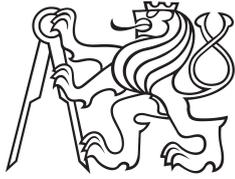


Master Thesis



Czech
Technical
University
in Prague

F4

Faculty of Nuclear Sciences and Physical Engineering
Department of Physics

Identifying Heavy-Flavor Jets Using Vectors of Locally Aggregated Descriptors

Bc. Georgij Ponimatkin

Supervisor: RNDr. Jana Bielčíková, Ph.D.
Supervisor–specialist: Dr. Ing. Josef Šivic
May 2021



Katedra: fyziky

Akademický rok: 2020/2021

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: Bc. Georgij Ponimatkin

Studijní program: Aplikace přírodních věd

Obor: Experimentální jaderná a částicová fyzika

Název práce: Identifikace jetů obsahujících těžký kvark pomocí vektorů lokálních
(česky) agregovaných deskriptorů

Název práce: Identifying Heavy-Flavor Jets Using Vectors of Locally Aggregated
(anglicky) Descriptors

Pokyny pro vypracování:

1. Úvod do problematiky studia jetů obsahujících těžké kvarky.
2. Úvod do strojového učení a počítačového vidění.
3. Přehled aktuálních výsledků v tagování po algoritmické stránce.
4. Aplikace JetVLAD modelu na simulované p+p srážky při těžišťové energii 200 GeV bez, resp. s termálním pozadím.
5. Diskuse získaných výsledků.

Práce bude vypracována v anglickém jazyce.

Doporučená literatura:

- [1] A. Zhang, Z. C. Lipton, M. Li, A. Smola, Dive into Deep Learning, 2020
interaktivní open source učebnice: <https://d2l.ai>.
- [2] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, J. Sivic, NetVLAD: CNN architecture
for weakly supervised place recognition, arXiv: 1511.07247.
- [3] L. Cunqueiro, M. Ploskon, Searching for the dead cone effects with iterative
declustering of heavy-flavor jets, Phys. Rev. D 99 (2019) 074027.
- [4] I. Goodfellow et al., Deep Learning, MIT Press, 2016.
- [5] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2011.
- [6] D. Guest et al., Jet Flavor Classification in High-Energy Physics with Deep Neural
Networks, Phys. Rev. D 94 (2016) 112002.

Jméno a pracoviště vedoucího diplomové práce:

RNDr. Jana Bielčíková, Ph.D.

FJFI ČVUT v Praze a Ústav jaderné fyziky AV ČR, v.v.i.

odborný konzultant: Dr. Ing. Josef Šivic, CIIRC, ČVUT v Praze

Datum zadání diplomové práce: 23.10.2020

Termín odevzdání diplomové práce: 03.05.2021

Doba platnosti zadání je dva roky od data zadání.

.....
garant oboru

.....
vedoucí katedry

.....
děkan

V Praze dne 23.10.2020

Acknowledgements

First, I would like to thank Jana Bielčíková for her thoughtful advising and for allowing me to have a freedom to pursue the research that I like this early. I also would like to thank Josef Šivic for his advising through machine learning and computer vision related topics and for fueling my further interest in the computer vision research. I would like to acknowledge Raghav Kunnawalkam Elayavalli and Jörn Putschke for teaching me many experimental aspects of jet physics and Monte-Carlo simulations. The research process would not have been fun without my colleagues at NPI and IMPACT CIIRC CTU, so I also owe them a big thanks for countless coffee breaks. One separate big thank belongs to Jan Kreps and the whole CIIRC IT department for their flawless operation of the CIIRC computational cluster. Finally, I would like to thank my family for their continuous support during my studies.

Declaration

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

Nemám závažný důvod proti použití tohoto školního díla ve smyslu § 60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze dne
Georgij Ponimatkin

Abstract

Identification of heavy flavor jets with high statistical precision is a crucial task needed for many physics analyses ranging from heavy-ion physics to new physics searches. Such precision can be achieved by using machine learning based tagging methods. For this purpose in this work a set based tagging model called JetVLAD is introduced. The performance of this model is evaluated using simulated p+p data at RHIC energies of $\sqrt{s} = 200$ GeV. At last, the effects of tracking efficiency, pileup and thermal background on model performance are studied. The resulting model achieves good performance across large jet transverse momentum (p_T) range from 5 to 40 GeV/ c with minor performance degradation caused by the effects of tracking efficiency and pileup. The JetVLAD model opens up the possibility of high precision heavy flavor measurements with lower dataset size requirements in comparison with standard methods.

Keywords: jet physics, jet tagging, machine learning, dead cone effect

Supervisor: RNDr. Jana Bielčíková, Ph.D.

Ústav jaderné fyziky AV ČR, v.v.i.

Abstrakt

Identifikace jetů pocházejících z rozpadů těžkých kvarků s vysokou statistickou přesností je klíčovou součástí mnoha fyzikálních analýz, od fyziky těžkých iontů po hledání nové fyziky. Taková přesnost může být dosažena s použitím metodik strojového učení. Za tímto účelem je v této práci představen tagovací model JetVLAD, který je založen na principu klasifikace množin. Výkon modelu je vyhodnocen na rekonstrukci jetů obsahujících těžký kvark v simulovaných p+p srážkách při energii $\sqrt{s} = 200$ GeV dosažitelné na urychlovači RHIC. Nakonec jsou studovány efekty účinnosti trackingu, pileupu a termálního pozadí na výkon modelu. Výsledný model dosahuje dobrého klasifikačního výkonu v rámci širokého intervalu příčných hybností (p_T) jetů od 5 do 40 GeV/ c s malou degradací výkonu v důsledku efektů účinnosti trackingu a pileupu. Model JetVLAD umožňuje provádět vysoce přesná měření jetů obsahujících těžký kvark s menšími požadavky na velikost naměřených dat oproti běžným metodám.

Klíčová slova: jetová fyzika, tagování jetů, strojové učení, dead cone efekt

Překlad názvu: Identifikace jetů obsahujících těžký kvark pomocí vektorů lokálních agregovaných deskriptorů

Contents

Introduction	1
1 Heavy Flavor Jets	3
1.1 Basics of Jet Physics	3
1.2 Experimental Reconstruction of Jets	5
1.2.1 k_T Algorithm Family	5
1.3 Lund Plane	6
1.4 Dead Cone Effect and Lund Plane	7
1.5 Recent Results of Heavy Flavor Jet Measurements in p+p Collisions ...	9
1.6 Jets in Heavy-Ion Collisions	12
2 Computer Vision and Machine Learning	15
2.1 Supervised Machine Learning ..	15
2.1.1 Overfitting, Underfitting, and Regularization	16
2.2 Evaluation Metrics	18
2.3 Neural Networks	18
2.4 Computer Vision, Descriptors and NetVLAD Layer	20
3 Overview of the Recent Results in Machine Learning Based Heavy-Flavor Jet Tagging	21
4 The JetVLAD Model	25
4.1 Formal Statement	25
4.2 JetVLAD Architecture	26
4.3 Dataset	27
4.4 Hyperparameter Optimization ..	29
4.5 Model Performance with Full Dataset and Background Effects ..	31
5 Conclusion	39
Bibliography	41
Contributions	47
Publications in Peer Reviewed Journals	47
Conference Proceedings	47
Talks at Conferences	47

Figures

<p>1.1 Lund plane regions and their relations to the jet evolution. Physical interpretation of different Lund plane regions (left). An example of simple shower mapping procedure in the Lund plane (right). Taken from [20]. 7</p> <p>1.2 Lund plane comparisons for p+p and Pb+Pb collisions. Lund plane generated by Pythia8 model (left), JEWEL model (middle) and difference between quenched JEWEL jet and vacuum JEWEL jet (right). Taken from [20]. 8</p> <p>1.3 R_{AA} factors for charmed mesons and inclusive hadrons. Taken from [28]. 9</p> <p>1.4 Flavor dependence of jet emissions. Ratio of charmed to light (left) and bottom to light (right) hadron level jet emission. The y axis shows the opening angle of the emission, while the x axis shows the energy of the radiator particle. Taken from [29]. 9</p> <p>1.5 Differential cross section of D^0 tagged jets in p+p collisions at $\sqrt{s} = 7$ TeV measured by the ALICE experiment. (left) The plot shows comparison of the experimental data with LO Monte Carlo generators. (right) The comparison of the experimental data with NLO effects. Notice how theoretical description improves in this case, which points at importance of higher order effects. Taken from [30]. 10</p>	<p>1.6 Substructure observables of D^0 tagged jets in p+p collisions at $\sqrt{s} = 7$ TeV measured by the ALICE experiment. (left) Groomed jet substructure of inclusive and D^0 tagged jets. (right) Ratio of D^0 tagged jets to inclusive jets logarithm of inverse of opening angle. Taken from [36]. 11</p> <p>1.7 Jet shapes in p+p collisions at $\sqrt{s} = 5.02$ TeV measured by the CMS experiment. (left) Jet shapes for inclusive jets. (middle) Jet shapes for b-jets. (left) Ratio of b-jet shapes to inclusive jet shapes. Taken from [37]. 11</p> <p>1.8 Dijet transverse momentum imbalance at $\sqrt{s} = 5.02$ TeV the CMS experiment. (left) Jet shapes for inclusive jets. (right) Jet shapes for b-jets. Taken from [38]. 12</p> <p>1.9 R_{CP} factors for charged jets and inclusive hadrons at RHIC and LHC energies. Taken from [39]. 13</p> <p>1.10 Large R measurements of the jet R_{AA} in Pb+Pb collisions at $\sqrt{S_{NN}} = 5.02$ TeV by CMS experiment. Taken from [40]. 13</p> <p>2.1 Effects of underfitting and overfitting. Different polynomial models fitted to data generated by $x^2 + \varepsilon$ process. (left) Underfitting by the linear model. (middle) Optimal model description of the data. (right) Overfitting phenomenon by the high-degree polynomial model. 17</p> <p>2.2 Graphical representation of the residual block. Taken from [53]. 19</p> <p>3.1 Feed forward architecture explored in [1]. 21</p>
---	--

3.2 Recursive architectures explored in [1]. (left) LSTM-based architecture. (right) Outer recursive architecture.	22
3.3 Rejection vs. efficiency graphs for different jet flavors. (left) Light quark rejection vs bottom quark efficiency graph with respect to different input feature combinations. (right) Charm quark rejection vs bottom quark efficiency graph with respect to different input feature combinations. Taken from [1].	22
3.4 DeepJet architecture schematics. Taken from [2].	23
3.5 Performance of the DeepJet tagger for different classes (b vs c, b vs udsg) as well as for different model architectures (DeepJet vs. DeepCSV). (left) Model performance for jets with $p_T > 30$ GeV/ c . (right) Model performance for jets with $p_T > 60$ GeV/ c . Taken from [2].	23
4.1 Example of some feature distributions. <i>udsg</i> (light) jets are represented by the green color, <i>c</i> (charm) jets are represented by the red color and <i>b</i> (beauty) jets are represented by the blue color.	30
4.2 Selection of input feature combinations - HardQCD dataset. (left) Efficiency vs. purity for different input feature combinations. (right) Efficiency vs. rejection for different input feature combinations. Each row shows different jet p_T ranges that were used to train the model.	34
4.3 Selection of input feature combinations - Balanced dataset. (left) Efficiency vs. purity for different input feature combinations. (right) Efficiency vs. rejection for different input feature combinations. Each row shows different jet p_T ranges that were used to train the model.	35
4.4 Optimal hyperparameter value selection. (left column) Integrated efficiencies vs. number of clusters (upper left) and depth (lower left). (right column) Integrated purities vs. number of clusters (upper right) and depth (lower right). The optimal parameters are given by $N_c = 33$ and $D = 4$	36
4.5 JetVLAD model performance across different jet p_T ranges. (left) Purity vs. efficiency for different jet p_T ranges. (right) Rejection vs. efficiency for different jet p_T ranges. Colors represent different p_T bins used to train model.	36
4.6 Effects of tracking efficiency on the JetVLAD model performance. (left) Purity vs. efficiency for different tracking efficiency values. (right) Rejection vs. efficiency for different tracking efficiency values. Colors represent tracking efficiency values.	37
4.7 Effects of pileup on the JetVLAD model performance. (left) Purity vs. efficiency for model evaluated on training data (dotted) and model evaluated on the data with pileup (dashed). (right) Rejection vs. efficiency for model evaluated on training data (dotted) and model evaluated on the data with pileup (dashed). Colors represent tracking efficiency values.	37

4.8 Effects of thermal background on JetVLAD model performance. (left) Purity vs. efficiency for default model evaluated on thermal background data (red) and model retrained on the thermal data (yellow). (right) Rejection vs. efficiency for model evaluated on thermal background data (red) and model retrained on the thermal data (yellow). This plot is not showing the default model performance since this would not lead to fair comparison. The $p_{T,jet}$ here denotes the p_T of a jet without thermal background. . . 38

Tables

4.1 Kinematic ranges used for dataset generation within Pythia8 model. 27

4.2 Model input variables per track. Here p_i denotes momentum vector components of a particle, d_i^p are secondary vertex coordinates of a particle and v_i are primary vertex coordinates. 30

4.3 Purity and rejection values for different jet p_T bins at fixed efficiency operating points of 50% and 80%. . 32



Introduction

Accurate measurement of jets containing a heavy quark (c or b), commonly referred as to heavy flavor jets, is a fundamental task that is required for a multitude of physics measurements - from Higgs boson decays in $H \rightarrow b\bar{b}$ channel, the search for new physics and last, but not least, precise studies of the quantum chromodynamics properties. In all cases, one needs to properly identify the heavy flavor jet. Most of the time, this is done by reconstructing a heavy flavor meson or hadron within the jet. More advanced techniques utilize statistical templates or primitive machine learning algorithms to mitigate the need for manual reconstruction.

With the recent dynamic development in the field of machine learning, the question arises - to what extent can those modern techniques be applied in the task of heavy flavor jet identification? Most of the current machine learning models for heavy flavor jet identification are built upon two ideas. The first is that it takes a fixed number of input features, thus potentially limiting the classification power of the algorithm [1]. Another approach tries to treat jets as a sequence of transverse momentum (p_T) or vertex ordered particles [1], [2]. While being more powerful than the previous class, those models introduce a nonphysical ordering of particles. To solve this issue, we introduce the JetVLAD model [3], a novel set-based tagging algorithm that allows to mitigate the drawbacks of both previous model classes.

This work is organized as follows: the first chapter deals with motivation for heavy flavor jet studies at medium energies. The second chapter briefly introduces basics of computer vision and machine learning. The third chapter reviews recent results on heavy flavor jet tagging with machine learning methods. The fourth, last chapter, is dedicated to author's contribution to the development of the JetVLAD tagging model and its applications.

Chapter 1

Heavy Flavor Jets

Every time we wish to measure a quark or gluon in a final state of some scattering process, we have to experimentally reconstruct and measure jets - which manifest themselves as a collimated shower of highly energetic particles. Because of that, jets and their properties are one of the most important observables in many subareas of particle and nuclear physics - be it energy loss mechanisms in heavy-ion collisions or searches for new physics. In this chapter, the basics of jet physics will be introduced, with a focus on heavy-flavor jets, since they play an important part in the development of the JetVLAD model, especially in the context of current and future high-precision measurements of heavy quark production in hot and dense QCD matter created in heavy-ion collisions at large accelerator facilities such as RHIC at Brookhaven National Laboratory or the LHC at CERN.

1.1 Basics of Jet Physics

One of the most important predictions of Quantum Chromodynamics (QCD) is the existence of jets - highly energetic, collimated showers of particles that are created in high- Q^2 processes, where Q^2 is the square of the four-momentum transfer. In those processes, the final state quarks and gluons have a large virtuality, which later on leads to the extensive partonic radiation. This radiation sheds off virtuality from the outgoing partons up to the values of $Q^2 \sim \Lambda_{QCD}^2$, where the parameter Λ_{QCD} describes the scale at which non-perturbative transition from partons to hadrons happens, with current estimated value being $\Lambda_{QCD} \sim 220$ MeV [4]. The resulting hadronic cascade is then manifested as a jet within the detector.

To calculate cross-section of perturbative processes in proton-proton (p+p) collisions, the QCD factorization theorem [5] can be used and the cross section is then commonly expressed in the form

$$\sigma = \frac{1}{N_f} \sum_i^{N_f} f_{q_i/p_1}(x_a, Q^2) \otimes f_{q_i/p_2}(x_b, Q^2) \otimes \hat{\sigma}_{ab \rightarrow cd} \otimes D(c \rightarrow h) \otimes D(d \rightarrow h). \quad (1.1)$$

Here x_a, x_b denote shared momentum fraction of the parton in the incoming process, $f_{q_i/p_j}(x_i, Q^2)$ are parton distribution functions for quark flavor i

in proton j that encode density of the partons within the incoming proton, $\hat{\sigma}_{ab \rightarrow cd}$ is parton level cross-section for $2 \rightarrow 2$ process obtained by the means of perturbation theory calculation, and D is a fragmentation function, which encodes the probability that the outgoing parton will transfer into hadron h after hard scattering. This function serves as a "black-box" approximation for the hadronization process, which is still not understood due to its non-perturbative nature.

The values of f and D used in this formula are fixed for each value of the Q^2 scale. To change the scale (and hence the behavior of these functions) one can use the DGLAP evolution equations [6]–[8], given for quark distributions by

$$\frac{\partial f_{q_i}(x, Q^2)}{\partial \ln(Q^2)} = \frac{\alpha_s(Q^2)}{2\pi} \int_x^1 \frac{dy}{y} \left[f_{q_i}(y, Q^2) P_{q \rightarrow qg} \left(\frac{x}{y} \right) + g(y, Q^2) P_{g \rightarrow q\bar{q}} \left(\frac{x}{y} \right) \right], \quad (1.2)$$

and for gluon distributions by

$$\frac{\partial g(x, Q^2)}{\partial \ln(Q^2)} = \frac{\alpha_s(Q^2)}{2\pi} \int_x^1 \frac{dy}{y} \left[\sum_{i=1}^{N_f} f_{q_i}(y, Q^2) P_{q \rightarrow gq} \left(\frac{x}{y} \right) + g(y, Q^2) P_{g \rightarrow gg} \left(\frac{x}{y} \right) \right]. \quad (1.3)$$

Here $f_{q_i}(x, Q^2)$ denotes the parton distribution function for a quark of flavor i and $g(x, Q^2)$ is a gluon distribution function. The $P_{i \rightarrow jk}$ is **QCD splitting function** that describes the probability of parton radiation with certain momentum fraction and $\alpha_s(Q^2)$ is a strong coupling constant at scale Q^2 , which value is given by

$$\alpha_s(Q^2) = \frac{12\pi}{(33 - 2N_f) \ln \left(\frac{Q^2}{\Lambda_{QCD}^2} \right)}. \quad (1.4)$$

Here the N_f is a number of quark flavors available at current scale.

The splitting functions play an important role within the parton-shower description of a jet evolution, which is a semiclassical approximation of the real jet evolution. In the leading order (LO) perturbative QCD expansion [6]–[8] they are given as

$$P_{q \rightarrow qg}(z) = C_F \left(\frac{1+z^2}{1-z} \right)_+, \quad (1.5)$$

$$P_{q \rightarrow gq}(z) = C_F \frac{1+(1-z)^2}{z}, \quad (1.6)$$

$$P_{g \rightarrow q\bar{q}}(z) = \text{Tr} \left(z^2 + (1-z)^2 \right), \quad (1.7)$$

$$P_{g \rightarrow gg}(z) = 2C_A \left(\frac{1}{(1-z)_+} + \frac{1-z}{z} + z(1-z) \right), \quad (1.8)$$

with z being the shared momentum fraction during splitting process, and $C_F = 4/3$ and $C_A = 3$ being colour factors, needed to take into account the

colour charge structure of QCD. This framework allows one to describe the jet evolution at the parton level, starting from the highly virtual parton and producing a cascade of emissions up to the $Q^2 \sim \Lambda_{QCD}^2$ level.

1.2 Experimental Reconstruction of Jets

The experimental reconstruction of jets relies on the jet clustering algorithms. Of those, the most prominent role is played by the sequential recombination algorithms [9]–[12]. To be well behaved, the jet clustering algorithm is required to have the following properties:

- **Collinear Safety** - in the case of collinear parton splitting during evolution, the algorithm of interest should produce the same jet.
- **Infra-Red Safety** - addition of particles with small p_T should produce the same number of jets as without them.
- **Process Independence** - jet algorithm should be independent of any underlying physical process as well as the level of description (partonic vs. hadronic level). The latter means that the reconstructed parton-level jet and its corresponding particle-level jet should have the same properties.

1.2.1 k_T Algorithm Family

The most widely used jet algorithms nowadays are those of the k_T family, implemented within FastJet package [13], [14] and represented by the k_T [15], anti- k_T [16] and C/A algorithms [17], [18]. Those algorithms operate in the (η, φ, p_T) space (with η being pseudorapidity, φ being azimuthal angle and p_T being particle transverse momentum). Below we describe the individual steps of these types of jet algorithms:

1. For all particles distance $d_i^2 = p_{T,i}^{2k}$ is calculated.
2. For each pair of particles i, j the distance $d_{ij} = \min\{p_{T,i}^{2k}, p_{T,j}^{2k}\} \frac{(\eta_i - \eta_j)^2 + (\varphi_i - \varphi_j)^2}{R^2}$ is calculated.
3. Minimum distance $d_{\min} = \min\{d_i, d_{ij}\}$ is obtained.
4. If $d_{\min} = d_{ij}$ then the particles i and j are merged into one and the procedure goes back to step 1. Otherwise, if $d_{\min} = d_i$, the obtained object is a jet.

