATLAS*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2023-021. Geneva: CERN, 2023. URL: %5Curl%7Bhttps://cds.cern.ch/record/2866601%7D.

[70] ATLAS Collaboration. "Optimisation of large-radius jet reconstruction for the ATLAS detector in 13 TeV proton–proton collisions". In: *The European Physical Journal C* 81.4 (2021), p. 334. DOI: 10.1140/epjc/s10052-021-09054-3. URL: https://doi.org/10.1140/epjc/s10052-021-09054-3.

[71] ATLAS Collaboration. "Jet reconstruction and performance using particle flow with the ATLAS Detector". In: *The European Physical Journal C* 77.7 (2017), p. 466. DOI: 10.1140/epjc/s10052-017-5031-2. URL: https://doi.org/10.1140/epjc/s10052-017-5031-2.

[72] ATLAS Collaboration. *Improving jet substructure performance in ATLAS using TrackCaloClusters*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2017-015. Geneva: CERN, 2017. URL: %5Curl%7Bhttps://cds.cern.ch/record/2275636%7D.

[73] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. "The anti-k_t jet clustering algorithm". In: *JHEP* 04 (2008), p. 063. DOI: 10.1088/1126-6708/2008/04/063. arXiv: 0802.1189 [hep-ph].

[74] ATLAS Collaboration. *Identification of Boosted Higgs Bosons Decaying Into $b\bar{b}$ With Neural Networks and Variable Radius Subjets in ATLAS*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2020-019. Geneva: CERN, 2020. URL: %5Curl%7Bhttps://cds.cern.ch/record/2724739%7D.

[75] ATLAS Collaboration. *Efficiency corrections for a tagger for boosted $H \to b\bar{b}$ decays in pp collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2021-035. Geneva: CERN, 2021. URL: %5Curl%7Bhttps://cds.cern.ch/record/2777811%7D.

[76] ATLAS Collaboration. *Calibration of the ATLAS b-tagging algorithm in $t\bar{t}$ semi-leptonic events*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/CONFNOTES/ATLAS-CONF-2018-045. Geneva: CERN, 2018. URL: %5Curl%7Bhttps://cds.cern.ch/record/2638455%7D.

[77] ATLAS Collaboration. *Calibration of light-flavour b-jet mistagging rates using ATLAS proton-proton collision data at $\sqrt{s} = 13$ TeV*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/CONFNOTES/ATLAS-CONF-2018-006. Geneva: CERN, 2018. URL: %5Curl%7Bhttps://cds.cern.ch/record/2314418%7D.

[78] ATLAS Collaboration. "Measurement of the c-jet mistagging efficiency in $t\bar{t}$ events using pp collision data at $\sqrt{s} = 13$ TeV collected with the ATLAS detector". In: *The European Physical Journal C* 82.1 (2022), p. 95. DOI: 10.1140/epjc/s10052-021-09843-w. URL: https://doi.org/10.1140/epjc/s10052-021-09843-w.

[79] ATLAS Collaboration. "Calibration of the light-flavour jet mistagging efficiency of the b-tagging algorithms with Z+jets events using 139 fb^{-1} of ATLAS proton-proton collision data at $\sqrt{s} = 13$ TeV". In: *Eur. Phys. J. C* 83.8 (2023), p. 728. DOI: 10.1140/epjc/s10052-023-11736-z. arXiv: 2301.06319 [hep-ex].

[80] ATLAS Collaboration. *Monte Carlo to Monte Carlo scale factors for flavour tagging efficiency calibration*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2020-009. Geneva: CERN, 2020. URL: %5Curl%7Bhttps://cds.cern.ch/record/2718610%7D.

[81] Johannes Bellm et al. *Herwig 7.1 Release Note*. 2017. arXiv: 1705.06919 [hep-ph].

[82] Enrico Bothmann et al. "Event generation with Sherpa 2.2". In: *SciPost Phys.* 7 (2019), p. 034. DOI: 10.21468/SciPostPhys.7.3.034. URL: https://scipost.org/10.21468/SciPostPhys.7.3.034.