Here, R is a jet resolution parameter that controls the size of a reconstructed jet.

By varying the value of the k parameter, we obtain the following algorithms:

- For $k = 1$ we obtain the k_T algorithm [15], which reconstructs jets with a p_T -ordered substructure. Jets found by the k_T algorithm are biased towards soft particles, have irregular shapes, and are generally used to estimate background effects on jets.

- For $k = 0$ we obtain the Cambridge-Aachen algorithm (C/A) [17], [18]. This algorithm results in jets having angular-ordered substructure, which is natural to QCD. The algorithm is the best candidate for reconstruction of jets for substructure studies.
- For $k = -1$ we obtain the anti- k_T algorithm [16]. This algorithm is biased towards hard particles and leads to jets with circular shapes. Its substructure is nonphysical, due to the clustering around the hard core particle. Nonetheless, this algorithm is most often used for the reconstruction of jets in the experiment where knowledge of the substructure is not needed (for example, inclusive jet spectra analysis).

1.3 Lund Plane

In the last few years, the Lund plane formalism [19] became a prime candidate for measuring jet radiation patterns both in p+p as well as heavy ion collisions [20]. The whole approach assumes that the jet of interest is reconstructed with the C/A algorithm and hence is reconstructed with respect to angular ordering, which is natural to QCD. Then the clustering history of the reconstructed jet is traversed back to the hardest branch. Assuming we have two branches a and b with their corresponding transverse momenta $p_{T,a}$ and $p_{T,b}$, such that $p_{T,a} > p_{T,b}$, we can construct a set of observables

$$\Delta = \sqrt{(\eta_a - \eta_b)^2 + (\phi_a - \phi_b)^2}, \quad (1.9)$$

$$k_T = p_{T,b} \Delta_{ab}, \quad (1.10)$$

$$z = \frac{p_{T,b}}{p_{T,a} + p_{T,b}}, \quad (1.11)$$

where η_i is the pseudorapidity coordinate of the subjet i and φ_i is the azimuthal coordinate of the subjet i . Adding a cut on the z value as

$$z_g = \frac{p_{T,b}}{p_{T,a} + p_{T,b}} > 0.1, \quad (1.12)$$

we can introduce the SoftDrop technique [21] which looks for a split, that satisfies the condition shown above. If such split is found, the jet is accepted, otherwise it is discarded. This greatly reduces the soft wide angle radiation within the jet and brings the substructure observables closer to the parton level.

By recording the full jet clustering history over many jets, we can arrive to an average Lund plane, which can be expressed in the double-differential form as

$$\rho(\Delta, k_T) = \frac{1}{N_{jets}} \frac{dN}{d \log(k_T) d \log(1/\Delta)}. \quad (1.13)$$

Here N_{jets} is a number of jets used to construct the emission diagram and $\frac{dN}{d \log(k_T) d \log(1/\Delta)}$ is a number of jets with particular distance between subjets $\log(1/\Delta)$ and emission strength $\log(k_T)$. This distribution can be expressed

at the leading order of perturbative QCD in the collinear approximation ($\Delta \ll 1$) as

$$\rho \simeq \frac{\alpha_S(k_T)C_F}{\pi} \bar{z} (P_{gq}(\bar{z}) + P_{gq}(1 - \bar{z})), \quad \bar{z} = \frac{k_T}{p_{T,jet}\Delta}, \quad (1.14)$$

with $\alpha_S(k_T)$ being the strong coupling constant dependent on k_T and $C_F = \frac{4}{3}$ being the colour factor. We can see that the Lund plane diagram is directly connected to the framework of parton shower evolution and splitting functions and thus is able to directly quantify the mechanisms behind it as well as the possible effect of QGP medium on the energy loss within jets. Examples of different Lund plane regions and their explanation can be seen on Fig. 1.1. Examples of Lund plane generated by `Pythia8` model [22], [23] (p+p collision), `JEWEL` model [24]–[26] (Pb+Pb collisions) and the difference between them can be seen on Fig. 1.2. The `Pythia8` is a LO Monte-Carlo generator for p+p collisions which incorporates Lund fragmentation model and p_T ordered parton showers. The `JEWEL` generator is a modification of the `Pythia6` Monte-Carlo model, which incorporates jet quenching effects due to the propagation in quark-gluon plasma. The difference between simulated p+p and Pb+Pb collisions, shown in the rightmost plot in Fig. 1.2 shows modification of jet emission due to the dense nuclear medium.

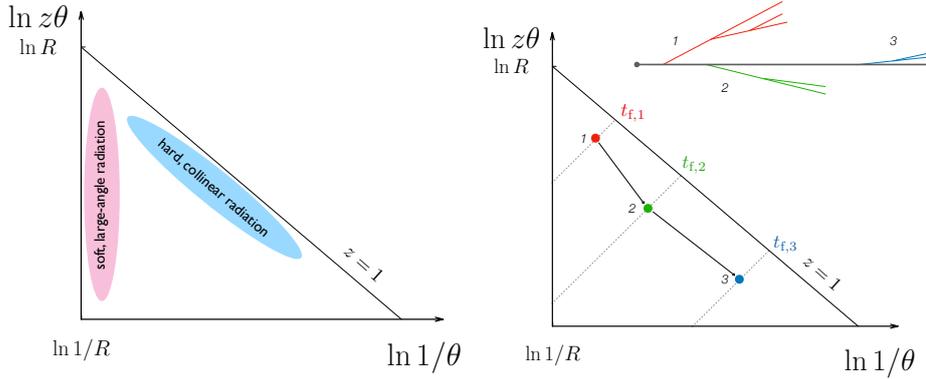


Figure 1.1: Lund plane regions and their relations to the jet evolution. Physical interpretation of different Lund plane regions (left). An example of simple shower mapping procedure in the Lund plane (right). Taken from [20].

1.4 Dead Cone Effect and Lund Plane

The gluon radiation spectrum in the medium can be written in the form of a modified Bethe-Heitler spectrum [27] as

$$\frac{dW}{d\omega} \simeq \frac{\alpha_S C_F}{\pi\omega} \sqrt{\frac{\omega_1}{\omega}}, \quad \omega < \omega_1 \equiv \hat{q}L^2. \quad (1.15)$$

Here \hat{q} is QGP transport coefficient and L is a path traversed by parton in the medium. This spectrum describes energy distribution of gluons emitted by

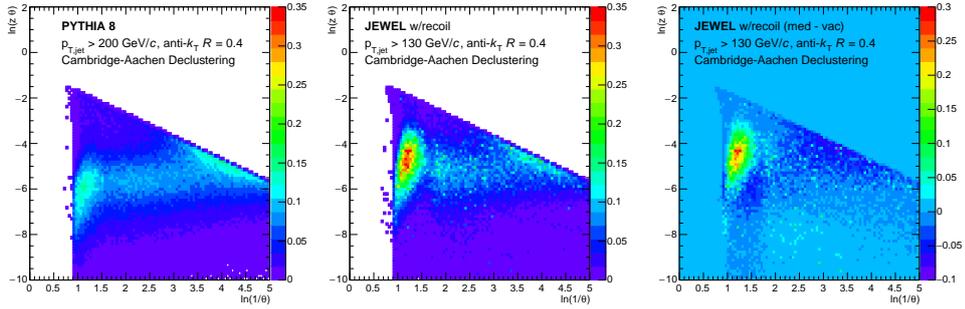


Figure 1.2: Lund plane comparisons for p+p and Pb+Pb collisions. Lund plane generated by Pythia8 model (left), JEWEL model (middle) and difference between quenched JEWEL jet and vacuum JEWEL jet (right). Taken from [20].

the massless quark in the medium. If one were to consider heavy-quarks (i.e. those such that $m_q^2 \gg \Lambda_{QCD}^2$) the Bethe-Heitler spectrum has to be modified to the form [27]

$$\frac{dW}{d\omega} \simeq \frac{\alpha_s C_F}{\pi \omega} \sqrt{\frac{\omega_1}{\omega}} \frac{1}{\left[1 + \hat{q}^{-1/3} \left(\frac{M}{E}\right)\right]^{2/3}}. \quad (1.16)$$

The addition of the later factor leads to the presence of so-called dead cone effect, which results in the suppression of the small angle radiation by heavy quarks. As a consequence of that, the energy loss of heavy quarks in the medium should be smaller than for lighter quarks. For a long time, the dead cone effect was evading experimental observations, most probably due to the fact that "standard" observables are not sensitive enough to observe it. One such observable is a nuclear modification factor R_{AA} , given by

$$R_{AA}(p_T) = \frac{1}{\langle N_{coll} \rangle} \frac{dN_{AA}^{jet}/dp_T dy}{dN_{pp}^{jet}/dp_T dy}. \quad (1.17)$$

Here $\langle N_{coll} \rangle$ is an average number of binary nucleon-nucleon collisions, the term $dN_{AA}^{jet}/dp_T dy$ is a jet yield in A+A collision and $dN_{pp}^{jet}/dp_T dy$ is a jet yield in p+p collisions. For $R_{AA} < 1$ we say that we observe the suppression, for $R_{AA} > 1$ we say that we observe an enhancement. As an example, Fig. 1.3 shows measurements of charmed meson R_{AA} factors both at RHIC and LHC energies and their comparison with inclusive hadron R_{AA} measurements. Notice, the small differences between the R_{AA} for D^0 meson and inclusive hadrons.

Recent research shows that Lund plane could be a tool that will finally allow to access the experimental signatures of the dead cone effect [29] and thus answer if it exists. The predicted ratios of hadron level jet emission angles dependent on the radiator energy for charmed hadrons to inclusive hadrons and bottom hadrons to inclusive hadrons can be seen on Fig. 1.4. Indeed, the direct measurement of Lund planes both for light and heavy-flavor

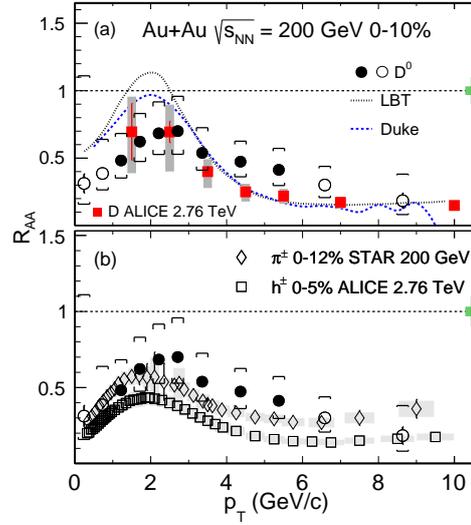


Figure 1.3: R_{AA} factors for charmed mesons and inclusive hadrons. Taken from [28].

jets will provide an unprecedented constraint for theory models that will hopefully allow us to more deeply understand the inner workings of the strong interaction.

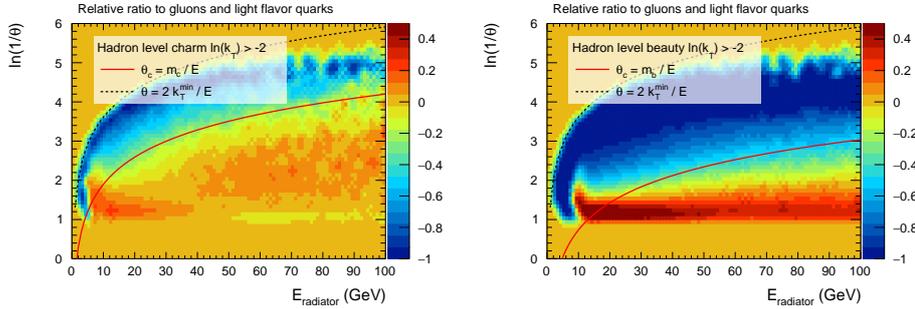


Figure 1.4: Flavor dependence of jet emissions. Ratio of charmed to light (left) and bottom to light (right) hadron level jet emission. The y axis shows the opening angle of the emission, while the x axis shows the energy of the radiator particle. Taken from [29].

1.5 Recent Results of Heavy Flavor Jet Measurements in $p+p$ Collisions

In this section, we will briefly review the recent experimental results on heavy-flavor jet measurements in $p+p$ collisions.

The measurements from the ALICE experiment explore properties of c -jets, which are built upon D meson reconstruction within a charged-particle jet,

i.e. the jet reconstructed solely from charged-particle tracks. The transverse momentum of D^0 meson tagged jets in p+p collisions at $\sqrt{s} = 7$ TeV [30] and its comparison with Monte-Carlo models can be seen in Fig. 1.5. Notice that basic Monte Carlo models (Pythia6 [22], Pythia8 [23] and Herwig7 [31], [32], which is a LO Monte-Carlo generator for p+p collisions that uses the clustering hadronization mechanism) are overestimating the resulting distribution. This could point to the fact that higher order effects become important for heavy flavor jets and a simple LO description, which is provided by most Monte Carlo generators is insufficient. This is confirmed in the right part of Fig. 1.5, where addition of next-to-leading order (NLO) effects within POWHEG [33]–[35] model improves the description. The POWHEG is a computational framework for implementation of NLO calculations for further use in Monte-Carlo generators.

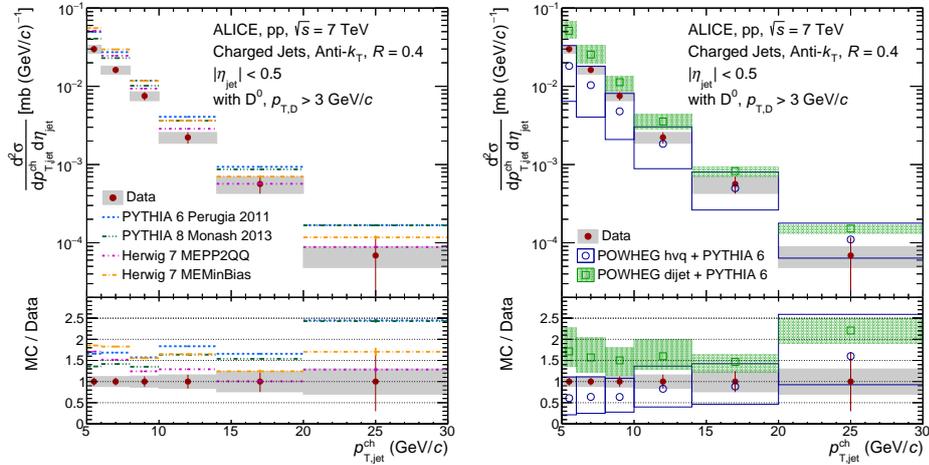


Figure 1.5: Differential cross section of D^0 tagged jets in p+p collisions at $\sqrt{s} = 7$ TeV measured by the ALICE experiment. (left) The plot shows comparison of the experimental data with LO Monte Carlo generators. (right) The comparison of the experimental data with NLO effects. Notice how theoretical description improves in this case, which points at importance of higher order effects. Taken from [30].

Next set of c -jet measurements are those of the differential jet substructure of D^0 tagged jets [36] in p+p collisions, which can be seen on Fig. 1.6. Left figure shows the measurement of a jet splitting function, which tells us that the D^0 tagged jets have more asymmetric splitting than inclusive jets, which is an effect of a large mass of a c quark. The right plot shows the distribution of $\log(1/\theta)$ (logarithm of inverse of opening angle) for ratio of D^0 tagged jets to inclusive jets. The fact that the ratio is smaller than 1 is a possible experimental hint that is pointing to the existence of **dead cone effect**.

The available results from the CMS experiment are centered around studies of b -jet properties. First we discuss b -jet shapes [37], which measure track dispersion within the jet. The results are shown in Fig. 1.7. It can be concluded, that b -jets are much wider than the inclusive jets, which is a

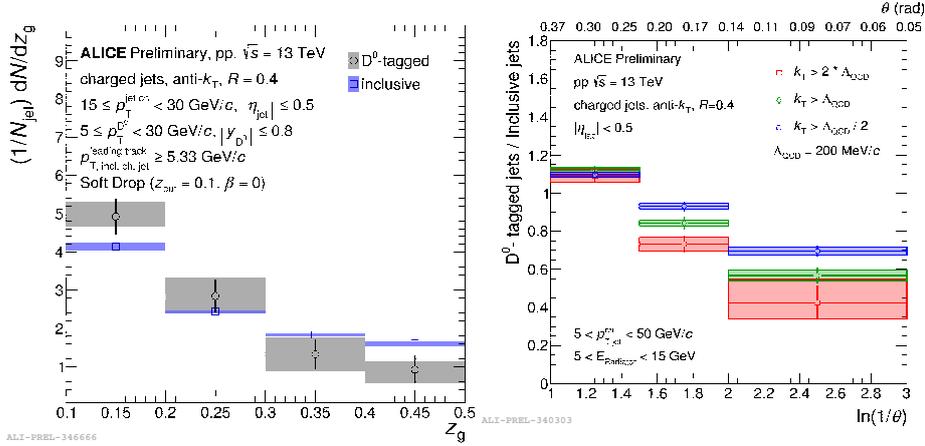


Figure 1.6: Substructure observables of D^0 tagged jets in p+p collisions at $\sqrt{s} = 7$ TeV measured by the ALICE experiment. (left) Groomed jet substructure of inclusive and D^0 tagged jets. (right) Ratio of D^0 tagged jets to inclusive jets logarithm of inverse of opening angle. Taken from [36].

consequence of high b quark mass.

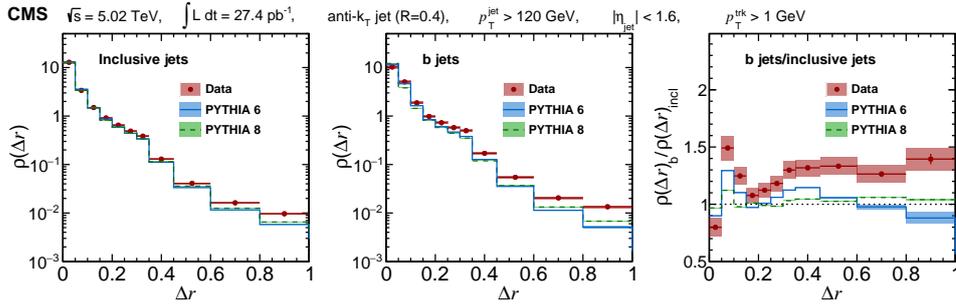


Figure 1.7: Jet shapes in p+p collisions at $\sqrt{s} = 5.02$ TeV measured by the CMS experiment. (left) Jet shapes for inclusive jets. (middle) Jet shapes for b -jets. (left) Ratio of b -jet shapes to inclusive jet shapes. Taken from [37].

The second result from CMS which we discuss here is related to the dijet imbalance [38] for inclusive jets and b -jets. The dijet imbalance is expressed by

$$x_J = \frac{p_{T, \text{sub}}}{p_{T, \text{lead}}}, \quad (1.18)$$

with $p_{T, \text{lead}}$ being the largest transverse momentum of a jet in the event and $p_{T, \text{sub}}$ being the second largest transverse momentum of a jet in the event. The results can be seen on Fig. 1.8. The identical momentum imbalance of b -jets and inclusive jets signifies flavor independence of the outgoing parton kinematics in p+p collisions.

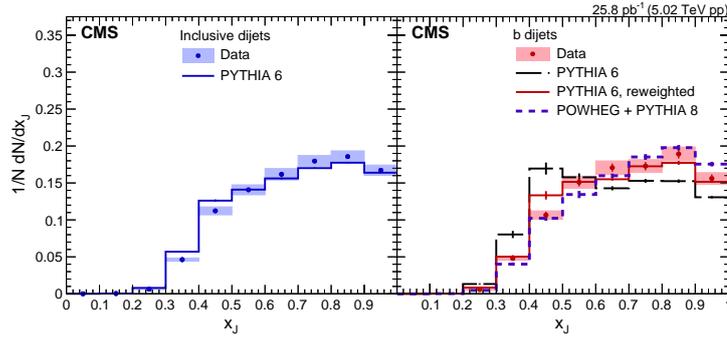


Figure 1.8: Dijet transverse momentum imbalance at $\sqrt{s} = 5.02$ TeV the CMS experiment. (left) Jet shapes for inclusive jets. (right) Jet shapes for b -jets. Taken from [38].

1.6 Jets in Heavy-Ion Collisions

Quark-Gluon Plasma (QGP), is a new state of matter that is believed to have existed in the first few microseconds after the Big Bang. Despite being an active area of research, still a lot is unknown about this state of matter and thus a lot of questions remain unanswered. Of particular interest is the question about the energy loss mechanisms of partons in this medium. This can be experimentally quantified, for example, via R_{AA} nuclear modification factor, shown in Eq. 1.17. Another way to compare suppression is via R_{CP} factor, which is defined as

$$R_{CP} = \frac{\langle N_{coll}^{per} \rangle}{\langle N_{coll}^{cent} \rangle} \frac{dN_{cent}^{jet}/dp_T dy}{dN_{per}^{jet}/dp_T dy} \quad (1.19)$$

Here $\langle N_{coll}^{per} \rangle$ is an average number of binary nucleon-nucleon collisions in peripheral collisions, $\langle N_{coll}^{cent} \rangle$ is an average number of binary nucleon-nucleon collisions in central collisions, and terms $dN_{cent}^{jet}/dp_T dy$, $dN_{per}^{jet}/dp_T dy$ are jet yields in central and peripheral collisions, respectively. R_{CP} factor is an analogue of the R_{AA} factor with the main difference being that instead of the p+p collision reference it uses the peripheral collisions, since we expect them to be close to the p+p collisions. One can observe, that there is a significant suppression happening at both energies, and despite ~ 10 times higher energy, the LHC measurements manifest a similar level of suppression as measurements at RHIC.

Another series of LHC measurements by CMS [40] and ATLAS [41] experiments deal with the large R jet measurements which allow to answer the question about the QGP medium effects on the jet broadening. The results by the CMS collaboration can be seen in Fig. 1.10. The data show a strong suppression for high- p_T jets in the most central collisions even for jets reconstructed with large R . This implies that a significant amount of jet energy is scattered to large angles.

We hope that this chapter provides a sufficient motivation that heavy-flavor jets are one of the keys to understanding parton energy loss mechanisms in

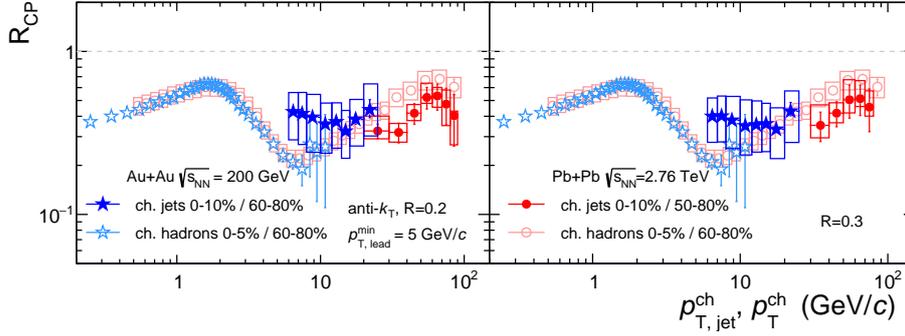


Figure 1.9: R_{CP} factors for charged jets and inclusive hadrons at RHIC and LHC energies. Taken from [39].

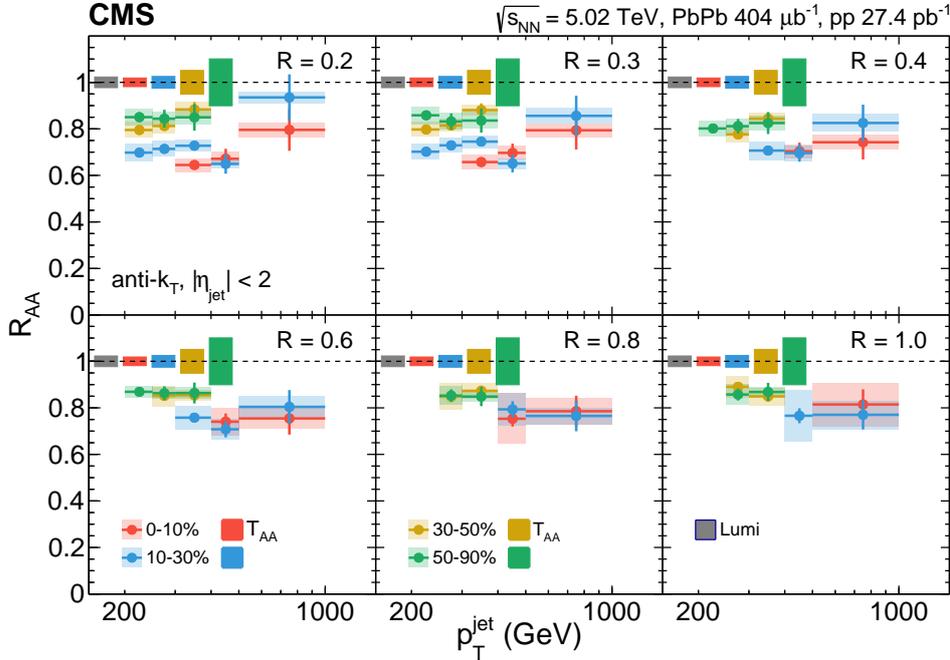


Figure 1.10: Large R measurements of the jet R_{AA} in Pb+Pb collisions at $\sqrt{s_{NN}} = 5.02$ TeV by CMS experiment. Taken from [40].

Quark-Gluon Plasma. Even in the inclusive case, experimental measurements of those effects require a lot of data, with heavy-flavor measurements requiring even more data due to the smaller cross-sections of heavy quark production. Thus, a standard experimental method for heavy-flavor jet identification, such as charmed/bottom-hadron reconstruction within jet, could be the limiting factor in the precision of the experimental results and the new approach to measurements is needed.

Chapter 2

Computer Vision and Machine Learning

Machine learning is a subfield of computer science that deals with algorithms that are able to learn from the data. These methods have a plethora of applications in many areas of science and engineering, including physics. In general, it is hard to introduce machine learning in one chapter, so we refer the reader to [42]–[44] for a more thorough introduction.

Computer vision [45] is a subfield of computer science that often utilizes machine learning to solve problems related to visual interpretation of the real world. Real-world applications include, for example, classification and detection of objects in pictures, reconstruction of 3D shapes from monocular inputs, and video analysis. Recent surge of deep learning methods led to fast-paced development of novel computer vision methods, which achieved significant performance compared to "classical" methods.

2.1 Supervised Machine Learning

In this section, the basic notions of supervised machine learning will be introduced. While there are other types of machine learning (unsupervised, self-supervised etc.), the JetVLAD model [3], which is the foundation of this thesis, is a fully supervised model. Given two sets \mathcal{X} and \mathcal{Y} , where \mathcal{X} is a set of inputs and \mathcal{Y} is a set of labels that can be summarized as a set $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, our goal is to find a map f from this dataset such that

$$\mathcal{X} \xrightarrow{f} \mathcal{Y}. \quad (2.1)$$

If the output is set $\mathcal{Y} \subseteq \mathbb{R}^n$ we call our task regression, if our output set $\mathcal{Y} \subseteq \{0, 1, \dots, n\}$ we call the task classification. There are many ways to construct this mapping, with the most popular approach using parametric models. In this approach, we construct a mapping f such that

$$y = f(\mathbf{x}; \boldsymbol{\theta}), \quad (2.2)$$

here $\boldsymbol{\theta}$ is a vector of parameters, which serve as degrees of freedom. Those degrees of freedom can be used in conjunction with the available dataset to learn our desired map. To do so, one has to select a loss function L , which

measures error between predicted value \hat{y} and ground-truth value y , i.e., in our notation

$$L \equiv L(f(\mathbf{x}; \boldsymbol{\theta}), y) = L(\hat{y}, y). \quad (2.3)$$

Our optimal parametrization $\boldsymbol{\theta}^*$ is then given as

$$\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta}} \mathbb{E}[L(f(\mathbf{x}; \boldsymbol{\theta}), y)], \quad (2.4)$$

where the expectation is taken over the whole dataset, which we assume is i.i.d. - independent and identically distributed. In practical terms, this means that we use an empirical distribution and our optimal parametrization is then given as

$$\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{i=1}^N L(f(\mathbf{x}_i; \boldsymbol{\theta}), y_i). \quad (2.5)$$

There are a multitude of ways to find the optimal parametrization of a given model, with the most popular being gradient descent [46], which is first-order iterative method. Its update rule is given by

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - \eta \nabla_{\boldsymbol{\theta}} \frac{1}{N} \sum_{i=1}^N L(f(\mathbf{x}_i; \boldsymbol{\theta}), y_i). \quad (2.6)$$

Unfortunately, for large datasets this approach breaks down, since the memory requirements can rise drastically. Because of that, another modification, called stochastic gradient descent (SGD) [47], [48] is used. In SGD, instead of the whole dataset, we randomly sample so-called batches of size $M \leq N$ and hence $\{(\mathbf{x}_i, y_i)\}_{i=1}^M \subseteq \{(\mathbf{x}_i, y_i)\}_{i=1}^N$. This leads to the update rule of the form

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - \eta \nabla_{\boldsymbol{\theta}} \frac{1}{M} \sum_{i=1}^M L(f(\mathbf{x}_i; \boldsymbol{\theta}), y_i), \quad (2.7)$$

which symbolically is identical to the one in gradient descent, but leads to approximate gradients instead of the true gradients. Fortunately, this property is useful, since it leads to models that generalize better.

2.1.1 Overfitting, Underfitting, and Regularization

An important topic in machine learning is that of generalization. Generalization capabilities directly correlate with model performance on the unseen data. As a simple toy model, consider the dataset generated by the following equation

$$y = x^2 + \varepsilon, \quad (2.8)$$

where ε is a Gaussian noise term. Fig. 2.1 shows results of fitting different polynomials to the observed dataset. On the left, one can see the fit of the first order polynomial (i.e., linear model) of the form

$$y = ax + b. \quad (2.9)$$

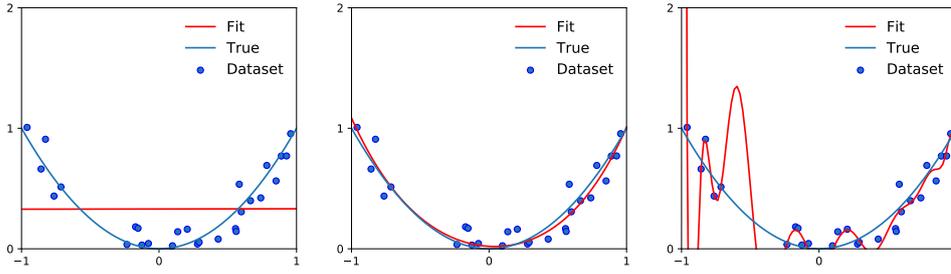


Figure 2.1: Effects of underfitting and overfitting. Different polynomial models fitted to data generated by $x^2 + \varepsilon$ process. (left) Underfitting by the linear model. (middle) Optimal model description of the data. (right) Overfitting phenomenon by the high-degree polynomial model.

As we can see, this model is incapable of fitting the observed data. In machine learning, this effect is called underfitting and it means that the model capacity (i.e. capability of the model to explain the data) is too low and thus out-of-sample performance will be bad. In the middle of Fig. 2.1 we can see a fit of a quadratic polynomial of the form

$$y = ax^2 + bx + c. \quad (2.10)$$

We can see that the obtained fit nicely aligns with the true function value. This means that the chosen model has an optimal capacity and in this case we can expect good performance on unseen samples. Lastly, on the Fig. 2.1 to the right we can see fit to the data using a polynomial of 18th degree, i.e., our model is given by

$$y = \sum_{i=0}^{18} a_i x^i. \quad (2.11)$$

We can see that this model tries to fit all points, which leads to overfitting. This means that the model tries to memorize every data point. Generalization of this model to the out-of-sample data will be bad. Of course, in the real world, it is not so straightforward to select a model with optimal capacity. One of the easiest ways to overcome overfitting is by adding the loss penalty term, for example, L^2 norm, to the model weights. In this case, the total loss will be

$$L = \frac{1}{N} \sum_{i=1}^N L_{\text{model}}(f(\mathbf{x}_i; \boldsymbol{\theta}), y_i) + \alpha \|\boldsymbol{\theta}\|_2^2, \quad (2.12)$$

and will lead to reduced overfitting due to the penalty on weights, which will constrain their value.

How does one spot the overfitting? For training purposes, it is beneficial to split the dataset into three parts - training, validation, and testing datasets. As its name suggests, one should optimize the model weights on the training dataset. Validation dataset is useful to check the model performance during the training. In case of overfitting, model performance on the training dataset will be much better than on the validation dataset. Finally, after the training finishes, one should obtain the final model performance using the testing

dataset that was unseen by the model in any way (even indirectly as the validation dataset).

2.2 Evaluation Metrics

An important part of machine learning model development is that of model performance evaluation. In classification settings, the basic metrics are recall (efficiency)

$$\text{TPR} = \frac{\text{TP}}{\text{P}}. \quad (2.13)$$

Here TP is a number of correctly identified positive samples and P is a total number of positive samples. Thus, this metric shows probability that the retrieved sample is positive.

Next metric, called false-positive rate (misidentification probability) is given by

$$\text{FPR} = \frac{\text{FP}}{\text{N}}. \quad (2.14)$$

The FP is a number of negative samples that were identified as a positive sample and N is a total number of negative samples. False-positive rate shows the probability that a positive sample is in reality negative.

In physics false-positive rates are often very low, so the alternative metric, called rejection is used, which is given by

$$\text{REJ} = \frac{1}{\text{FPR}}. \quad (2.15)$$

This metric tells us how many negative samples are rejected per one false-positive sample accepted.

The last metric, called precision (purity), which is defined as

$$\text{PREC} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (2.16)$$

shows how much of false-positive contamination there is in the retrieved sample.

2.3 Neural Networks

In this subsection, we will take a look at one specific type of parametric models, called neural networks, which today serve as a basis for many advanced applications. Fundamental building block of a neural network is a layer, which is represented by a linear operation (affine transformation or convolution [49], [50]) which is then followed by the activation function. The neural network then is a composition of N layers, i.e.

$$F(\mathbf{x}; \mathbf{W}, \mathbf{b}) = f_n(\mathbf{h}_{n-1}; \mathbf{W}_n, \mathbf{b}_n) \circ \cdots \circ f_1(\mathbf{x}; \mathbf{W}_1, \mathbf{b}_1), \quad (2.17)$$

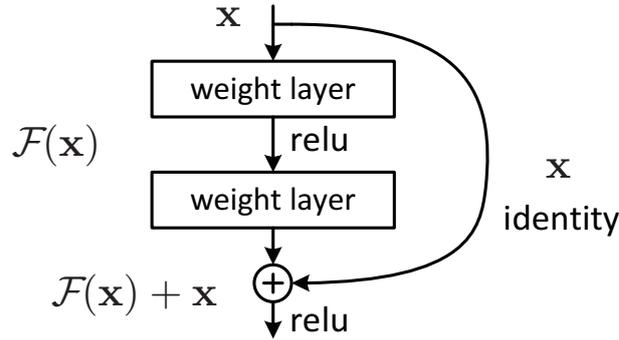


Figure 2.2: Graphical representation of the residual block. Taken from [53].

Here f_i is a layer, \mathbf{W}_i is layer weight and \mathbf{b}_i is layer bias. \mathbf{x} denotes input to the network and \mathbf{h}_i are hidden representations that are obtained after each layer. Let us take a look at how each of the network layers looks explicitly. For an affine transformation network (more generally known as dense network), the layer is given by

$$\mathbf{y} = f(\mathbf{W}\mathbf{x} + \mathbf{b}). \quad (2.18)$$

For the convolution layer, the general equation will look like

$$\mathbf{y} = f(\mathbf{W} * \mathbf{X}) = f\left(\sum_m \sum_n \mathbf{X}(i+m, j+n) \mathbf{W}(m, n)\right). \quad (2.19)$$

The function f can take many forms, but nowadays the most common choice is a ReLU(x) function [51]

$$\text{ReLU}(x) = \max(x, 0), \quad (2.20)$$

but of course there are other activation functions available [52].

In the last years, neural networks became a de facto standard backbone for many machine learning tasks, thanks to the advancement in GPU computing as well as better theoretical understanding of them. One of those is the invention of residual blocks and ResNet [53] architectures that play an important role in the construction of the JetVLAD model. The general idea of residual block is that instead of learning a direct mapping, one can try to learn the output mapping in the form

$$\mathbf{y} = \mathcal{F}(\mathbf{x}, \mathbf{W}, \mathbf{b}) + \mathbf{x}, \quad (2.21)$$

i.e., one tries to learn deviation from the identity, which leads to much simpler model training. The graphical representation of a residual block can be seen in Fig. 2.2.

2.4 Computer Vision, Descriptors and NetVLAD Layer

Computer vision is a field of computer science and engineering that deals with algorithms whose goal is the analysis and interpretation of visual data. Such algorithms have very wide applications, for example, in autonomous driving, place recognition, 3D object reconstruction. There is an intriguing correspondence between particle physics and computer vision. Indeed, most of the particle physics detectors can be thought of as a large, ultra fast camera that records particle tracks. Thus, it is an interesting topic to investigate to what extent computer vision algorithms can be applied to physical data.

The JetVLAD model, which is the basis of this thesis, is one such example. The full details of this algorithm will be explored in the later chapter, but at this stage it is important to investigate the fundamental building block of the JetVLAD algorithm, the NetVLAD layer [54]. Originally created for the task of weakly supervised place recognition, the NetVLAD is an intelligent aggregation layer that allows to gather a set of descriptors obtained from the convnet backbone and produce a fixed-size vector, that characterizes this set. In the computer vision context, a descriptor is a feature vector that encodes presence of object in a certain region.

Given a set of n descriptors, each being a d -dimensional vector, the NetVLAD layer assumes presence of k clusters in the input space, where each cluster is parametrized by vectors \mathbf{w}_k , \mathbf{c}_k and scalar b_k , that are learned from the data. The NetVLAD layer then produces a $d \times k$ matrix \mathbf{V} , which is given by

$$\mathbf{V}_{j,k} = \sum_{i=1}^n \frac{e^{\mathbf{w}_k^T \mathbf{x}_i + b_k}}{\sum_{k'} e^{\mathbf{w}_{k'}^T \mathbf{x}_i + b_{k'}}} (\mathbf{x}_{i,j} - \mathbf{c}_{k,j}). \quad (2.22)$$

Here $\mathbf{x}_{i,j}$ is a j -th element of the i -th particle descriptor and $\mathbf{c}_{k,j}$ is the j -th element of the k -th cluster center vector. This matrix is then L^2 normalized column-wise, transformed into a vector and then again L^2 normalized to obtain the final set feature vector. Thus, NetVLAD is an effective way to work with an unordered set of inputs, which makes it an ideal candidate for the classification of heavy-flavor jets if we are able to find a good descriptor representation.

Chapter 3

Overview of the Recent Results in Machine Learning Based Heavy-Flavor Jet Tagging

The recent advances in deep learning techniques do not go unnoticed in high energy physics. Indeed, a lot of analyses, especially those requiring reconstruction of heavy flavor jets, were already utilizing machine learning techniques, such as Boosted Decision Trees [55], [56]. The possibility to obtain much better performance, and thus more efficiently utilize the available data led to a large surge of deep learning based techniques, which are going to be reviewed in this chapter. Here, we are not going to review papers by experimental collaborations, since those papers go beyond the algorithmic side, which is the main focus of this work.

One of the first papers that dealt with the task of deep learning based heavy-flavor jet tagging was [1]. In this paper, the authors explored 3 different neural architectures applied towards the identification of light flavor, gluon and heavy (c and b) jets. The first architecture is based upon feed-forward neural network, which uses the top 15 impact parameter ordered tracks and their input features to solve the problem with a variable number of tracks in input. The corresponding architecture can be seen in Fig. 3.1.

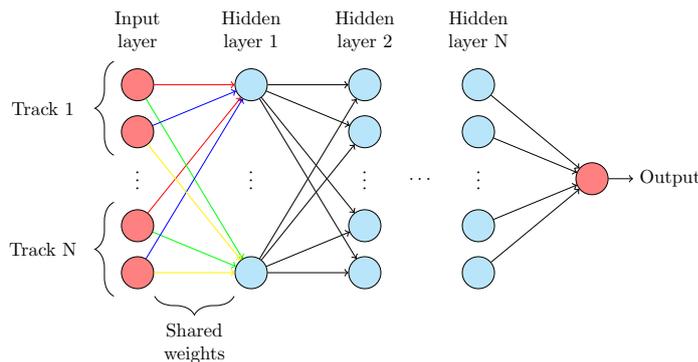


Figure 3.1: Feed forward architecture explored in [1].

The second architectures that are explored are utilizing recursive architectures, which are specially built to work with sequence based inputs. The advantage of this model is that it is able to deal with variable-length sequences by their nature, and thus does not require careful selection of the input feature

size, as in the previous model. On the other hand, this model still requires an ordering, which was again selected to be according to the impact parameter. The graphical representation of those architectures can be seen on Fig. 3.2.

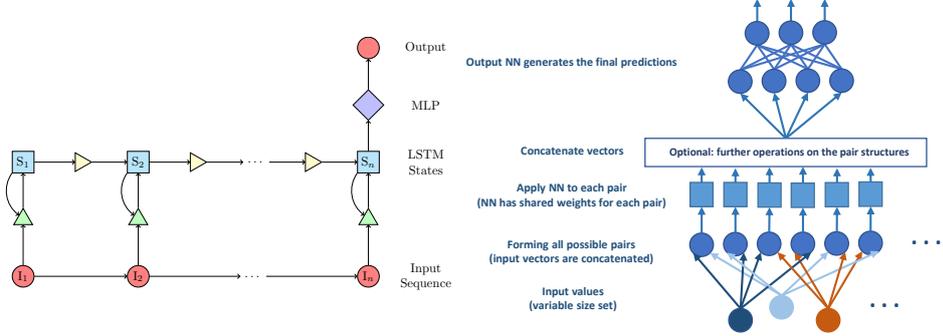


Figure 3.2: Recursive architectures explored in [1]. (left) LSTM-based architecture. (right) Outer recursive architecture.