[83] J. Alwall et al. “The automated computation of tree-level and next-to-leading order differential cross-sections, and their matching to parton shower simulations”. In: *Journal of High Energy Physics* 2014.7 (2014), p. 79. DOI: 10.1007/JHEP07(2014)079. URL: https://doi.org/10.1007/JHEP07(2014)079.

Appendices

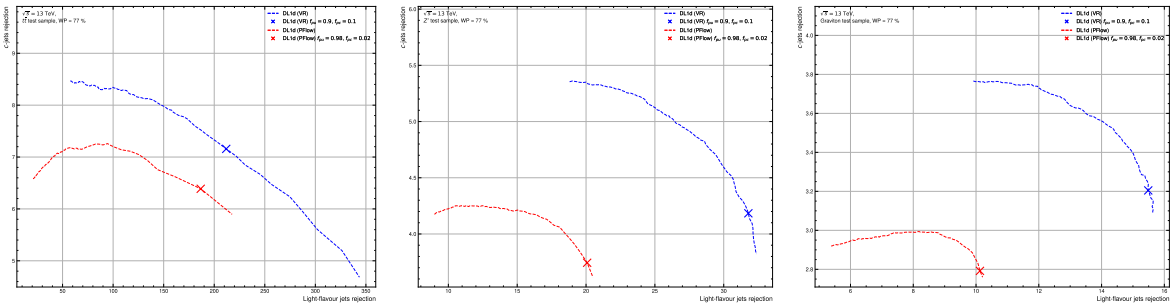
APPENDIX A

Flavour Tagging

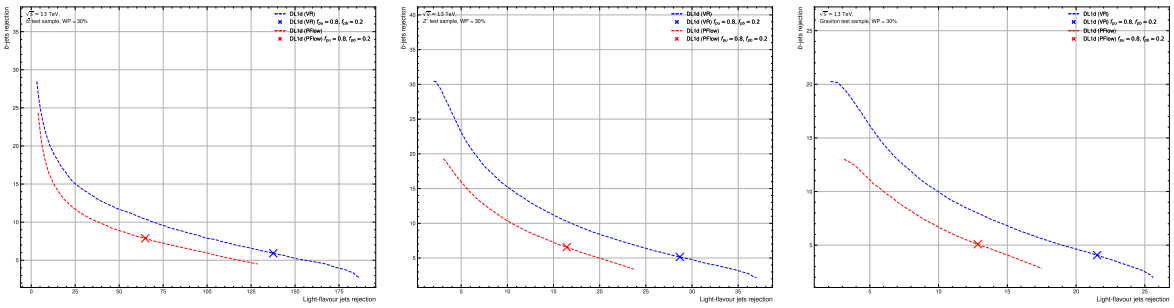
This Appendix lists some additional results in support of Chapter 1.

A.1 DL1d with Variable Radius Jets

This section presents some plots on the VR-training of DL1d. Figure A.1 displays some flavour fractions scans for the b-tagging and c-tagging.



(a) Flavour fraction f_c^b for b-tagging scans.



(b) Flavour fraction f_b^c for c-tagging scans.

Figure A.1: The flavour fractions scans of the VR- and PFlow-trained DL1d model in blue and red respectively: left is $t\bar{t}$, centre Z', and right the graviton test samples. The chosen values are marked on the curves, displaying on the y-axis the c-rejection (b-rejection) for b-tagging (c-tagging) vs the light-rejection on the x axis at a fixed working point of 77% (33%). Increasing f_c or f_b shifts the marker upwards along the curves.