Later on, the authors explored the effects of input feature selection, where they considered 3 groups - low level tracking information, intermediate level information taken from the vertexing algorithm, and at last, hand-crafted high level features. It is shown that generally the mixture of all input features achieves the best performance while just tracking information is providing the lowest performance. The rejection vs efficiency graphs for different flavors and different input feature combinations can be seen in Fig. 3.3. The disadvantage of this publication is the absence of efficiency vs. purity plots, which is a crucial information needed for a complete understanding of the model performance.

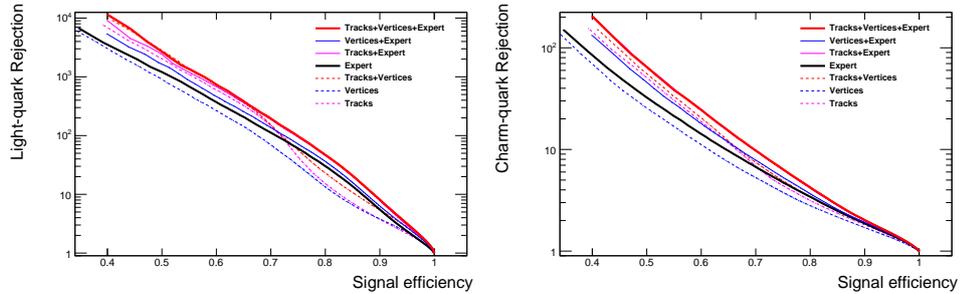


Figure 3.3: Rejection vs. efficiency graphs for different jet flavors. (left) Light quark rejection vs bottom quark efficiency graph with respect to different input feature combinations. (right) Charm quark rejection vs bottom quark efficiency graph with respect to different input feature combinations. Taken from [1].

The most recent paper related to heavy-flavor jet tagging moves the idea of tagging based on the uncurated information even further. In this paper, the authors introduce the DeepJet architecture [2], whose primary goal is to mitigate the need for manual feature selection by considering a large set of input features. The architecture takes as input approximately 650

features per jet - event level features, features related to charged and neutral particles, and secondary vertex information. The architecture itself combines one dimensional convolution layers with recurrent layers, which are then fed into the dense network. While the authors claim that this model should not discriminate against ordering, the model is not fully permutationally invariant, since the presence of RNN cells assumes that the sequence is ordered. The graphical representation of the model can be seen on Fig. 3.4.

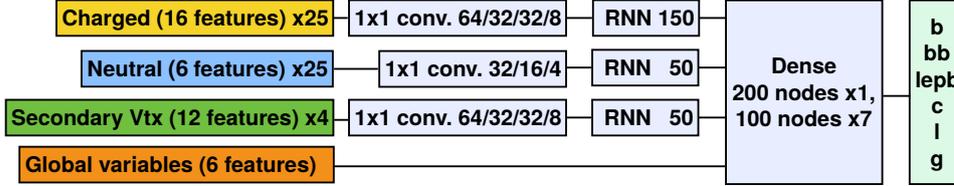


Figure 3.4: DeepJet architecture schematics. Taken from [2].

The resulting model performance can be seen in Fig. 3.5. Here, the performance is compared for different physics classification tasks (b vs. c , b vs. light flavor quarks and gluons) as well as different model architectures (DeepJet vs. DeepCSV). DeepCSV is an old, experiment-specific deep architecture, developed by the CMS collaboration [57]. As in the previous case, the authors do not show efficiency vs purity plots, which makes the full performance evaluation of the classifier hard to achieve.

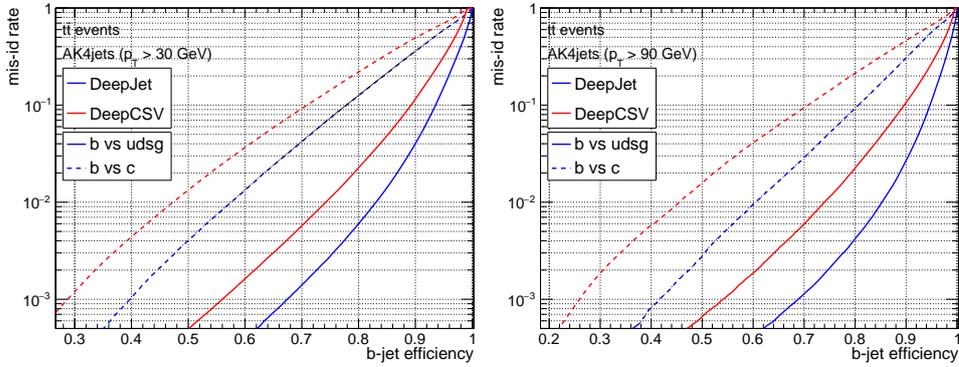


Figure 3.5: Performance of the DeepJet tagger for different classes (b vs c , b vs $udsg$) as well as for different model architectures (DeepJet vs. DeepCSV). (left) Model performance for jets with $p_T > 30$ GeV/ c . (right) Model performance for jets with $p_T > 90$ GeV/ c . Taken from [2].

Unfortunately, most of the developed architectures are experiment-specific and depend on the exact detector configuration and the information provided by it. Moreover, to the best of our knowledge, none of the models are truly permutation invariant to the input. We believe that having a general approach towards heavy-flavor jet tagging that will be independent of detector specifics and permutation invariant is beneficial and timely, considering the rising interest in the measurements of heavy-flavor jet emission diagrams, searches for new physics etc. To solve those issues, in the next chapter we will introduce the JetVLAD model, which tries to overcome all those issues at the same time.

Chapter 4

The JetVLAD Model

The focus of this chapter is an introduction to the JetVLAD model, which is the main contribution of this thesis and also of the paper "Identifying Heavy-Flavor Jets Using Vectors of Locally Aggregated Descriptors", published in JINST, of which the author of this thesis is a principal author. First, the model architecture is discussed with the main motivations that were driving its development. Then the data modeling process is described. The last part of this chapter deals with the experimental evaluation and study of the JetVLAD model in application to heavy-flavor jet identification in p+p collisions at RHIC energy of $\sqrt{s} = 200$ GeV.

4.1 Formal Statement

In this section, we will formalize the problem that we wish to solve. Each p+p collision, which we refer to as an event, is assumed to be a set of the form

$$\mathcal{E} = \{\mathbf{r}_i\}_{i=1}^N, \quad (4.1)$$

where N is the number of particles in the event and \mathbf{r}_i is a **particle state vector**. The particle state vector is constructed using all possible information related to the particle i , i.e.,

$$\mathbf{r}_i = (p_{T,i}, \eta_i, \varphi_i, d_{xy,i}, d_{z,i}, \dots), \quad (4.2)$$

where $p_{T,i}$ is a transverse momentum of particle i , η_i is a pseudorapidity of particle i , φ_i is a radial angle of particle i and $d_{xy,i}, d_{z,i}$ are distances of closest approach to primary vertex in $x - y$ and z plane for particle i . In this context, each jet is a subset of the event \mathcal{E} , that was identified by the jet clustering algorithm

$$\mathcal{J} = \{\mathbf{r}_k\}_{k=1}^{N_{const}}, \quad (4.3)$$

where N_{const} is the number of particle constituents in a jet. We can treat the jet as an unordered subset if we assume that it was clustered using the anti- k_T algorithm, since the hard core clustering strategy leads to unphysical ordering. After this, we wish to construct a model $f(\cdot)$ such that

$$f(\mathcal{J}) \rightarrow \{bc, udsq\}, \quad (4.4)$$

i.e., given a jet \mathcal{J} we ask what is the flavor of that particular jet. As a next step, we need to construct an appropriate particle descriptors. However, in physical measurements, compared to computer vision, all measurements are already high level and represent the highest possible information content available. Hence, we can consider particle state vectors as descriptors, which we call a **particle descriptors**. With that in mind, we can continue with constructing the JetVLAD architecture.

4.2 JetVLAD Architecture

The JetVLAD architecture was built specifically for the task of heavy-flavor jet identification, with the following goals in mind:

1. The model should utilize a set of particles as an input.
2. Simple architecture to minimize possibility of overfitting.
3. Minimal use of high-level input features.
4. Largely independent of low-level Monte-Carlo generator implementation details.

The NetVLAD layer introduced in Chapter 2 was chosen as a basis for this model, since it allows to solve the first goal. To solve the second requirement, we consider a simple multilayer perceptron architecture with residual skip connections that mimic those in the ResNet [53] architecture, which was experimentally shown to significantly simplify the optimization procedure when training the JetVLAD model. Despite being simple, this architecture can potentially still overfit the training data. To reduce such possibility, we also add the DropOut layer [58] with a probability of $p = 0.5$. The last two goals can be solved by a careful construction of the training dataset, that is as close to the real data as possible, a process which is described in the following section.

In total, the model formula for the JetVLAD model can be written as

$$\text{JetVLAD} = \text{NetVLAD}(N_c) \rightarrow D \times [\text{ResidualBlock}] \rightarrow \text{Softmax}. \quad (4.5)$$

The parameters N_c and D represent the model hyperparameters, whose optimal values are selected further on. The JetVLAD model is trained using stochastic gradient descent with cosine annealing and warm restart [59], which varies the optimization intensity using cosine function, which was shown to produce a better and more robust models. This combination performed the best based upon the experiments, with the optimal learning rate value $\eta = 0.013$, and cosine annealing parameters being $T_0 = 1$ (initial amplitude length in epoch) and $T_{\text{mult}} = 3$ (amplitude extension length).

Technical details. The model is trained on one Tesla V100 GPU for approximately 2 hours. For each dataset we train 3 models whose predictions are then averaged into one, to estimate the effects of random weight initialization on model performance.

p_T^{jet} [GeV/c]	\hat{p}_T^{min} [GeV/c]	\hat{p}_T^{max} [GeV/c]	R [-]
5-10	3	12	0.4
10-15	8	17	0.4
15-20	13	22	0.4
20-25	18	27	0.4
25-40	23	42	0.4

Table 4.1: Kinematic ranges used for dataset generation within Pythia8 model.

4.3 Dataset

In this work, the simulated Monte-Carlo data were used for the purposes of model development. The simulations were made with the `Pythia8.235` [60] Monte-Carlo model that was adjusted to the particle level data by means of a fast simulator approach. This allows one to create datasets that resemble experimentally measured data without the need to do a full detector simulation. Of course, such simulation is not the best description of the experimental data, but for the purposes of tagging model development, such an approach is sufficient.

Model configuration. The model was run in $p + p$ collisions at $\sqrt{s} = 200$ GeV mode to generate two dataset samples - one, labeled as `HardQCD`, which is supposed to describe the real world distribution of jet flavors and `Balanced`, which is specially constructed such that the ratio of $udsg : c : b$ jets is 50%:25%:25%. This dataset serves as an idealized benchmark case for the tagger performance. Tab. 4.1 shows the kinematic configurations that were used in the simulations. The particle decays are set to be limited to a cylinder with a diameter of 600 mm and a length of 4000 mm, which mimics the typical dimensions of a time projection chamber used by the STAR experiment at RHIC for detection of charged particles created in the collision. In the simulations, we consider only charged particles with $\tilde{p}_T \in [0.2, 30]$ GeV/c, $|\tilde{d}_z| < 60$ mm and $|\tilde{d}_{xy}| < 20$ mm, where \tilde{p}_T denotes the particle momentum that was smeared by means of the fast simulation and $|\tilde{d}_{xy}|/|\tilde{d}_z|$ denote the particle $x - y$ and z vertex distances that were also smeared by the fast simulation approach, which is described below. The primary vertex of all collisions is fixed to $(0, 0, 0)$ to simplify the calculations, although modifications to the random primary vertex location can be easily incorporated within such framework.

Fast simulator. The fast simulator approach is based on the multiple observable smearing by means of Gaussian distribution sampling which aims to reproduce a typical collider experiment, such as STAR, which has a time projection chamber with full azimuthal coverage and pseudorapidity acceptance of $|\eta| < 1$ embedded within a magnetic field that allows to measure track momenta via their curvature and a tracker. The second important part is a precise silicon tracker placed close to the interaction vertex, which

allows precise identification of particle secondary vertex, that is of the most importance for the heavy flavor analyses. In the STAR experiment, for this purpose the Heavy Flavor Tracker (HFT) has been used [61], [62]. Although not ideal, this fast simulator approach can serve as a good approximation for the particle level to detector level smearing process.

The first part of the fast simulation process is dealing with simulating the time projection chamber and magnetic field effects on the created particle, since it is the combination of a finite resolution of the TPC detector with magnetic field that smears the information about the particle in question. The first effect that one needs to capture is that of reduced tracking efficiency. This can be easily captured by sampling the rejection factor $t \sim \mathcal{U}([0, 1])$. If $t > t_{thresh}$ where t_{thresh} is some threshold, then the particle is rejected, generally with $1 - t_{thresh}$ probability. In this work, $t_{thresh} = 0.8$ was chosen, which corresponds to 80% tracking efficiency. The second effect is that of a change in a momentum resolution. It was shown in [63] that this effect can be simply captured by the Gaussian process of the form

$$\tilde{p}_T \sim \mathcal{N}(p_T, \sigma(p_T)), \quad (4.6)$$

with $\sigma(p_T)$ being the empirical smearing function obtained by the fit and p_T being the transverse momentum of the input particle. The chosen form of $\sigma(p_T)$ was

$$\sigma(p_T) = 0.003 \cdot p_T^2, \quad (4.7)$$

and corresponds to the STAR experiment at RHIC [39].

The second part of the fast simulator deals with the simulation of tracker finite resolution. For this purpose, the parametrization [28] of STAR [61], [62] tracker was used. For $x - y$ plane, the fitted resolution is given by

$$\sigma_{xy}(\mathbf{p}) = \mathbf{1}_{[0, 2.5]} \left[\frac{0.03}{|\mathbf{p} - 0.04|} + 0.006 \right] + \mathbf{1}_{[2.5, \infty)} \cdot 0.02. \quad (4.8)$$

Here \mathbf{p} is a particle momentum vector and $\mathbf{1}$ are indicator functions. Analogously, we obtain the following functional form by the fit to the z resolution:

$$\sigma_z(\mathbf{p}) = \mathbf{1}_{[0, 2.5]} \left[\frac{0.03}{|\mathbf{p} - 0.02|} + 0.008 \right] + \mathbf{1}_{[2.5, \infty)} \cdot 0.02. \quad (4.9)$$

This class of fits gives a good approximation to the detector resolution, albeit their main disadvantage is the incorrect treatment of the long tails related to the particles with very high momentum. Nonetheless, empirically those give good results and therefore they were chosen as a final form. The smearing for each particle is then conducted by sampling the smeared vertex distances from Gaussian distribution, i.e.

$$\tilde{d}_{xy} \sim \mathcal{N}(d_{xy}, \sigma_{xy}(\mathbf{p})), \quad (4.10)$$

$$\tilde{d}_z \sim \mathcal{N}(d_z, \sigma_z(\mathbf{p})). \quad (4.11)$$

Here \mathbf{p} is a momentum of the input particle, d_z and d_{xy} are the secondary vertex distances of the input to the primary vertex.

Pileup simulations. Experimental measurements are often influenced by presence of the so-called pileup effects. Pileup effects could be caused by a multiple collisions within the detector at the same time. To estimate the effect of pileup on the classification performance, we simulate the pileup effects in one of the samples. To achieve those results, we overlay `SoftQCD` event with primary vertex located at $(0, 0, z)$ with $z \sim \mathcal{U}([-60, 60])$ mm. The two events are then merged into one and processed as is described below. This approach is an effective way of simulating pileup effect at energies of $\sqrt{s} = 200$ GeV.

Thermal background simulation. We also ask question related to the effects of thermal heavy-ion like background on model performance. For this we sample $N = 100$ particles with p_T sampled from thermal distribution with $T = 230$ MeV and φ, η sampled from $[\varphi_{jet} - 0.4, \varphi_{jet} + 0.4]$ and $[\eta_{jet} - 0.4, \eta_{jet} + 0.4]$. We chose this form of noise instead of event level noise addition since it allows us to represent extreme cases, which give an upper bound on a model performance in the studied region.

Data analysis. The simulated events are then clustered by the anti- k_T algorithm with resolution parameter of $R = 0.4$. The resulting jets are then assigned a flavor according to the following procedure:

1. All outgoing hard scattered partons (denoted by $\hat{\mathcal{P}}$) in event are considered.
2. For each jet we find parton i such that

$$i = \arg \min_{j \in \hat{\mathcal{P}}} \Delta R(\eta_{jet}, \varphi_{jet}, \eta_j, \varphi_j), \quad m_i > m_j \forall j \in \hat{\mathcal{P}}, \quad (4.12)$$

i.e. we seek the closest parton in $\eta - \varphi$ space that has the highest mass.

3. The jet is then assigned the flavor of parton i .

The resulting jets are deconstructed into separate particle constituents which are later on used as an input to the tagging algorithm. The Tab. 4.2 summarizes the input variables that are used as an input features by the model.

Statistics. In this work, we generated 2 datasets, each with kinematic configurations given in Tab. 4.1. Since our center of mass energy is $\sqrt{s} = 200$ GeV, we merge c and b jets into one class, which we call HF-jets. The first dataset is using `Pythia8 HardQCD` process to simulate a realistic sample of charged $udsg$, c and b jets. The second one is constructed such that the ratio of light to heavy-flavor charged jets is 50% : 50%. Examples of input feature distributions for each separate class can be seen on Fig. 4.1.

4.4 Hyperparameter Optimization

Input feature selection. As a first step, we find the optimal hyperparameters for our model. Recall that `JetVLAD` has 2 hyperparameters N_c , number of

Variable	Definition
p_T	$p_T = \sqrt{p_x^2 + p_y^2}$
η	Pseudorapidity
φ	Azimuthal angle
z	Shared momentum fraction $z = p_T/p_T^{jet}$
ΔR	$\Delta R = \sqrt{(\varphi - \varphi_{jet})^2 + (\eta - \eta_{jet})^2}$
m	$(p_T/p_T^{jet}) \cdot (\sqrt{(\varphi - \varphi_{jet})^2 + (\eta - \eta_{jet})^2})$
d_{xy}	$d_{xy} = \sqrt{(d_x^p - v_x)^2 + (d_y^p - v_y)^2}$
d_z	$d_z = d_z^p - v_z$

Table 4.2: Model input variables per track. Here p_i denotes momentum vector components of a particle, d_i^p are secondary vertex coordinates of a particle and v_i are primary vertex coordinates.

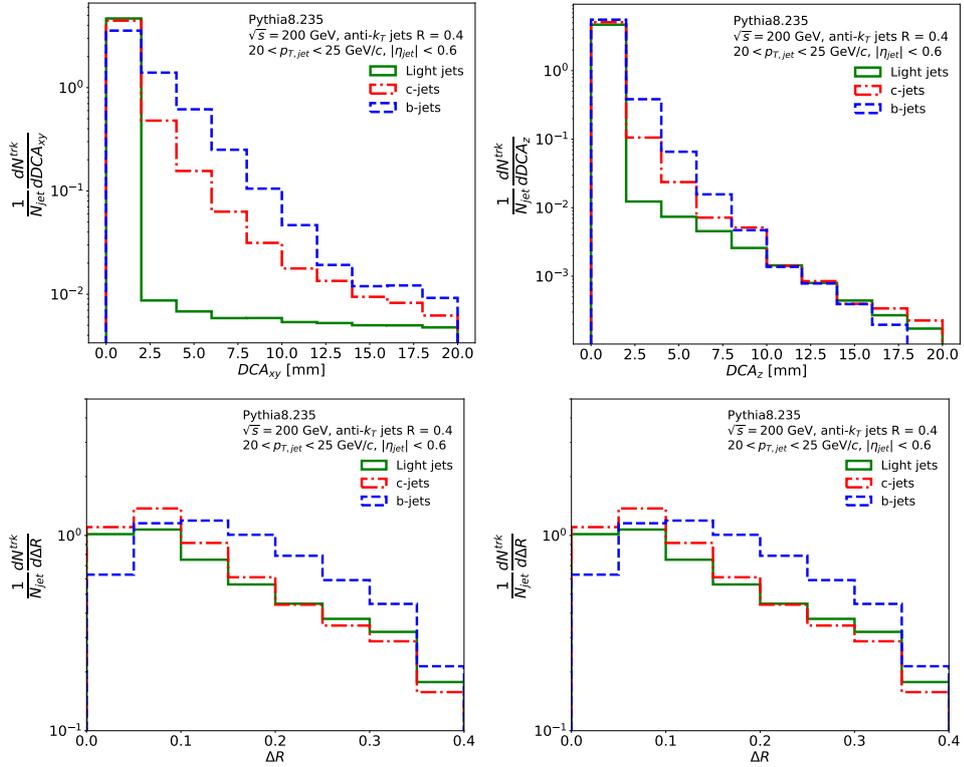


Figure 4.1: Example of some feature distributions. $udsg$ (light) jets are represented by the green color, c (charm) jets are represented by the red color and b (beauty) jets are represented by the blue color.

clusters in the NetVLAD layer and D , the number of residual blocks (model depth). For input features, introduced in Tab. 4.2 we also have to select their optimal combination. To find it, we train the model on multiple jet p_T bin datasets, to assess model performance with each input feature combination across different jet p_T ranges. The resulting purity vs. efficiency and rejection vs. efficiency plots across different jet p_T bins in **HardQCD** dataset can be seen on Fig. 4.2 and for **Balanced** dataset can be seen on Fig. 4.3. From

this, we can see that the optimal combination is made of tracking (p_T , η , φ) and vertexing (d_{xy} , d_z) features. Interestingly, just vertexing (d_{xy} , d_z) information alone is a good enough descriptor of jet flavor. Nonetheless, tracking (p_T , η , φ) information is straightforward to add and the benefit of improved performance is worth of a slight increase in an input feature complexity. Because of that, we select the combination of tracking (p_T , η , φ) and vertexing (d_{xy} , d_z) as a final combination which will be used in studies further on. As expected, tracking (p_T , η , φ) information alone is insufficient for high purity measurements. Combination of tracking (p_T , η , φ) and fragmentation (z , ΔR , m) information is slightly increasing the purity, but it is still less than 10% and in real life scenarios we can expect it to be even smaller. Interestingly, for the **Balanced** dataset this combination provides a decent performance. Hence, if one is able to construct a trigger that will yield an approximately balanced jet flavor dataset, it is possible to identify such jets without the need for expensive tracking hardware.