A.2 GN2 public plots

A comparison of the global performance of this GN2 model to the DL1d and GN1 models is displayed in the b- and c-tagging ROC curves of Figures 1.31 and 1.32. These results are taken from Ref [17], for which the DL1d model was retrained on the same dataset as GN2, and the DL1r and GN1 models are taken from Chapter 1.3.1. GN2 delivers yet another significant boost to performance, drasticaly surpassing the GN1 rejections at all efficiencies considered. The largest improvement is again obtained at lower b-jet efficiencies. Compared to GN1, GN2 delivers $\times 1.5$ ($\times 1.7$) the c-rejection (light-rejection) on $t\bar{t}$ at the 70% b-tagging WP and $\times 1.75$ ($\times 1.2$) on Z' at 30% WP. With respect to DL1d, the gains in c-rejection (light-rejection) are respectively close to $\times 3$ ($\times 2$) for $t\bar{t}$ and $\times 3.4$ ($\times 4$) on Z'.

Turning to c-tagging, a similar large performance gained is obtained by the new GNN family over DL1d, although the change on the $t\bar{t}$ is more impressive for the b-jet ratio than for light-jet. This indicates a non-optimal choice for the flavour fraction f_b^c, which was set at 0.2 for all models.

A.3 GN2 supporting plots

This section presents more plots in support of Chapter 1.3.2. Figure A.4 presents the c-tagging efficiency per bin for an overall c-tagging working point of 30% per region displayed.

Figure A.5 presents the c-tagging efficiency per bin for a per bin light-rejection of 50 for $t\bar{t}$ and 10 for Z'. The GN2 performance dominates across the board, except for the highest energy bin of the Z'.

Figures A.5 presents the c- and light-rejection at an inclusive 70% b-tagging WP. The equivalent information for c-tagging at a c-tagging WP of 30% is displayed in Figures A.8 and A.9 for b- and light-rejection.

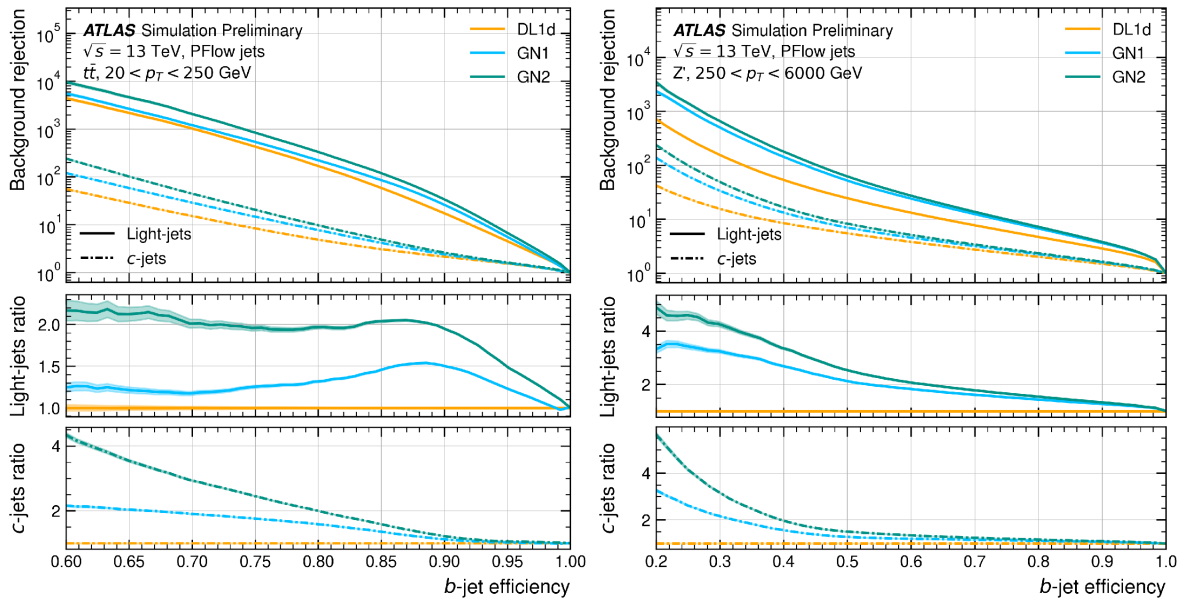


Figure A.2: The c- and light-rejections as a function of the b-jet tagging efficiency in the $t\bar{t}$ with $20 < p_T < 250$ GeV (left) and Z' with $250 < p_T < 6000$ GeV (right) test samples, from [17]. Models compared are DL1d in orange, GN1 in turquoise, and GN2 in blue. The bottom plots show the ratio with respect to the DL1d performance. Flavour fractions are set at $f_c^b = 0.018$ for DL1d, 0.05 for GN1, and 0.1 for GN2. Shaded regions represent the binominal error band.