Optimal hyperparameter selection. The next step consists of an optimal hyperparameter selection. The best performance is achieved via random grid search, which consists of a random sampling step followed by the model training. The resulting graphs for integrated efficiency and purity vs hyperparameter value can be seen in Fig. 4.4. From this, we can see that $N_c = 33$ and $D = 4$ (resulting in $D = 2$ for the residual block) are the optimal hyperparameter values.

4.5 Model Performance with Full Dataset and Background Effects

In this section, we will evaluate the JetVLAD model performance on the full **HardQCD** dataset introduced in Sec. 4.3. We omit **Balanced** dataset since it does not represent a realistic cross-section distribution that one can expect in the real data. Fig. 4.5 shows main results for the tracking+vertexing version of the JetVLAD model across different jet p_T ranges. The numerical results for different operating points are summarized in Tab. 4.3. We can see that the model achieves excellent performance down to the lowest jet p_T bin. On average, the tagger achieves universally $\sim 80\%$ purity, $\sim 80\%$ efficiency and rejection factor of ~ 200 across whole jet p_T range.

As a next step, we consider three important aspects of p+p and A+A collision - changes in tracking efficiency, the presence of pileup, and the presence of a thermal background and their influence on the model performance.

Tracking efficiency. First, we explore how changes in tracking efficiency affect model performance. This is done by varying the tracking efficiency resolution during the data generation process. To assess the full effect in an unbiased way, we do not retrain the model to account for new tracking efficiencies. The results are displayed in Fig. 4.6. As can be seen, the model performance is not affected by the changes in the tracking efficiency.

Jet p_T [GeV/ c]	Efficiency	Purity	Rejection
5 - 10	80%	83%	223
	50%	88%	540
10 - 15	80%	85%	223
	50%	88%	476
15 - 20	80%	85%	259
	50%	88%	506
20 - 25	80%	85%	310
	50%	88%	624
25 - 40	80%	81%	322
	50%	85%	677

Table 4.3: Purity and rejection values for different jet p_T bins at fixed efficiency operating points of 50% and 80%.

Pileup. Next, we explore the effects of the pileup on the model performance on the dataset that was described in Sec. 4.3. The resulting performance curves are shown in Fig. 4.7. As expected, the model performance has reduced, but not significantly, which shows the strength of the NetVLAD layer and its resistance to noise.

Thermal background. As a last step, we explore the effects of thermal background on the model performance. We model this process with $N = 100$ particles **per jet**, which represents an extreme case that is unlikely to be present at RHIC energies. The resulting model performance is shown in Fig. 4.8. As we can see, adding the effects of extreme thermal background leads to a significant decrease in model performance. Yet, retraining the model from scratch on a new data restores the good model performance. This could be explained by the fact that the general pattern of thermal particles is easy to separate in this model. Indeed, as shown in Fig. 4.2 the model gets most of its discriminative power from the secondary vertex position. Most of the thermal particles are produced near the primary vertex and thus they can be easily separated just based on this variable alone.

Effects of model performance on measurements. To practically illustrate, how different tagger performance affects measurements, assume that we are interested in measuring an unspecified observable and we have a dataset of 500000 $udsg$ -jets and 20000 HF-jets, which roughly corresponds to the realistic cross-section ratio expected from the theory. The quality of the reconstructed sample is measured via significance value, which is given by

$$\sigma = \frac{S}{\sqrt{S+B}}, \quad (4.13)$$

where S is a number of signal jets and B is a number of background jets in sample. Let us consider two taggers, one of which has efficiency of 80% and purity of 80%, corresponding to the JetVLAD model and a less performant tagger, that has for simplicity 30% efficiency and 30% purity.

The signal is obtained via

$$S = \varepsilon_{HF} N_{HF}, \quad (4.14)$$

where ε_{HF} is a tagger efficiency and N_{HF} is a number of HF jets in the sample. The background is obtained from the definition of purity. Since the purity is given by

$$p_{HF} = \frac{S}{S + B}, \quad (4.15)$$

we can find by the means of simple algebra that

$$B = \left(\frac{1}{p_b} - 1 \right) S. \quad (4.16)$$

Putting it all together in the definition of significance yields us

$$\sigma = \frac{\varepsilon_{HF} N_{HF}}{\sqrt{\varepsilon_{HF} N_{HF} + \varepsilon_{HF} N_{HF} \left(\frac{1}{p_{HF}} - 1 \right)}}. \quad (4.17)$$

Plugging in all the values we get that the significance of 80%/80% tagger is

$$\sigma = 113.1, \quad (4.18)$$

while the 30%/30% tagger gets

$$\sigma = 42.4. \quad (4.19)$$

Thus, usage of the JetVLAD model can get ~ 3 times better statistical precision, or alternatively, require ~ 3 times less data for the same statistical precision as can be achieved with less performant tagger.

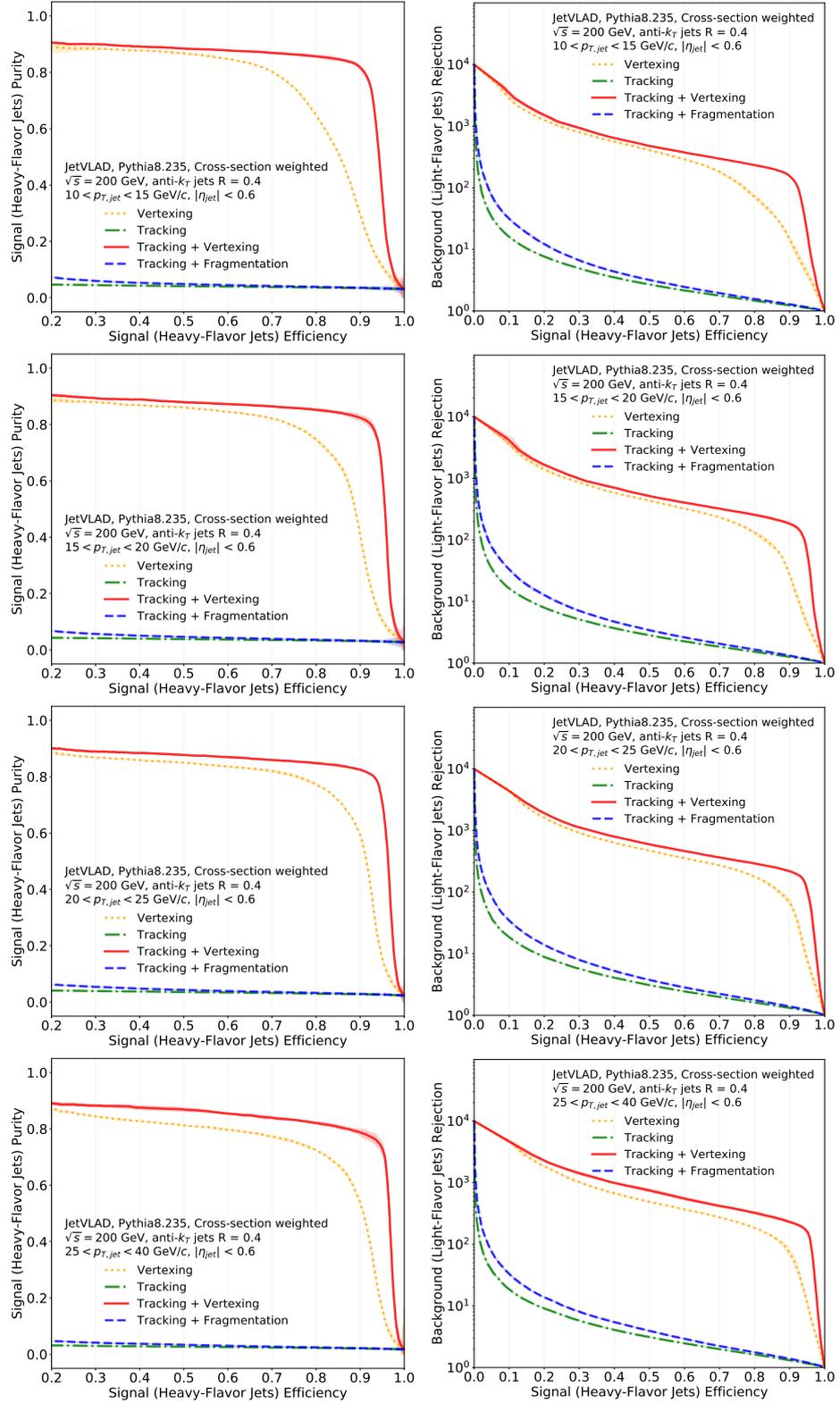


Figure 4.2: Selection of input feature combinations - HardQCD dataset. (left) Efficiency vs. purity for different input feature combinations. (right) Efficiency vs. rejection for different input feature combinations. Each row shows different jet p_T ranges that were used to train the model.

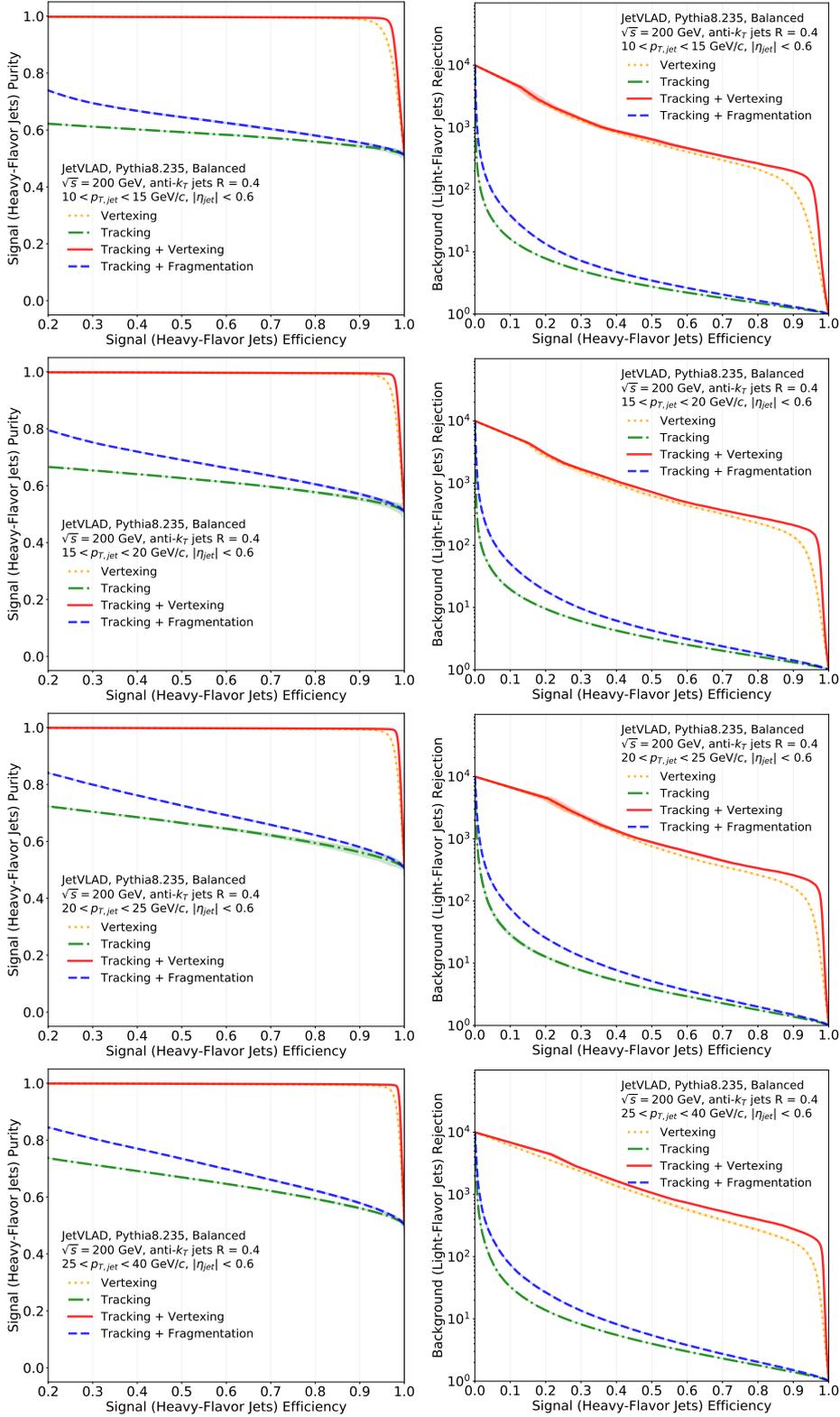


Figure 4.3: Selection of input feature combinations - Balanced dataset. (left) Efficiency vs. purity for different input feature combinations. (right) Efficiency vs. rejection for different input feature combinations. Each row shows different jet p_T ranges that were used to train the model.

4. The JetVLAD Model

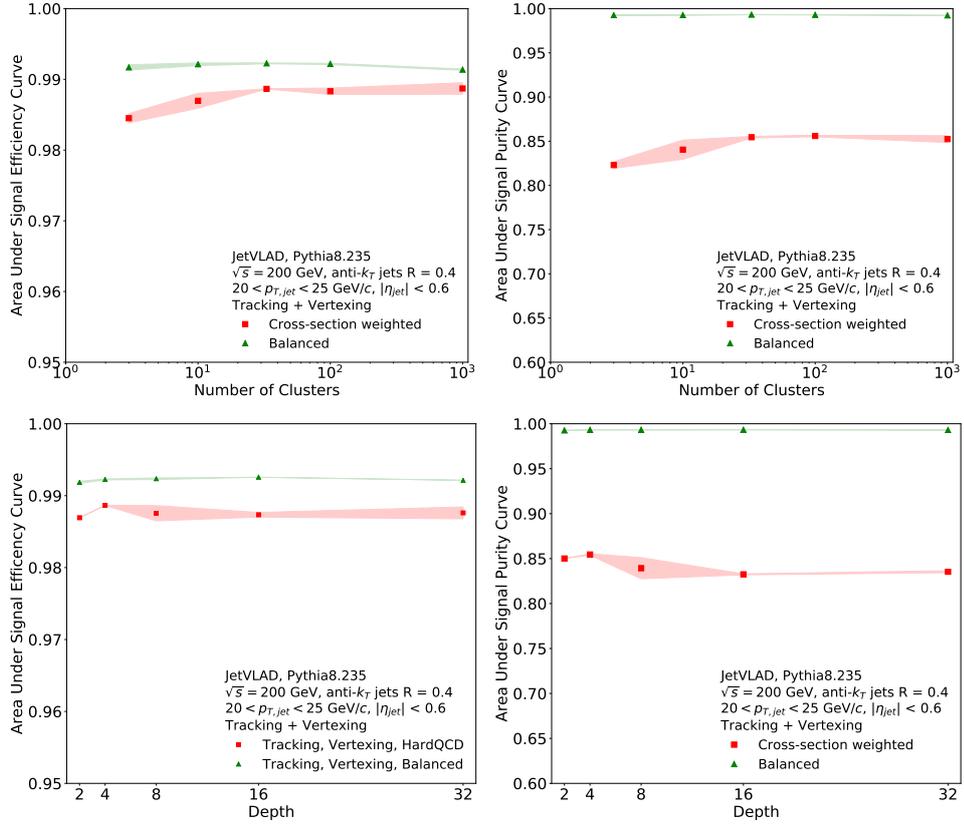


Figure 4.4: Optimal hyperparameter value selection. (left column) Integrated efficiencies vs. number of clusters (upper left) and depth (lower left). (right column) Integrated purities vs. number of clusters (upper right) and depth (lower right). The optimal parameters are given by $N_c = 33$ and $D = 4$.

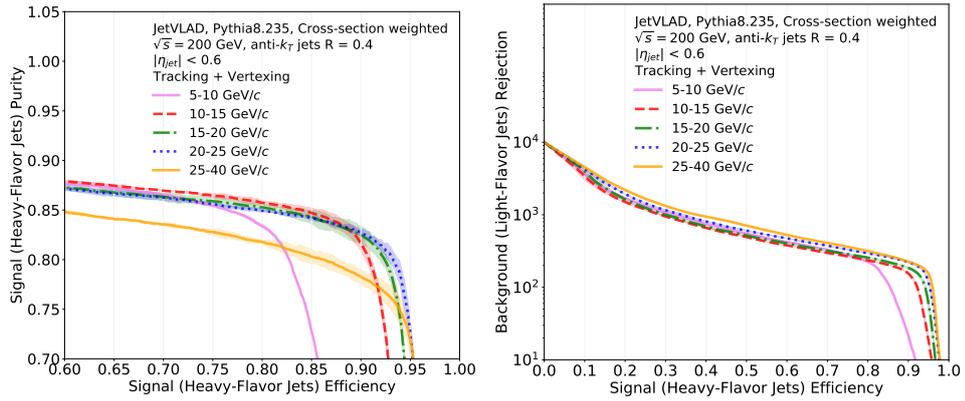


Figure 4.5: JetVLAD model performance across different jet p_T ranges. (left) Purity vs. efficiency for different jet p_T ranges. (right) Rejection vs. efficiency for different jet p_T ranges. Colors represent different p_T bins used to train model.

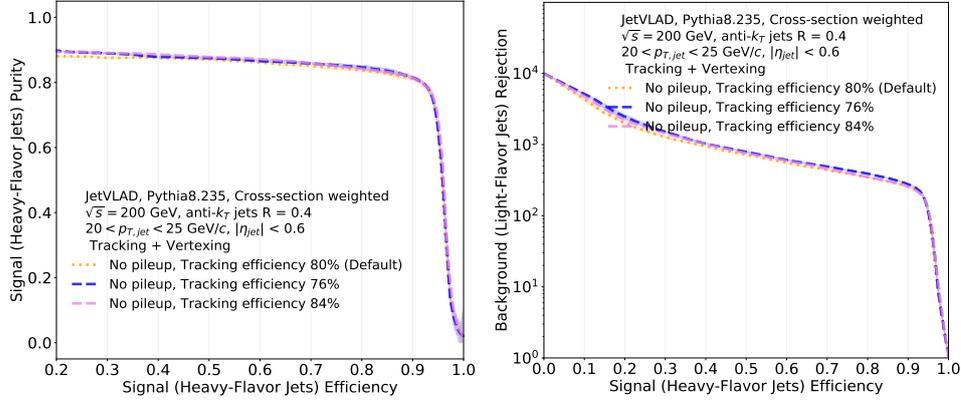


Figure 4.6: Effects of tracking efficiency on the JetVLAD model performance. (left) Purity vs. efficiency for different tracking efficiency values. (right) Rejection vs. efficiency for different tracking efficiency values. Colors represent tracking efficiency values.

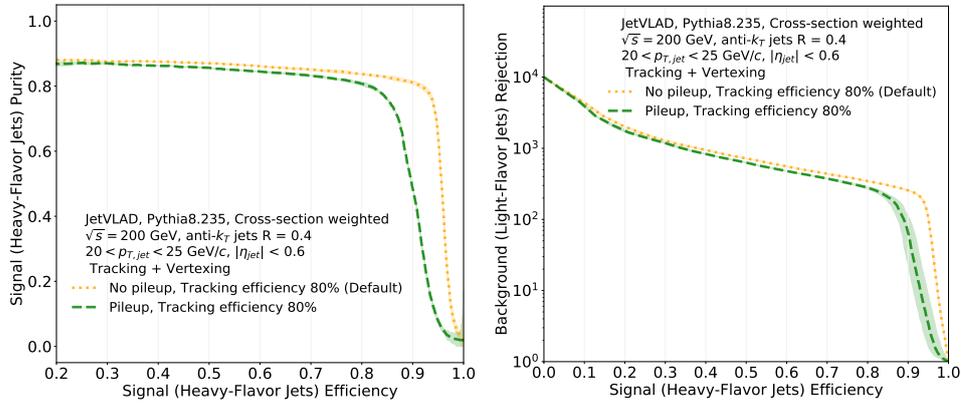


Figure 4.7: Effects of pileup on the JetVLAD model performance. (left) Purity vs. efficiency for model evaluated on training data (dotted) and model evaluated on the data with pileup (dashed). (right) Rejection vs. efficiency for model evaluated on training data (dotted) and model evaluated on the data with pileup (dashed). Colors represent tracking efficiency values.

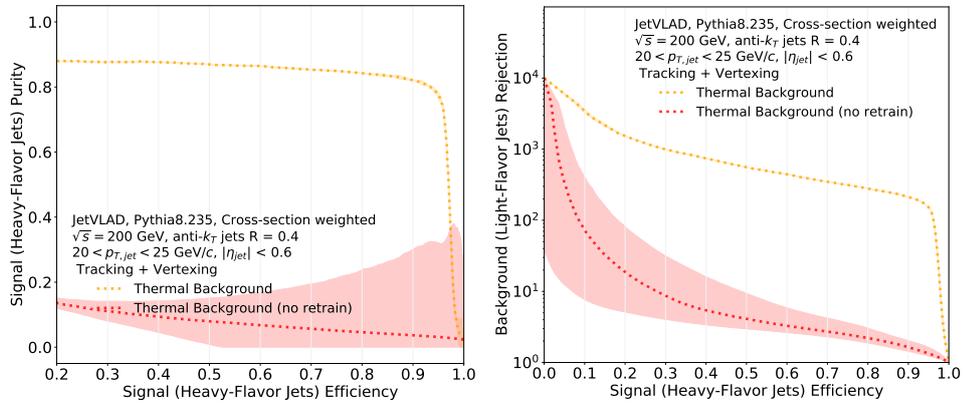


Figure 4.8: Effects of thermal background on JetVLAD model performance. (left) Purity vs. efficiency for default model evaluated on thermal background data (red) and model retrained on the thermal data (yellow). (right) Rejection vs. efficiency for model evaluated on thermal background data (red) and model retrained on the thermal data (yellow). This plot is not showing the default model performance since this would not lead to fair comparison. The $p_{T,jet}$ here denotes the p_T of a jet without thermal background.



Chapter 5

Conclusion

In this work, a novel machine learning based jet tagging architecture was introduced. In Chapter 1, the basics of jet physics were introduced, including jet substructure measurements and dead cone effect. We also reviewed some recent experimental results related to the area of heavy flavor jet substructure. In Chapter 2, the basics of machine learning and computer vision were introduced. Chapter 3 reviews the recent results on experiment independent machine learning based heavy flavor jet tagging methods. Chapter 4 introduces the main contribution of this work, the JetVLAD tagging model. The JetVLAD model achieves an excellent performance in p+p collisions at $\sqrt{s} = 200$ GeV across wide jet p_T range from 5 to 40 GeV/ c , where it generally achieves an efficiency of $\sim 80\%$, purity of $\sim 80\%$ and rejection factor of 200. Such performance opens a wide range of possibilities for flavor tagged jet substructure studies at STAR and new sPHENIX experiments at RHIC. Our experiments with enhanced background as presented in Figs. 4.7, 4.8 show that the resulting model should be resistant to pileup and background fluctuations, which makes possible application to real data much easier.

There are few technical details that one needs to take care of before applying the JetVLAD model to the real data. First, one needs to do a full detector simulation representing the detector of interest to include fine-grained detector smearing effects in the data. The second part is concerned with Monte Carlo simulations. To mitigate the possibility of indirectly induced Monte Carlo model-dependent bias in the resulting model, one needs to consider a training data made of the mixture of Monte Carlo models, which in principle should mitigate the effects coming from the underlying physics descriptions of each model. At last, one needs to consider a realistic background in the training sample, preferably taken from the real data via an embedding procedure.

Despite great performance of the JetVLAD model, the results that will be obtained in real data need to be scrutinized, due to the black-box nature of the deep neural networks. The model can be validated by comparing the JetVLAD results with the results that are obtained with the classical transparent method within multiple processes. In case of validity, on average, the results should be close to each other within statistical errors.



Bibliography

- [1] D. Guest, J. Collado, P. Baldi, S.-C. Hsu, G. Urban, and D. Whiteson, “Jet Flavor Classification in High-Energy Physics with Deep Neural Networks”, *Phys. Rev. D*, vol. 94, no. 11, p. 112002, 2016. DOI: 10.1103/PhysRevD.94.112002. arXiv: 1607.08633 [hep-ex].
- [2] E. Bols, J. Kieseler, M. Verzetti, M. Stoye, and A. Stakia, “Jet Flavour Classification Using DeepJet”, *JINST*, vol. 15, no. 12, P12012, 2020. DOI: 10.1088/1748-0221/15/12/P12012. arXiv: 2008.10519 [hep-ex].
- [3] J. Bielčíková, R. Kunnawalkam Elayavalli, G. Ponimatkin, J. H. Putschke, and J. Sivic, “Identifying Heavy-Flavor Jets Using Vectors of Locally Aggregated Descriptors”, *JINST*, vol. 16, no. 03, P03017, 2021. DOI: 10.1088/1748-0221/16/03/P03017. arXiv: 2005.01842 [hep-ph].
- [4] M. Tanabashi *et al.*, “Review of Particle Physics”, *Phys. Rev.*, vol. D98, no. 3, p. 030001, 2018. DOI: 10.1103/PhysRevD.98.030001.
- [5] J. C. Collins, D. E. Soper, and G. F. Sterman, “Factorization of Hard Processes in QCD”, *Adv. Ser. Direct. High Energy Phys.*, vol. 5, pp. 1–91, 1989. DOI: 10.1142/9789814503266_0001. arXiv: hep-ph/0409313 [hep-ph].
- [6] V. N. Gribov and L. N. Lipatov, “Deep inelastic e p scattering in perturbation theory”, *Sov. J. Nucl. Phys.*, vol. 15, pp. 438–450, 1972, [*Yad. Fiz.*15,781(1972)].
- [7] G. Altarelli and G. Parisi, “Asymptotic Freedom in Parton Language”, *Nucl. Phys.*, vol. B126, pp. 298–318, 1977. DOI: 10.1016/0550-3213(77)90384-4.
- [8] Y. L. Dokshitzer, “Calculation of the Structure Functions for Deep Inelastic Scattering and e+ e- Annihilation by Perturbation Theory in Quantum Chromodynamics.”, *Sov. Phys. JETP*, vol. 46, pp. 641–653, 1977, [*Zh. Eksp. Teor. Fiz.*73,1216(1977)].
- [9] S. Catani, Y. L. Dokshitzer, M. H. Seymour, and B. R. Webber, “Longitudinally invariant K_t clustering algorithms for hadron hadron collisions”, *Nucl. Phys.*, vol. B406, pp. 187–224, 1993. DOI: 10.1016/0550-3213(93)90166-M.

- [10] M. Cacciari, G. P. Salam, and G. Soyez, “The anti- k_t jet clustering algorithm”, *JHEP*, vol. 04, p. 063, 2008. DOI: 10.1088/1126-6708/2008/04/063. arXiv: 0802.1189 [hep-ph].
- [11] Y. L. Dokshitzer, G. D. Leder, S. Moretti, and B. R. Webber, “Better jet clustering algorithms”, *JHEP*, vol. 08, p. 001, 1997. DOI: 10.1088/1126-6708/1997/08/001. arXiv: hep-ph/9707323 [hep-ph].
- [12] M. Wobisch, “Measurement and QCD analysis of jet cross-sections in deep inelastic positron proton collisions at $\sqrt{s} = 300$ GeV”, Ph.D. dissertation, Aachen, Tech. Hochsch., 2000. [Online]. Available: <http://www-library.desy.de/cgi-bin/showprep.pl?desy-thesis00-049>.
- [13] M. Cacciari, G. P. Salam, and G. Soyez, “FastJet User Manual”, *Eur. Phys. J. C*, vol. 72, p. 1896, 2012. DOI: 10.1140/epjc/s10052-012-1896-2. arXiv: 1111.6097 [hep-ph].
- [14] M. Cacciari and G. P. Salam, “Dispelling the N^3 myth for the k_t jet-finder”, *Phys. Lett. B*, vol. 641, pp. 57–61, 2006. DOI: 10.1016/j.physletb.2006.08.037. arXiv: hep-ph/0512210.
- [15] S. Catani, Y. L. Dokshitzer, M. H. Seymour, and B. R. Webber, “Longitudinally invariant K_t clustering algorithms for hadron hadron collisions”, *Nucl. Phys.*, vol. B406, pp. 187–224, 1993. DOI: 10.1016/0550-3213(93)90166-M.
- [16] M. Cacciari, G. P. Salam, and G. Soyez, “The anti- k_t jet clustering algorithm”, *JHEP*, vol. 04, p. 063, 2008. DOI: 10.1088/1126-6708/2008/04/063. arXiv: 0802.1189 [hep-ph].
- [17] Y. L. Dokshitzer, G. D. Leder, S. Moretti, and B. R. Webber, “Better jet clustering algorithms”, *JHEP*, vol. 08, p. 001, 1997. DOI: 10.1088/1126-6708/1997/08/001. arXiv: hep-ph/9707323 [hep-ph].
- [18] M. Wobisch, “Measurement and QCD analysis of jet cross-sections in deep inelastic positron proton collisions at $\sqrt{s} = 300$ GeV”, Ph.D. dissertation, Aachen, Tech. Hochsch., 2000. [Online]. Available: <http://www-library.desy.de/cgi-bin/showprep.pl?desy-thesis00-049>.
- [19] F. A. Dreyer, G. P. Salam, and G. Soyez, “The Lund Jet Plane”, *JHEP*, vol. 12, p. 064, 2018. DOI: 10.1007/JHEP12(2018)064. arXiv: 1807.04758 [hep-ph].
- [20] H. A. Andrews *et al.*, “Novel tools and observables for jet physics in heavy-ion collisions”, *J. Phys. G*, vol. 47, no. 6, p. 065 102, 2020. DOI: 10.1088/1361-6471/ab7cbc. arXiv: 1808.03689 [hep-ph].
- [21] A. J. Larkoski, S. Marzani, G. Soyez, and J. Thaler, “Soft Drop”, *JHEP*, vol. 05, p. 146, 2014. DOI: 10.1007/JHEP05(2014)146. arXiv: 1402.2657 [hep-ph].

- [22] T. Sjöstrand, S. Mrenna, and P. Z. Skands, “PYTHIA 6.4 Physics and Manual”, *JHEP*, vol. 05, p. 026, 2006. DOI: 10.1088/1126-6708/2006/05/026. arXiv: hep-ph/0603175 [hep-ph].
- [23] T. Sjöstrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C. O. Rasmussen, and P. Z. Skands, “An Introduction to PYTHIA 8.2”, *Comput. Phys. Commun.*, vol. 191, pp. 159–177, 2015. DOI: 10.1016/j.cpc.2015.01.024. arXiv: 1410.3012 [hep-ph].
- [24] K. C. Zapp, J. Stachel, and U. A. Wiedemann, “A local Monte Carlo framework for coherent QCD parton energy loss”, *JHEP*, vol. 07, p. 118, 2011. DOI: 10.1007/JHEP07(2011)118. arXiv: 1103.6252 [hep-ph].
- [25] K. C. Zapp, F. Krauss, and U. A. Wiedemann, “A perturbative framework for jet quenching”, *JHEP*, vol. 03, p. 080, 2013. DOI: 10.1007/JHEP03(2013)080. arXiv: 1212.1599 [hep-ph].
- [26] K. C. Zapp, “JEWEL 2.0.0: directions for use”, *Eur. Phys. J. C*, vol. 74, no. 2, p. 2762, 2014. DOI: 10.1140/epjc/s10052-014-2762-1. arXiv: 1311.0048 [hep-ph].
- [27] Y. L. Dokshitzer and D. Kharzeev, “Heavy quark colorimetry of QCD matter”, *Phys. Lett. B*, vol. 519, pp. 199–206, 2001. DOI: 10.1016/S0370-2693(01)01130-3. arXiv: hep-ph/0106202.
- [28] J. Adam *et al.*, “Centrality and transverse momentum dependence of D^0 -meson production at mid-rapidity in Au+Au collisions at $\sqrt{s_{NN}} = 200$ GeV”, *Phys. Rev. C*, vol. 99, no. 3, p. 034908, 2019. DOI: 10.1103/PhysRevC.99.034908. arXiv: 1812.10224 [nucl-ex].
- [29] L. Cunqueiro and M. Płoskoń, “Searching for the dead cone effects with iterative declustering of heavy-flavor jets”, *Phys. Rev. D*, vol. 99, no. 7, p. 074027, 2019. DOI: 10.1103/PhysRevD.99.074027. arXiv: 1812.00102 [hep-ph].
- [30] S. Acharya *et al.*, “Measurement of the production of charm jets tagged with D^0 mesons in pp collisions at $\sqrt{s} = 7$ TeV”, *JHEP*, vol. 08, p. 133, 2019. DOI: 10.1007/JHEP08(2019)133. arXiv: 1905.02510 [nucl-ex].
- [31] M. Bahr *et al.*, “Herwig++ Physics and Manual”, *Eur. Phys. J. C*, vol. 58, pp. 639–707, 2008. DOI: 10.1140/epjc/s10052-008-0798-9. arXiv: 0803.0883 [hep-ph].
- [32] J. Bellm *et al.*, “Herwig 7.0/Herwig++ 3.0 release note”, *Eur. Phys. J. C*, vol. 76, no. 4, p. 196, 2016. DOI: 10.1140/epjc/s10052-016-4018-8. arXiv: 1512.01178 [hep-ph].
- [33] P. Nason, “A New method for combining NLO QCD with shower Monte Carlo algorithms”, *JHEP*, vol. 11, p. 040, 2004. DOI: 10.1088/1126-6708/2004/11/040. arXiv: hep-ph/0409146.
- [34] S. Frixione, P. Nason, and C. Oleari, “Matching NLO QCD computations with Parton Shower simulations: the POWHEG method”, *JHEP*, vol. 11, p. 070, 2007. DOI: 10.1088/1126-6708/2007/11/070. arXiv: 0709.2092 [hep-ph].

- [35] S. Alioli, P. Nason, C. Oleari, and E. Re, “A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX”, *JHEP*, vol. 06, p. 043, 2010. DOI: 10.1007/JHEP06(2010)043. arXiv: 1002.2581 [hep-ph].
- [36] R. Vértési, “Jet measurements with ALICE: substructure, dead cone, charm jets”, *PoS*, vol. LHCP2020, B. Mansoulie, G. Marchiori, R. Salern, and T. Bos, Eds., p. 143, 2021. DOI: 10.22323/1.382.0143. arXiv: 2009.11228 [hep-ex].
- [37] A. M. Sirunyan *et al.*, “Measurement of b jet shapes in proton-proton collisions at $\sqrt{s} = 5.02$ TeV”, May 2020. arXiv: 2005.14219 [hep-ex].
- [38] —, “Comparing transverse momentum balance of b jet pairs in pp and PbPb collisions at $\sqrt{s_{NN}} = 5.02$ TeV”, *JHEP*, vol. 03, p. 181, 2018. DOI: 10.1007/JHEP03(2018)181. arXiv: 1802.00707 [hep-ex].
- [39] J. Adam *et al.*, “Measurement of inclusive charged-particle jet production in Au + Au collisions at $\sqrt{s_{NN}} = 200$ GeV”, *Phys. Rev. C*, vol. 102, no. 5, p. 054913, 2020. DOI: 10.1103/PhysRevC.102.054913. arXiv: 2006.00582 [nucl-ex].
- [40] A. M. Sirunyan *et al.*, “First measurement of large area jet transverse momentum spectra in heavy-ion collisions”, Feb. 2021. arXiv: 2102.13080 [hep-ex].
- [41] “Measurement of suppression of large-radius jets and its dependence on substructure in Pb+Pb at 5.02 TeV by ATLAS detector”, CERN, Geneva, Tech. Rep., Nov. 2019, All figures including auxiliary figures are available at <https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/CONFNOTES/ATLAS-CONF-2019-056>. [Online]. Available: <http://cds.cern.ch/record/2701506>.
- [42] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, *Dive into Deep Learning*. 2020, <https://d2l.ai>.
- [43] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.
- [44] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.
- [45] R. Szeliski, *Computer Vision: Algorithms and Applications*, 1st. Berlin, Heidelberg: Springer-Verlag, 2010, ISBN: 1848829345.
- [46] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004, ISBN: 0521833787.
- [47] L. Bottou, “On-line learning in neural networks”, in D. Saad, Ed., New York, NY, USA: Cambridge University Press, 1998, ch. On-line Learning and Stochastic Approximations, pp. 9–42, ISBN: 0-521-65263-4. [Online]. Available: <http://dl.acm.org/citation.cfm?id=304710.304720>.

- [48] L. Bottou and O. Bousquet, “The tradeoffs of large scale learning”, in *Proceedings of the 20th International Conference on Neural Information Processing Systems*, ser. NIPS’07, Vancouver, British Columbia, Canada: Curran Associates Inc., 2007, pp. 161–168, ISBN: 978-1-60560-352-0. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2981562.2981583>.
- [49] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition”, in *Proceedings of the IEEE*, vol. 86, 1998, pp. 2278–2324. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.42.7665>.
- [50] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, “Efficient backprop”, in *Neural Networks: Tricks of the Trade, This Book is an Outgrowth of a 1996 NIPS Workshop*, London, UK, UK: Springer-Verlag, 1998, pp. 9–50, ISBN: 3-540-65311-2. [Online]. Available: <http://dl.acm.org/citation.cfm?id=645754.668382>.
- [51] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines”, in *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, J. Fürnkranz and T. Joachims, Eds., Omnipress, 2010, pp. 807–814. [Online]. Available: <https://icml.cc/Conferences/2010/papers/432.pdf>.
- [52] J. T. Barron, “Continuously differentiable exponential linear units”, *CoRR*, vol. abs/1704.07483, 2017. arXiv: 1704.07483. [Online]. Available: <http://arxiv.org/abs/1704.07483>.
- [53] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition”, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016.
- [54] R. Arandjelovic, P. Gronát, A. Torii, T. Pajdla, and J. Sivic, “Netvlad: CNN architecture for weakly supervised place recognition”, *CoRR*, vol. abs/1511.07247, 2015. arXiv: 1511.07247. [Online]. Available: <http://arxiv.org/abs/1511.07247>.
- [55] J. H. Friedman, “Stochastic gradient boosting”, *Computational Statistics and Data Analysis*, vol. 38, no. 4, pp. 367–378, 2002, Nonlinear Methods and Data Mining, ISSN: 0167-9473. DOI: [https://doi.org/10.1016/S0167-9473\(01\)00065-2](https://doi.org/10.1016/S0167-9473(01)00065-2). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167947301000652>.
- [56] —, “Greedy function approximation: A gradient boosting machine.”, *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001. DOI: 10.1214/aos/1013203451. [Online]. Available: <https://doi.org/10.1214/aos/1013203451>.
- [57] A. M. Sirunyan *et al.*, “Identification of heavy-flavour jets with the CMS detector in pp collisions at 13 TeV”, *JINST*, vol. 13, no. 05, P05011, 2018. DOI: 10.1088/1748-0221/13/05/P05011. arXiv: 1712.07158 [physics.ins-det].

- [58] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting”, *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014. [Online]. Available: <http://jmlr.org/papers/v15/srivastava14a.html>.
- [59] I. Loshchilov and F. Hutter, “SGDR: stochastic gradient descent with restarts”, *CoRR*, vol. abs/1608.03983, 2016. arXiv: 1608.03983. [Online]. Available: <http://arxiv.org/abs/1608.03983>.
- [60] T. Sjöstrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C. O. Rasmussen, and P. Z. Skands, “An Introduction to PYTHIA 8.2”, *Comput. Phys. Commun.*, vol. 191, pp. 159–177, 2015. DOI: 10.1016/j.cpc.2015.01.024. arXiv: 1410.3012 [hep-ph].
- [61] J. Adam *et al.*, “Centrality and transverse momentum dependence of D^0 -meson production at mid-rapidity in Au+Au collisions at $\sqrt{s_{NN}} = 200$ GeV”, *Phys. Rev. C*, vol. 99, no. 3, p. 034908, 2019. DOI: 10.1103/PhysRevC.99.034908. arXiv: 1812.10224 [nucl-ex].
- [62] G. Contin *et al.*, “The STAR MAPS-based PiXeL detector”, *Nucl. Instrum. Meth. A*, vol. 907, pp. 60–80, 2018. DOI: 10.1016/j.nima.2018.03.003. arXiv: 1710.02176 [physics.ins-det].
- [63] J. Rusnak, “Jet Reconstruction in Au+Au collisions at RHIC”, Ph.D. dissertation, Nuclear Physics Institute, The Czech Academy of Sciences, 2017. [Online]. Available: https://drupal.star.bnl.gov/STAR/system/files/phd_thesis_rusnak.pdf.

■ Contributions

■ Statement of Contribution

The results of this thesis are based on the publications shown below. My explicit contributions were:

- Implementation of FastSim and parton matching within Monte Carlo simulations.
- Implementation of the JetVLAD model.
- Computational experiments.
- Analysis of the obtained results.

All codes developed as part of this thesis are in the public domain and are available at [ponimatkin/NetVLAD-tagger-pytorch](#) (tagger implementation) and [ponimatkin/jet-generator-tagging](#) (Monte Carlo simulation code) at GitHub service.

■ Publications in Peer Reviewed Journals

J. Bielcikova, R. Kunnawalkam Elayavalli, **G. Ponimatkin**, J. H. Putschke, J. Sivic. "Identifying Heavy-Flavor Jets Using Vectors of Locally Aggregated Descriptors". In JINST 16 (2021) 03, P03017. arXiv:2005.01842 [hep-ph].

■ Conference Proceedings

G. Ponimatkin. Identifying heavy-flavor jets using vectors of locally aggregated descriptors. In 20th Conference of Czech and Slovak Physicists, Proceedings, 2020, pp. 51–52.

■ Talks at Conferences

G. Ponimatkin. Identifying heavy-flavor jets using vectors of locally aggregated descriptors. XXVIII International Workshop on Deep-Inelastic Scattering and Related Subjects, April 12-16 2021, Stony Brook University.