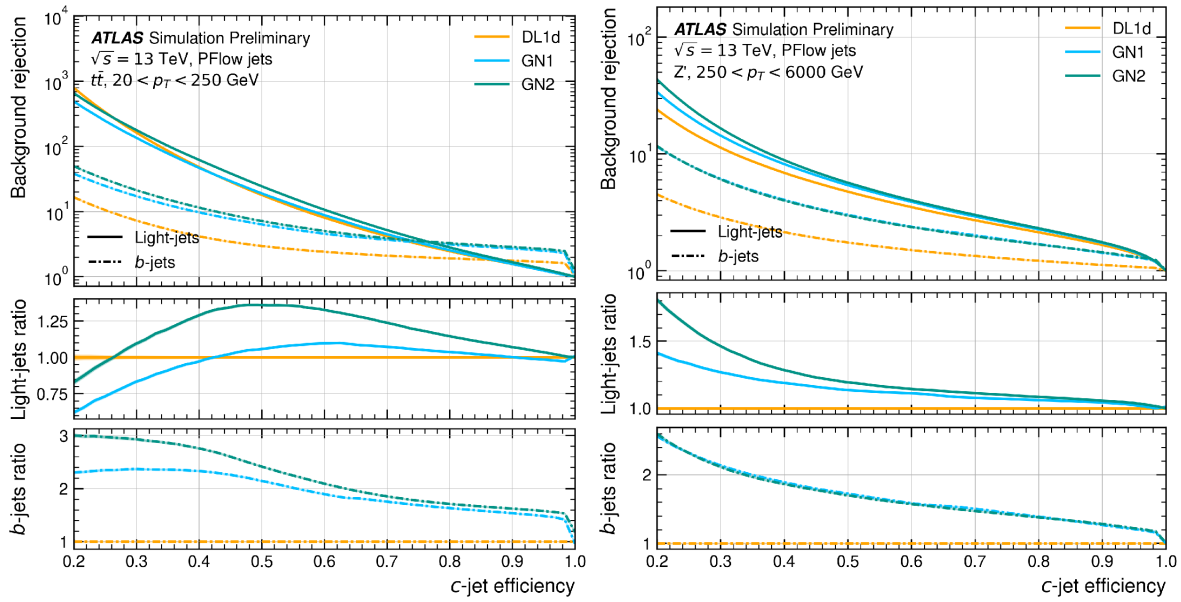


Figure A.3: The b- and light-rejections as a function of the c-jet tagging efficiency in the $t\bar{t}$ with $20 < p_T < 250$ GeV (left) and Z' with $250 < p_T < 6000$ GeV (right) test samples, from [17]. Models compared are DL1d in orange, GN1 in turquoise, and GN2 in blue. The bottom plots show the ratio with respect to the DL1d performance. Flavour fractions are set at $f_b^c = 0.2$ for all taggers. Shaded regions represent the binominal error band.

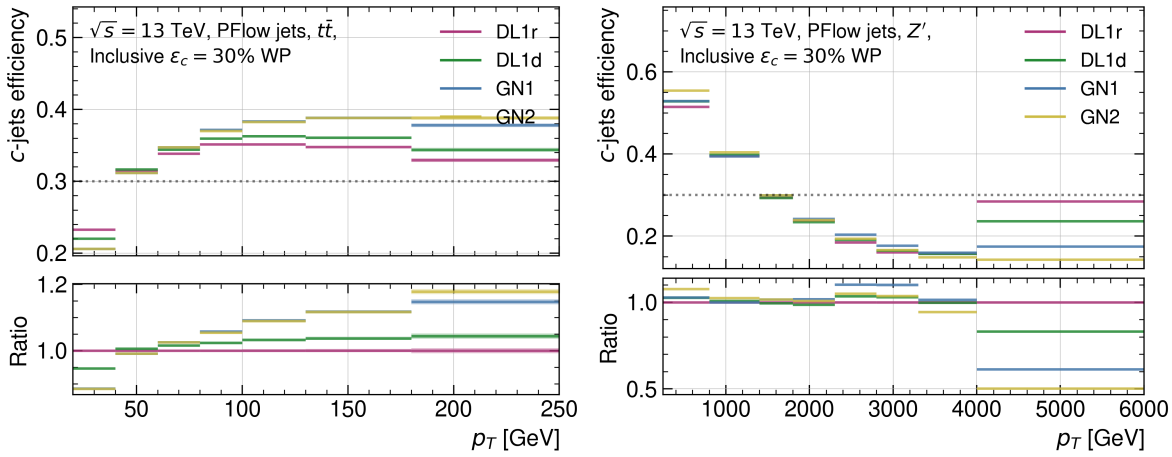


Figure A.4: Comparing the different models c-tagging efficiency as a function of jet p_T for the inclusive c-tagging 30% working point on the $t\bar{t}$ (left) and Z' (right). The flavour fraction is set at $f_b^c = 0.2$ for all taggers.

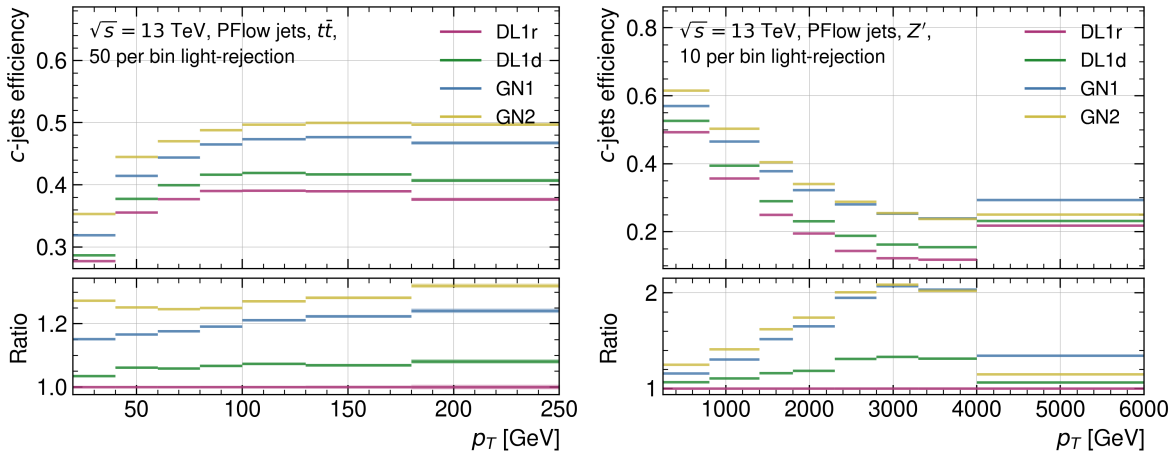


Figure A.5: Comparing the different models c-tagging efficiency as a function of jet p_T at a fixed light-jet rejection per bin of 50 for the $t\bar{t}$ (left) and 10 for the Z' (right) test samples. The flavour fraction is set at $f_b^c = 0.2$ for all taggers.