Identifying heavy-flavor jets using vectors of locally aggregated descriptors

J. Bielčková,^{a,b} R. Kunnawalkam Elayavalli,^c G. Ponimatkin,^{a,b,d,*} J.H. Putschke^c
and J. Sivic^{d,e,f}

^aDepartment of Nuclear Spectroscopy, Nuclear Physics Institute, Czech Academy of Sciences, Hlavní 130, 250 68 Řež, Czech Republic

^bDepartment of Physics, Czech Technical University in Prague, FNSPE, Prague 115 19, Czech Republic

^cDepartment of Physics and Astronomy, Wayne State University, 666 W. Hancock, Detroit, U.S.A.

^dCzech Institute of Informatics, Robotics and Cybernetics, Czech Technical University, Prague, Czech Republic

^eWillow team, Inria, Paris, France

^fEcole Normale Supérieure, PSL Research University, Paris, France

E-mail: ponimatkin@ujf.cas.cz

ABSTRACT: Jets of collimated particles serve a multitude of purposes in high energy collisions. Recently, studies of jet interaction with the quark-gluon plasma (QGP) created in high energy heavy ion collisions are of growing interest, particularly towards understanding partonic energy loss in the QGP medium and its related modifications of the jet shower and fragmentation. Since the QGP is a colored medium, the extent of jet quenching and consequently, the transport properties of the medium are expected to be sensitive to fundamental properties of the jets such as the flavor of the parton that initiates the jet. Identifying the jet flavor enables an extraction of the mass dependence in jet-QGP interactions. We present a novel approach to tagging heavy-flavor jets at collider experiments utilizing the information contained within jet constituents via the JetVLAD model architecture. We show the performance of this model in proton-proton collisions at center of mass energy $\sqrt{s} = 200$ GeV as characterized by common metrics and showcase its ability to extract high purity heavy-flavor jet sample at various jet momenta and realistic production cross-sections including a brief discussion on the impact of out-of-time pile-up. Such studies open new opportunities for future high purity heavy-flavor measurements at jet energies accessible at current and future collider experiments.

KEYWORDS: Data processing methods; Pattern recognition, cluster finding, calibration and fitting methods; Performance of High Energy Physics Detectors; Software architectures (event data models, frameworks and databases)

ARXIV EPRINT: [2005.01842](https://arxiv.org/abs/2005.01842)

*Corresponding author.

Contents

1	Introduction	1
2	Datasets and inputs	3
3	JetVLAD-model	5
4	Classification performance metrics	7
5	Sensitivity to hyper-parameters and model uncertainties	8
6	Heavy-flavor jet tagging at RHIC energies	9
7	Conclusions and outlook	11
A	Fast simulation	13

1 Introduction

Jets arise from hard scattering of quarks/gluons in high energy collisions resulting in a collection of collimated particles in the detector. Jets are multi-scale objects that are sensitive to perturbative physics, such as in their production and in their parton shower or evolution [1–7] and non-perturbative effects such as in hadronization [8–10]. Recent theoretical advancements have extended quantum chromo-dynamics (QCD) calculations of the jet production cross-sections to beyond leading-order [11] and leading-log [12–14] and their resummations results in predictions that reproduce trends in data over several orders of magnitude for different collision energies. In addition to being useful in extracting the strong coupling constant (α_S) [15], jets have irreversibly established themselves as key probes of QCD, searches of beyond the standard model particles and in explorations of the quark-gluon plasma (QGP) produced in relativistic heavy ion collisions. Jets produced in heavy ion collisions undergo a phenomenon called jet quenching, which manifests as energy-loss and modifications to the jet structure due to interactions with the QGP medium. Jet quenching is an important signature of the QGP and we can extract the medium transport properties by comparing data with theoretical calculations of energy loss [16, 17]. For more details regarding jets in heavy ion collisions, we refer the reader to these review articles from experiment [18] and theory [19, 20]. Thus, in both these seemingly orthogonal areas of jet physics, the ability to identify and characterize a jet based on fundamental properties in relation to its progenitor parton such as its energy, momentum and flavor are desired.

In this paper we focus on the topic of identifying or tagging a jet based on the flavor of the hard scattered parton. In the case of jet quenching studies, knowing the jet flavor presents an opportunity to systematically study the mass dependence of parton energy loss in the QGP. When

only considering QCD radiations, the mass of the radiating particle effectively controls the phase space of the radiation as prescribed in the dead-cone effect [21]. The dead-cone effect has been measured and studied in electron-positron collisions [22] and recently explored in pp collisions at the Large Hadron Collider (LHC) [23–25]. In heavy ion collisions, the mass dependence of energy loss is still an open question with measurements at the LHC [26, 27] showing no significant differences between jets identified as heavy-flavor jets originating from a b/c -quark to light-flavor (gluons and u, d, s -quarks). It is possible that jets at the LHC (momenta $O(100 - 1000)$ GeV) are in high energy domain where the originating parton mass does not play a significant role in its interactions with the medium. Therefore such studies are especially important at the Relativistic Heavy Ion Collider (RHIC), where the smaller center of mass energy ($\sqrt{s} = 200$ GeV) produces jets with momenta $O(10)$ GeV which are expected to a greater extent to undergo interaction with medium and thus have enhanced sensitivity to parton flavor and mass [28, 29]. The smaller collision energy however makes the measurement challenging due to the significantly smaller jet production cross-section which in turn introduces a dependence on available statistics.

We present a machine learning (ML) model which utilizes experimental information based on jet and its constituents and we identify and tag populations of light- and heavy-flavor jets with increased efficiency and purity as compared to current state of the art classifications algorithms. In contrast to light quarks, the heavy-flavor quarks are produced early in the hard scattering due to their large mass and travel a significant distance in the detector before they decay. Upon jet evolution involving fragmentation and hadronization, these massive quarks leave a characteristic experimental signature of charged particle tracks pointing back to a displaced (secondary) vertex, as opposed to the primary vertex which corresponds to the hard scattering point of interaction. These vertices can be measured by high-resolution tracking detectors. Since these displaced vertices are an important feature of heavy-flavor jets, classification algorithms predominantly take into consideration some experimental quantity related to the displaced vertex such as the distance of closest approach (DCA), or the secondary vertex mass amongst others [30, 31]. At LHC energies, it is important to note that highly-virtual gluons originating from the hard scattering could produce jets that mimic heavy-flavor jets [32, 33]. This process is often treated as a part of the background since the gluon can split to a pair of heavy-flavor quarks during its evolution which could behave as a heavy-flavor jet. At the center of mass energies available at RHIC [34], this gluon splitting process is significantly suppressed due to the jet kinematics and the steeply falling parton momentum spectra.

There are two general categories of measurements involving identifying heavy-flavor jets i.e, ensemble based approaches and jet-by-jet approaches. Extracting the heavy-flavor jet fraction from an inclusive jet sample is typically done via template fits to utilizing distributions of signal and background. The latter approach of identifying jets individually by associating a light- or heavy-flavor probability is more adaptable to ML approaches. The early examples of such taggers employed boosted-decision trees (BDT) and shallow neural networks (NN) to train on a sample of signal and background jets which were subsequently applied on data after correcting for the differences between data and the simulations [30, 31, 35]. Current state of the art studies and measurements at the LHC have expanded to include deep, convolutional and recurrent networks [36–39]. The classification procedure where the networks were trained on MC with associated signal/background labels is commonly known as supervised training and is dependent to an extent on the MC accurately representing data. Since jets are essentially collections of objects (tracks/towers in experiment and particles in

MC), a majority of the high performing heavy-flavor classification models currently used in experiment utilize information contained within these jet constituents. Experiments with charged particle and vertex tracking detectors with high pointing resolution, $O(10-100\mu m)$, can associate tracks originating from different vertices. Providing the jet constituents (4-momenta and vertex information) to a sufficiently complicated model should effectively include all available physics required to distinguish between heavy- and light-flavor jets. We introduce a model that utilizes these jet constituents and study the performance in detail for jets of varying momenta and for varying categories of inputs. The rest of the paper is organized as follows. The MC samples are outlined in section 2 along with a discussion of the different inputs types to the classification model. The JetVLAD model architecture is presented in section 3 and we present a discussion of performance metrics that are studied in this paper in section 4. We discuss the results for RHIC energies in section 6 and conclude our study with an outlook focusing on applicability in current and future experiments in section 7.

2 Datasets and inputs

We use PYTHIA 8.235 [40] to generate di-jet events in proton-proton (pp) collisions at $\sqrt{s} = 200$ GeV. In order to maximize the classification performance, we name two classes of jets as light (originating from gluon, u, d, s quarks) and heavy-flavor (c, b quarks). Flavor labelling is done by a requiring the initiating parton to be contained within the jet radius. Recently, the heavy flavor tagging community has started to look at jet-flavor association by utilizing reconstructed mesons from heavy flavor quark such as D_0 [41]. We will explore this style of identification/tagging in an upcoming publication but for the purposes of this paper, we utilize the hard-scattered quark to jet matching for defining our jet classes. To compare the effect of the production cross-section, we produce two sets of samples which are labeled as follows:

- Cross-section weighted
- Balanced — 50% light, 25% c -jet and 25% b -jet.

Particle decays in PYTHIA along $x - y$ and z are limited to maximum distances to 2000 mm and 600 mm, respectively. For each dataset, we generate both light- and heavy-flavor di-jet events with the invariant $p_{\hat{T}}$ corresponding to [3 – 12], [8 – 17], [13 – 22], [18 – 27] and [23 – 42] GeV/ c . The overlap in the upper and lower limits is to maximize statistics when combining the datasets together.¹ The datasets are split into 80 : 10 : 10 for training, testing and validation covering a total of 2 million events for the balanced sample and 4 million events for the cross-section weighted sample.

In order to simulate particle interaction with the detector, we apply a fast simulation (Fast-Sim) of the STAR detector [42]. The Fast-Sim framework includes a parametrization of charged particle tracking efficiency, momentum resolution smearing and secondary vertex DCA smearing according to the STAR Time Projection Chamber (TPC) [43] and Heavy-Flavor Tracker (HFT) [44], respectively. The Fast-Sim procedure is outlined in greater detail in appendix A. Post smearing, we reconstruct jets from all smeared charged particles using the anti- k_t reconstruction algorithm [45] as implemented in FastJet [46] with a jet resolution parameter $R = 0.4$. The charged particles which

¹The samples used in this study are made available along with all the necessary software tool-kits for processing and training/testing upon publication.

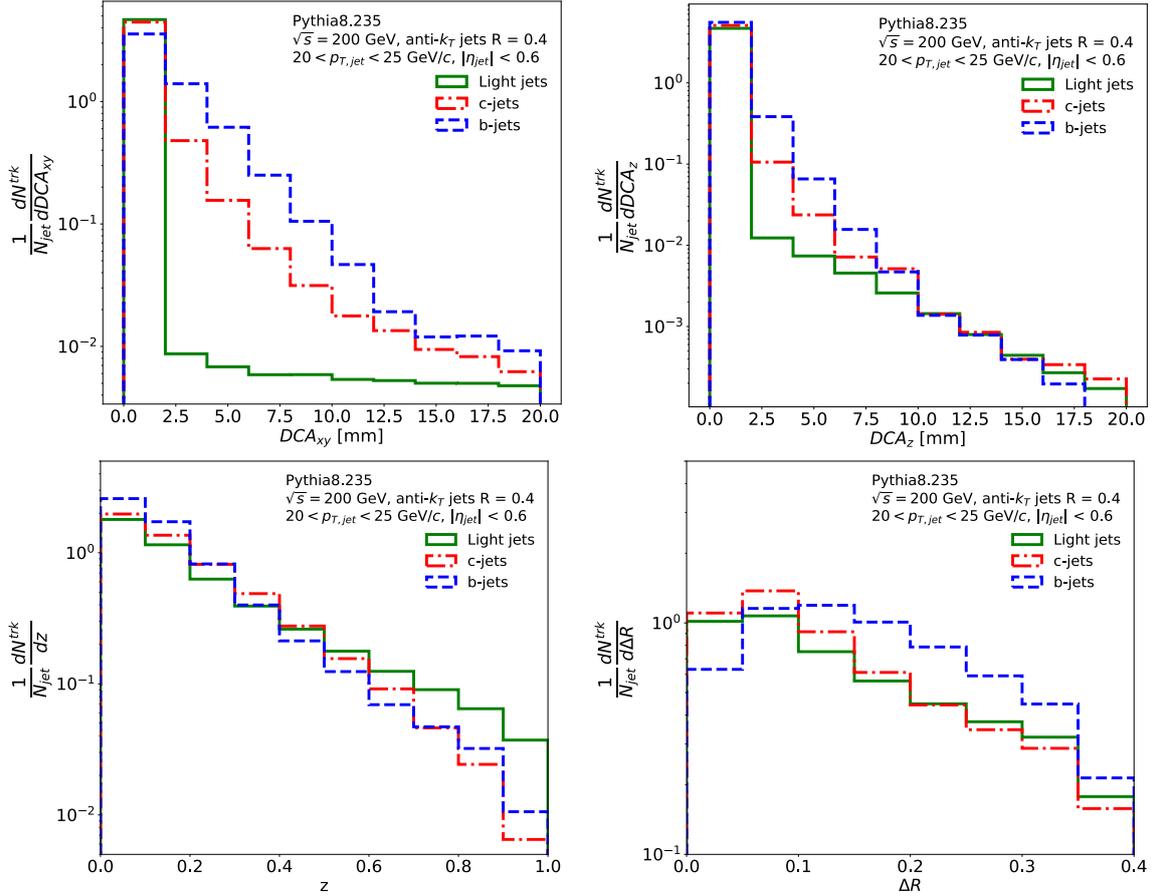


Figure 1. Distributions of a subset of input variables DCA_{xy} (top left), DCA_z (top right), z (bottom left) and ΔR (bottom right) for light (black), c (red) and b-jets (blue) as generated in PYTHIA. (COLOR ONLINE).

are associated to the jet are then further considered for classification procedure. These jets are often referred to as charged-jets and future extensions of this study will consider both charged and neutral components of a jet, taking into account the energy depositions recorded by both electromagnetic and hadronic calorimeters.

We separate the inputs into three different types including: tracking, fragmentation and secondary vertices as shown and defined in table 1. For each of these classes, the inputs correspond to the values mentioned for each track in the jet. We do not take into account the particle ID or mass which are left for future studies. In addition to track kinematic variables, we also include one high level observable related to jet fragmentation such as the $z\Delta R^2$, a quantity related to the jet mass when summed over all particles [47]. Furthermore, by comparing the performance of the model with tracking and vertexing or tracking and fragmentation as inputs, one can estimate the impact such information will have on the tagging of heavy-flavor jets. This can help to clarify detector performance needed in future experiments or in upgrading current experiments.

The input distributions for $20 < p_T < 25$ GeV/c jets are shown in figure 1, DCA_{xy} (top left), DCA_z (top right), z (bottom left) and ΔR (bottom right) for light jets in the green solid lines and heavy-flavor separated into b and c -jets in the blue dashed and red dot-dashed lines, respectively.

Table 1. Input variables and their types and definitions utilized in the classification.

Type	Inputs	Definition
Tracking	p_T	Transverse momentum in the $x - y$ plane
	η	pseudorapidity
	ϕ	azimuthal angle
Fragmentation	z	momentum fraction $\frac{p_T^{track}}{p_T^{jet}}$
	ΔR	distance between track and jet axis $\sqrt{\Delta\phi^2 + \Delta\eta^2}$
	$z\Delta R^2$	higher level feature
Secondary Vertex	DCA_{xy}	Distance of closest approach in $x - y$
	DCA_z	Distance between primary and secondary vertex in the z axis

As the mass of the originating parton increases, we find jets to have a larger width in the secondary vertex and distinctive shifts in the opening angles and fragmentation. For the purposes of the training, we combine both b and c -jets into a single class called heavy-flavor jets.

The MC sample utilized for training including the Fast-Sim procedure, has been shown to be comparable to the STAR detector performance in previous publications [48, 49]. Since the training sample is at the detector-level (i.e after Fast-Sim), any unfolding corrections for detector effects/uncertainties are expected to be done post classification. Given that the tracking efficiency within STAR is $\pm 4\%$ [49], we find no significant difference to the jet energy scale/resolution in the light- and heavy-flavor jet samples. As such, in the rest of the paper we will discuss jets (and their momenta) at the detector-level.

Before we train our model, variables are normalized to their mean to ensure uniformity amongst datasets, which is a necessary step in ML often referred to as pre-processing.

3 JetVLAD-model

There are several different ways of representing jets for machine learning tasks [50]. Two of these are the graph-based or the set-based approach. The former comes from the fact that we can traverse back through the jet clustering history to recreate in some metric, the original parton shower. While such a graphical representation might be considered a bonus for jet flavor tagging, it has disadvantages, namely that such ordering is clearly dependent on the underlying model of parton shower evolution and also factors in the specific algorithm utilized for the de-clustering. This might lead to a wider domain gap between simulated jets from MC which are used for the training of a machine learning algorithm and experimentally reconstructed jets in data.

The set based approach utilizes a simpler view, where we can consider a jet as a set or collection of particles. Such approaches could also be dependent on the type of algorithm used for jet finding but those effects are typically small. As an appreciable consequence, this approach has a smaller domain gap due to an easier theoretical description of jets at this level and it has less model dependence than the graph based description. Another benefit of this approach is that primary jet

finding is predominantly done using anti- k_t jet clustering algorithm and such jets do not have a physical clustering tree (in comparison to the k_t [51] and C/A [52] algorithms).

We begin description of our model by formalizing the dataset notation. We are given a jet \mathcal{J} composed of a set of particles which are created by an action of the anti- k_t jet clustering algorithm, i.e.

$$\mathcal{J} = \{(p_{T,i}, \eta_i, \phi_i, \dots)\}_{i=1}^n, \quad (3.1)$$

where n corresponds to the total number of jet constituents. In recent publications, set represented jets were classified using Recurrent Neural Network (RNN) models [53], where an artificial ordering in p_T or vertex distance was introduced. To overcome such an arbitrary ordering, one might use an aggregation method, that will allow us to characterize a set of inputs into a fixed-length feature vector. An analogous situation is found in computer vision with respect to the procedure of place recognition, where one wants to recognize a landmark (say the Eiffel tower). In this case we often deal with a variable size set of feature descriptors extracted from the image. For example, depending on the place where the photo was taken, we might have variable amount of trees, cars and other background objects.

The NetVLAD is an adaptive pooling layer that takes a set of feature descriptors as an input and returns a fixed-length feature vector that characterizes each set [54, 55]. While in computer vision one needs a feature extractor that yields meaningful descriptors, in physics, observables already hold rich information regarding the jet shower. Thus we can omit the feature extractor stage and use tracks belonging to a jet which we call henceforth as particle descriptors. The ordering of the particle descriptors in the input was varied in a randomized fashion with no particular ordering having an impact on the classification performance.

Given a jet \mathcal{J} with n tracks, each of which is represented by a d -dimensional particle descriptor, as described in equation (3.1), we define k clusters in the input space of the model, where each cluster is represented by parameter vectors \mathbf{w}_k , \mathbf{c}_k and scalar b_k that are learnt from data. The output of NetVLAD layer is a $d \times k$ -dimensional matrix, whose elements are given by

$$\mathbf{v}_{j,k} = \sum_{i=1}^n \frac{e^{\mathbf{w}_k^T \mathbf{x}_{i,j} + b_k}}{\sum_{k'} e^{\mathbf{w}_{k'}^T \mathbf{x}_{i,j} + b_{k'}}} (\mathbf{x}_{i,j} - \mathbf{c}_{k,j}). \quad (3.2)$$

Here $\mathbf{x}_{i,j}$ is a j -th element of the i -th particle descriptor and $\mathbf{c}_{k,j}$ is the j -th element of the k -th cluster center vector. This matrix is then L^2 normalized column-wise, transformed into a vector and then again L^2 normalized. NetVLAD hence summarizes a set of particle descriptors into one fixed-length feature vector, that is then fed into the standard feed-forward neural network. Please note that the vectors \mathbf{w}_k , \mathbf{c}_k and scalar b_k for each of cluster are parameters of this NetVLAD layer and learnt from data (together with other parameters of the model) in a discriminating manner using back-propagation as described below.

We chose our network architecture to mimic the ResNet model family [56], by utilizing residual blocks with batch normalization, in order to simplify the learning problem. Width of our model was chosen to be the same as the output of NetVLAD layer. We also utilize DropOut method [57] in order to increase generalization of our chosen model. Our total architecture thus can be written as

$$\text{JetVLAD} = \text{NetVLAD}(N_c) \rightarrow D \times [\text{ResidualBlock}] \rightarrow \text{Softmax}, \quad (3.3)$$

where N_c is the number of clusters and D is the depth. We train our model using momentum stochastic gradient descent (SGD) [58] with cosine annealing and a warm restart. We chose learning rate of 0.013, and annealing parameters $T_0 = 1$ and $T_{\text{mult}} = 3$ by utilizing random grid search. The model was trained for maximum of 2000 epochs, with early stopping criterion of 10 epochs used. We also found that number of clusters $N_c = 33$ and depth $D = 4$ were good set of hyper-parameters.