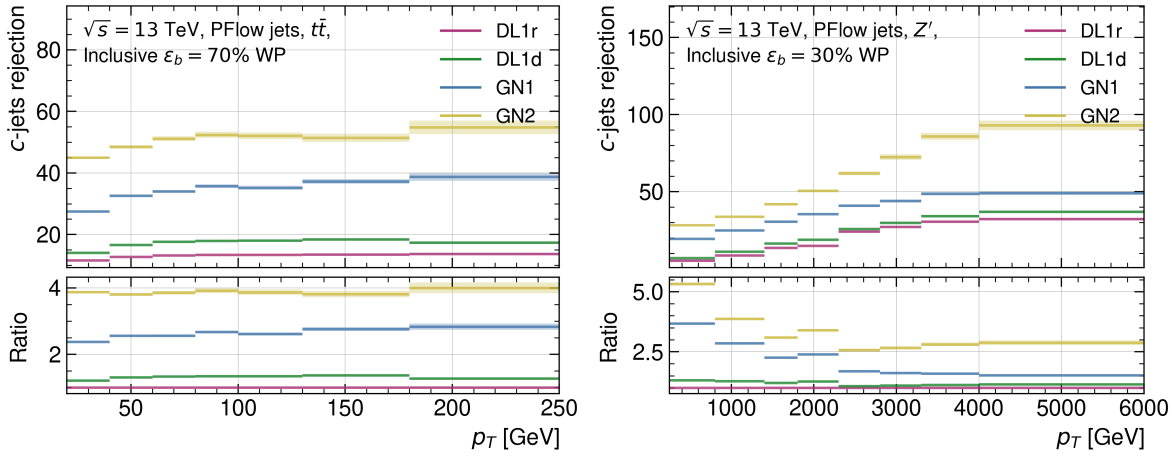


Figure A.6: Comparing the different models c-rejection as a function of jet p_T for the b-tagging inclusive 70% working point on the $t\bar{t}$ (left) and 30% working point on Z' (right). The flavour fraction is set at $f_c^b = 0.018$ for DL1r and DL1d, 0.05 for GN1, and 0.1 for GN2.

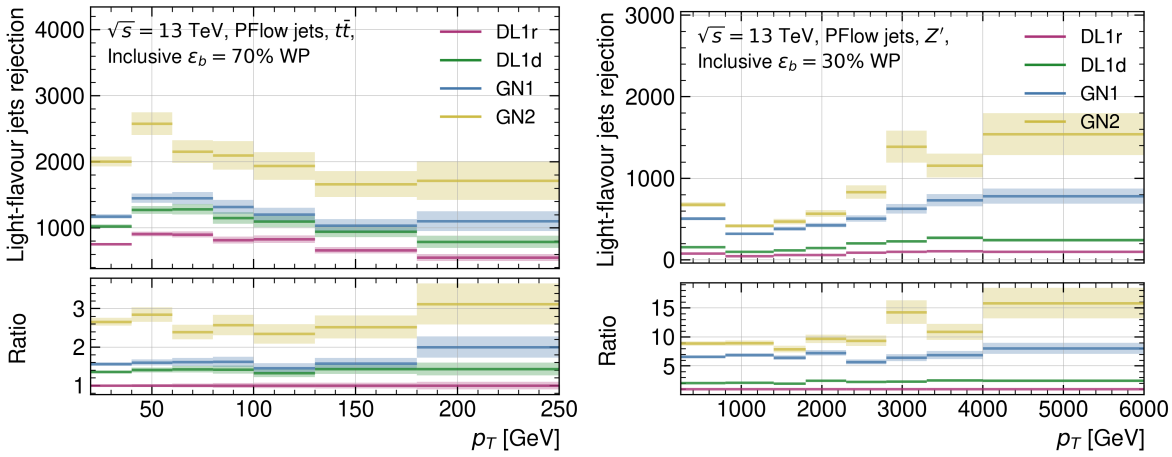


Figure A.7: Comparing the different models light-rejection as a function of jet p_T for the b-tagging inclusive 70% working point on the $t\bar{t}$ (left) and 30% working point on Z' (right). The flavour fraction is set at $f_c^b = 0.018$ for DL1r and DL1d, 0.05 for GN1, and 0.1 for GN2.

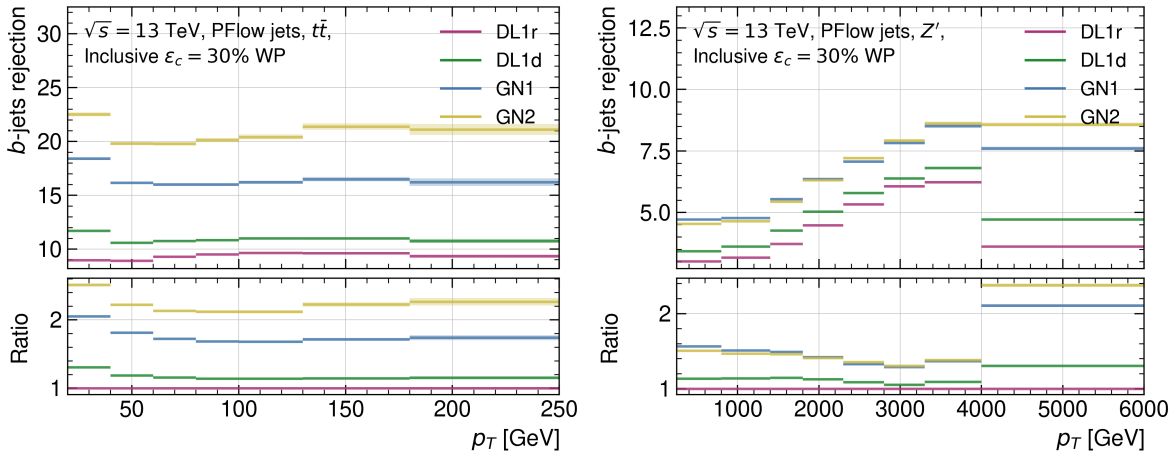


Figure A.8: Comparing the different models b-rejection as a function of jet p_T for the c-tagging inclusive 30% working point on the $t\bar{t}$ (left) and Z' (right). The flavour fraction is set at $f_b^c = 0.2$ for all taggers.

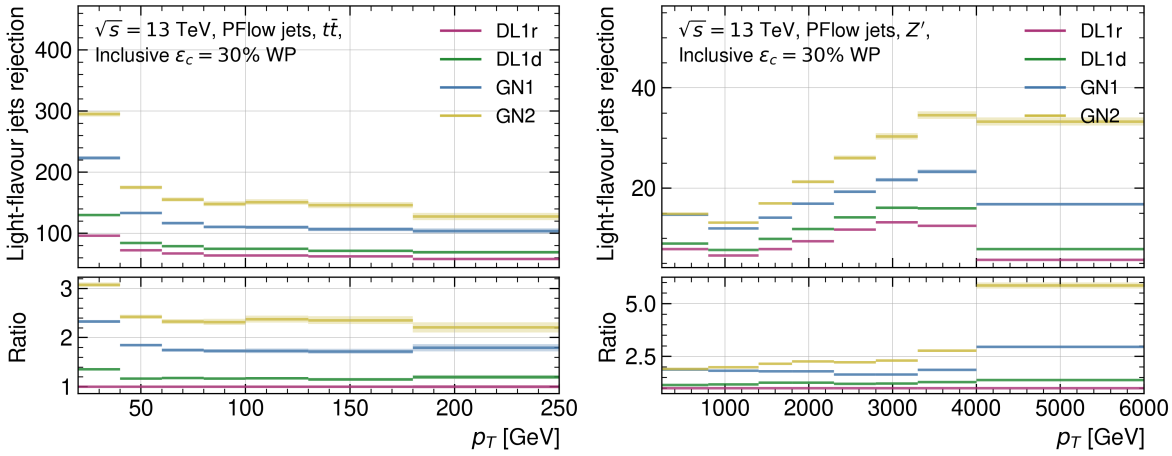


Figure A.9: Comparing the different models light-rejection as a function of jet p_T for the c-tagging inclusive 30% working point on the $t\bar{t}$ (left) and Z' (right). The flavour fraction is set at $f_b^c = 0.2$ for all taggers.