4 Classification performance metrics

In order to evaluate performance of the model one has to choose a meaningful set of metrics that will quantify key aspects of model performance. The first metric is called efficiency, in physics or true positive rate (TPR), in machine learning/computer vision and it is defined as

$$\text{TPR} = \frac{\text{TP}}{\text{P}}. \quad (4.1)$$

Here TP is a number of positively identified heavy-flavor jets and P is a total number of heavy-flavor jets in the testing sample. Hence, this metric tells us the fraction of the signal that the model will extract from the sample.

Next metric that is closely related to the efficiency is mis-identification probability, in physics or false positive rate (FPR), in machine learning/computer vision,

$$\text{FPR} = \frac{\text{FP}}{\text{N}}. \quad (4.2)$$

Here FP is a number of false-positive samples identified in the testing sample and N is a total number of background objects in the testing sample. This metric quantifies the amount of background that still persists in the signal post classification. Another related metric is background rejection (REJ), which has no analogies in machine learning literature, and is given by

$$\text{REJ} = \frac{1}{\text{FPR}}. \quad (4.3)$$

This determines how much of the true background will be rejected per one false-positive detection. It is a useful quantity particularly for heavy-flavor jet classification where the signal is two orders of magnitude smaller than the background due to the difference in the production cross-sections.

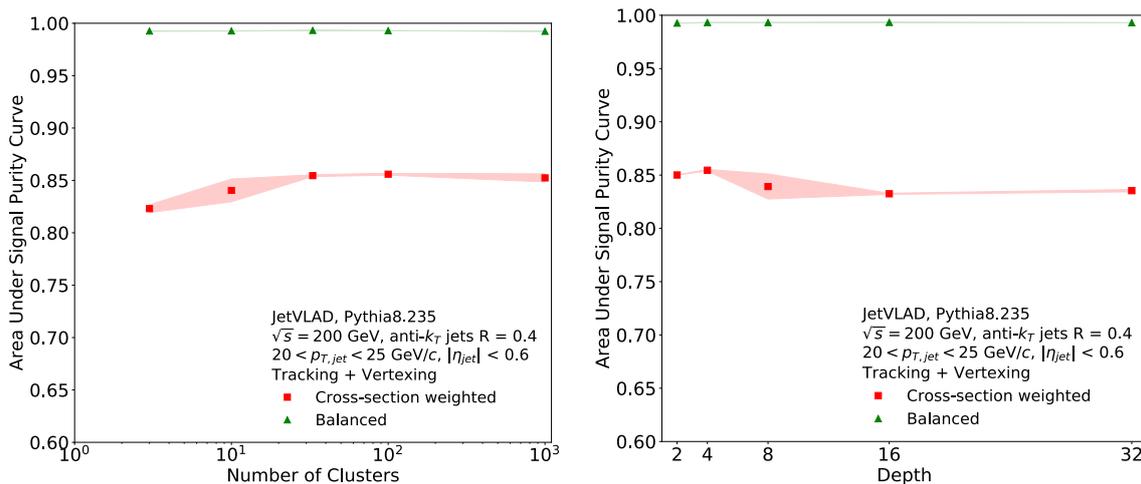
Last relevant metric is purity, in physics or precision, in machine learning and it is given by

$$\text{PREC} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (4.4)$$

where TP is a number of true positive objects found in the testing sample and FP is a number of false positive objects found in the testing sample. As its name suggests, this metric tells us the extent of contamination in the signal with false-positive objects. Summary of all the metrics and their definitions are given in table 2.

Table 2. Classification metrics used in physics and machine learning.

Physics	Machine Learning	Definition
Tagging Efficiency	True Positive Rate (TPR)/Recall	$TPR = \frac{TP}{P}$
Misidentification Prob.	False Positive Rate (FPR)	$FPR = \frac{FP}{N}$
Background Rejection		$REJ = \frac{1}{FPR}$
Signal Purity	Precision	$PREC = \frac{TP}{TP+FP}$

**Figure 2.** Hyper-parameter scans for NetVLAD layer showing the performance for heavy-flavor identification purity as we vary the number of clusters (left) and the model depth (right).

5 Sensitivity to hyper-parameters and model uncertainties

In order to study the dynamics of our model, we run a study to quantify effects of hyper-parameter change on the model performance. In each run, we fixed one of the parameters (for example, number of clusters) and varied the other one (model depth). Results for cluster scan, during which we fixed depth of the model $D = 4$, and varied the number of clusters can be seen in the left panel of figure 2. The area under the purity curve is used as a metric of performance and we see no significant improvement beyond a total of 33 clusters. Similarly, the depth hyper-parameter sensitivity study shown in the right panel figure 2 during which we fixed $N_c = 33$.

Estimating systematic uncertainties of a deep neural networks is a rather recent development with different ideas [59–61]. Given that the model hyper-parameters were fixed for optimal performance, modifying those is not an appropriate way to truly estimate the uncertainties inherent in the model. Given a fixed representation of JetVLAD used in this study, we find a total of 111608 trainable parameters. Each of these parameters or weights are initially randomized and are later fixed during the training. We further randomize these input weights with three iterations of the model training and the different results are taken as a systematic variation of the classification and show in the corresponding shaded regions with their average taken as the central value.

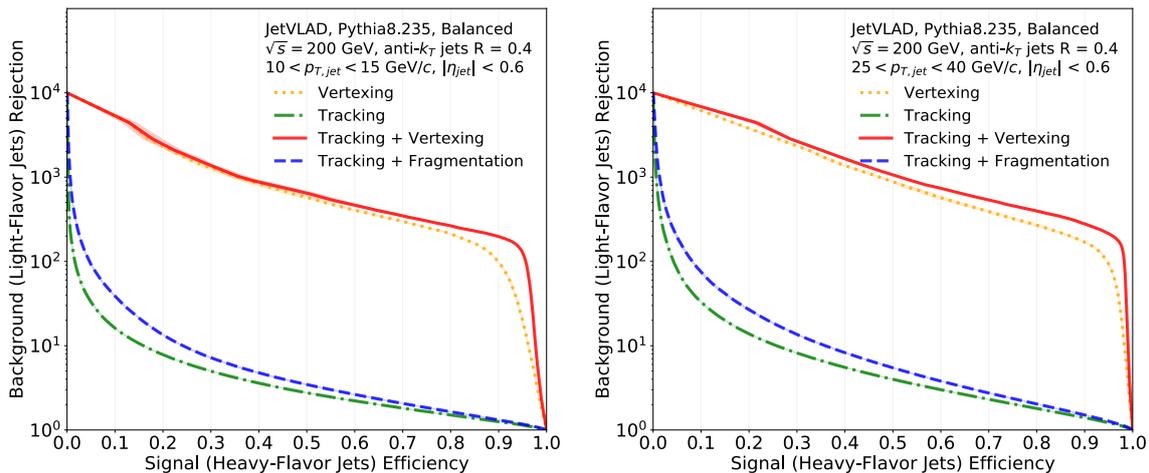


Figure 3. Background rejection vs efficiency curves shown for different inputs in the lines (described in the text) for balanced samples. The left and right panels show jets with $10 < p_T < 15$ and $25 < p_T < 40$ GeV/ c .

6 Heavy-flavor jet tagging at RHIC energies

Each sample of heavy-flavor jets, cross-section weighted and balanced, is trained and validated in parallel. The cross-section weighting is included in the datasets and is not considered explicitly in the training procedure. Once trained, we have two sets of model weights corresponding to the different datasets, which we can then use to further test the performance of JetVLAD on the respective samples. The background rejection vs efficiency curves for the balanced dataset are shown in figure 3, where the left and right panels represent $R = 0.4$ jets with $10 < p_T < 15$ GeV/ c and $20 < p_T < 40$ GeV/ c , respectively.

Each curve in the plot represents different inputs to the model such as vertexing (dotted yellow), tracking (dot-dashed green), tracking + vertexing (solid red) and tracking + fragmentation (dashed blue). Similarly the performance for the cross-section weighted sample is shown in figure 4. For both weighted and balanced datasets, we find the secondary vertex information provides the maximal impact on tagging heavy-flavor jets. The inclusion of tracking information in the input shows improved background rejection at large efficiencies while models trained with fragmentation information show a slight improvement on top of tracking at small efficiencies but do not significantly improve the performance at large efficiencies.

We also study the purity vs efficiency for the different inputs and weighted samples in figure 5. In contrast to the background rejection that did not show a large effect when considering the cross-section weighting, the purity on the other hand shows a distinct improvement for balanced samples. The purity for balanced dataset being close to 100% can be understood due to the unrealistic yield of heavy-flavor jets in the sample. A more realistic performance, comparable to experimental data is shown for the cross-section weighted sample where for 80% efficiency, we have close to 80% purity and a background rejection of 236. In both studies, we find the tracking + vertexing still has the largest purity at given efficiency.

The purity (left) and background rejection (right) as a function of jet momenta for the tracking + vertexing input are shown in figure 6. We find a consistent trend of increasing background rejections

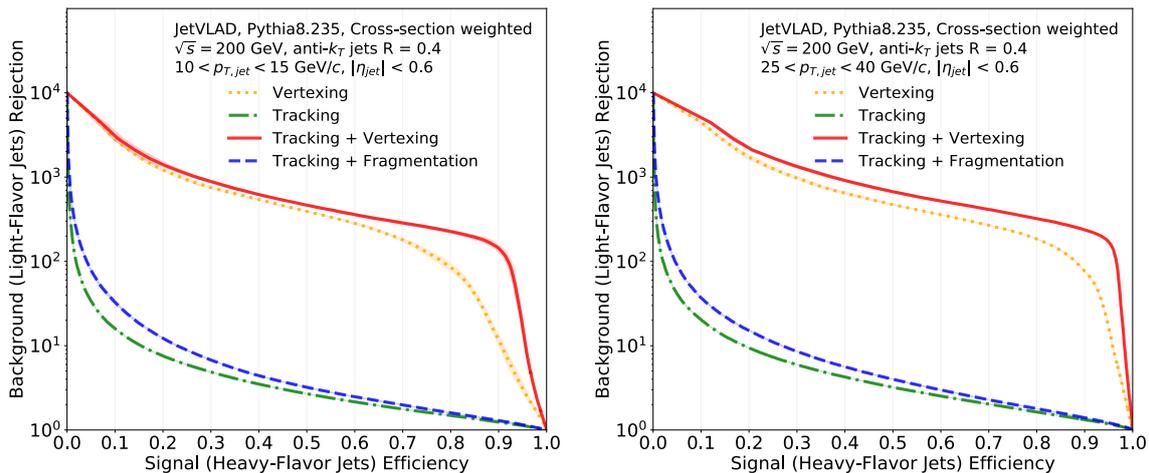


Figure 4. Background rejection vs efficiency curves shown for different inputs for cross-section weighted samples. The left and right panels show jets with $10 < p_T < 15$ and $25 < p_T < 40$ GeV/c.

at given efficiencies with increasing jet p_T . At fixed purity of 70%, we also find a similar trend as before with increasing efficiency. At signal purity closer to 80%, we find an interesting trend for the highest momenta jets where the efficiency drops significantly indicating a kinematic effect of the jets and their substructure. We understand the drop in purity for the highest jet momenta to partly be due to the overlap in a high-level feature space between the light- and heavy-flavor samples and a further exploration of this behavior is reserved for an upcoming publication.

We choose two working points based on efficiencies of 81% and 50% for which the corresponding purity and background rejections in cross-section weighted and balanced datasets are shown in table 3. As we increase the jet p_T , we generally find that the signal purity is relatively consistent for both working points with a notable exception at the lowest and highest jet momenta. The background rejection increases in a non-linear fashion as jet momenta increase with the working point at 50% efficiency having a rejection similar to the difference in the cross-sections. Extending the tagging to the lowest jet energies we see a drop in the purity at higher efficiencies due to the overlap between the jet topologies for light- and heavy-flavor jets where the reduced boost resulting in a smaller overall DCA .

In our study, we applied a fast simulation of the STAR detector to compare the performance at the detector-level. In order for such a network to be fully applicable on data, one needs to also include additional effects such as out-of-time pileup for example. As the luminosity of the beams increases, the probability to have particles in your jet that arise from pileup vertices increases and these additional particles can affect the performance (since they tend to have very large DCA values). The impact of out-of-time pile-up contribution to jets were studied by embedding minimum-bias PYTHIA events along with our hard-scattered event. The extent of pile-up contribution is dependent on the beam luminosities and data-taking rate and since they are relatively small at RHIC compared to the LHC, we run two scenarios of including 1 pile-up vertex, which is randomly placed along the z -direction of STAR’s acceptance. We see an overall degradation of $\approx 2 - 3\%$ in the tagging purity at fixed efficiency (80%). As the classification performances are non-monotonic to input variation, we recommend retraining the weights with an accurate simulation of pileup contribution to negate

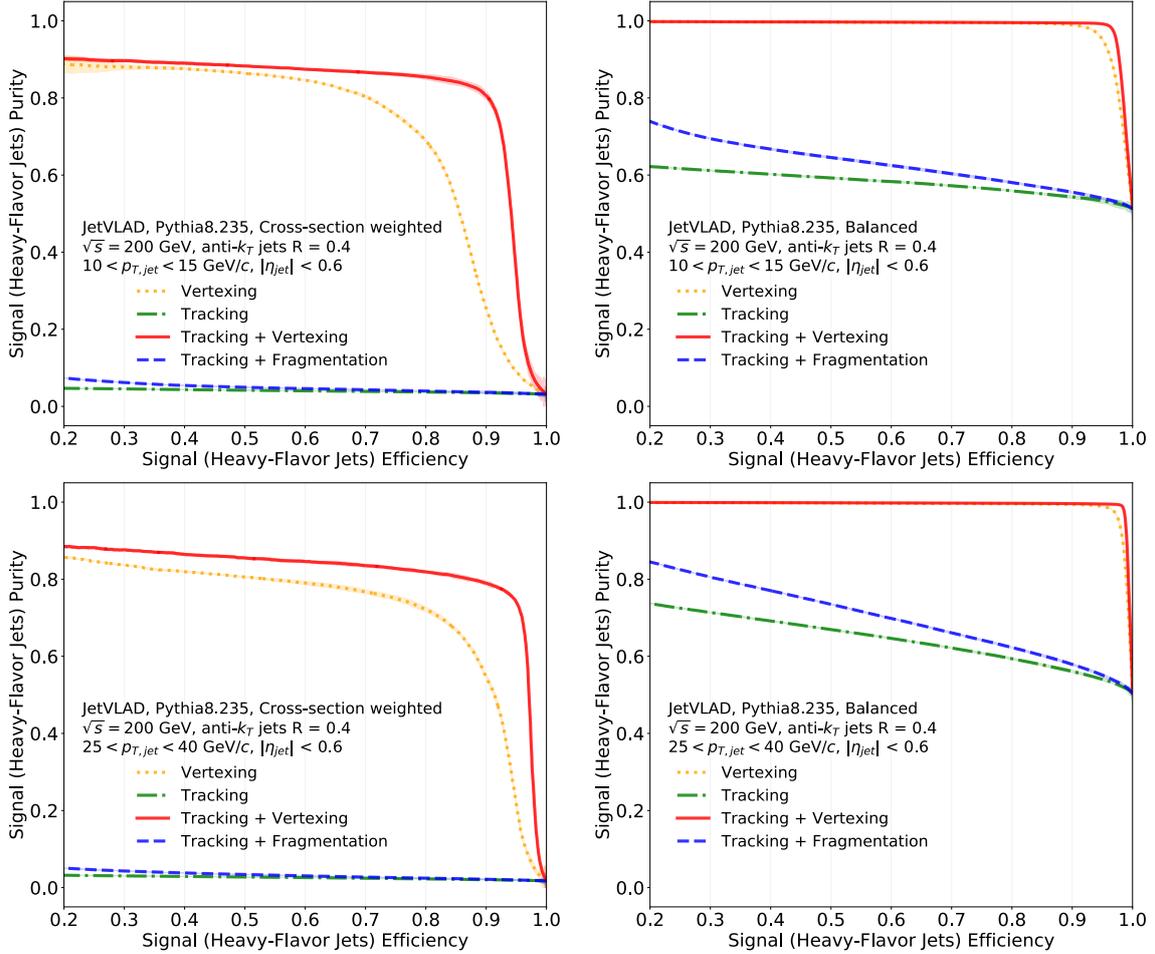


Figure 5. Purity vs efficiency curves shown for different inputs in the colored lines and for cross-section weighted (left) and balanced (right) samples. The top and bottom panels show jets with $10 < p_T < 15$ and $25 < p_T < 40$ GeV/c.

some of these degradation effects. In an upcoming publication we will focus more on the impact of in-time pileup such as the underlying event of heavy ion collisions and also increased pileup similar to what is observed at the LHC in order to stress-test the performance of the JetVLAD tagger.

7 Conclusions and outlook

We focused on identifying jets originating from heavy quarks such as b and c , as opposed to those that originate from lighter quarks and gluons. We introduced the JetVLAD model which takes charged jet constituents with varying quantities as input and aggregates to a descriptor vector which can then be used to compare different jet populations. We trained the model on light- and heavy-flavor jets in PYTHIA and compared the classification performance for different varieties of track inputs based on metrics such as purity and background rejection at various signal efficiencies. In our studies we identified a combination of track inputs such as the secondary vertex ($DCA_{x,y}$, DCA_z) and the kinematics of the tracks (p_T , η , ϕ) performed optimally leading to a signal purity

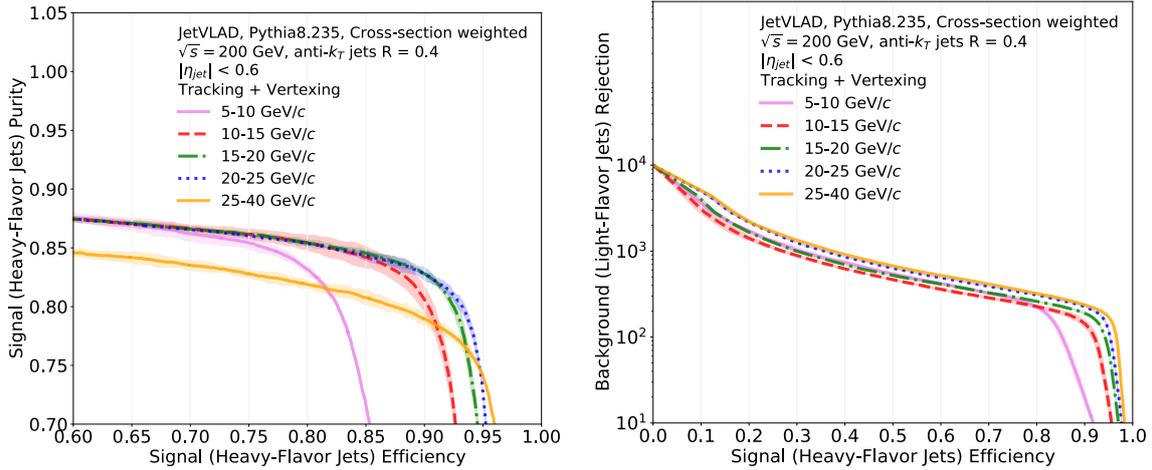


Figure 6. Purity (left) and background rejection (right) vs efficiency for tracking + vertexing input. Each curve corresponds to a different jet p_T selections.

Table 3. JetVLAD classification performance in purity and rejection for different jet p_T ranges for the cross-section weighted (balanced) datasets with two working points based on efficiencies of 81% and 50%, respectively. Note: the balanced samples include unrealistic production cross-section for heavy flavor jets which results in artificially increased classification performance.

Range in jet p_T [GeV/c]	Tagging Efficiency	Signal Purity	Background Rejection
[5 - 10]	80%	83% (99%)	223 (268)
	50%	88% (99%)	540 (579)
[10 - 15]	80%	85% (99%)	223 (230)
	50%	88% (99%)	476 (449)
[15 - 20]	80%	85% (99%)	259 (259)
	50%	88% (99%)	506 (476)
[20 - 25]	80%	85% (99%)	310 (336)
	50%	88% (99%)	624 (740)
[25 - 40]	80%	81% (99%)	322 (366)
	50%	85% (99%)	677 (740)

of 85% for an efficiency of 81% when we consider the cross-section weighted sample. We also studied the effect of the jet momenta and found that with increasing jet momenta, we increased the background rejection while the signal purity was relatively consistent at a given efficiency which can be utilized as an experimental working point. Our studies highlight the importance of a precision vertex detector for heavy flavor studies.

We demonstrated for the first time at RHIC energies the use of particle descriptors for identifying jet flavor. These low momenta jets at RHIC are particular important for studies of the

QGP transport properties since they are the ones that are expected to have the largest interactions with the medium. Aggregated particle descriptors offer a feature space where the inherent differences between light- and heavy-flavor jets are highlighted leading to significant improvements in classification performance.

Future explorations of this model include studying the effects of the heavy ion underlying event and also its extension to jet production at the LHC where one can explore BSM searches. While we have utilized the detector smearing based on the STAR Heavy-Flavor Tracker detector, this technique can be utilized to create a standard performance toolkit at sPHENIX [62] and in guiding detector design at the recently approved Electron Ion Collider (EIC). Studies at sPHENIX, whose vertex detector is designed for a high density heavy ion environment, are of immediate benefit for our model. Conversely, detectors at the EIC are more tuned towards precision measurements in a low density environment where one could take advantage of the JetVlad model architecture that effectively reduces the dimensionality of inputs to an aggregated vector, leading to variety of applications. In principle such a model can be utilized for studies related to particle flow, taking into account both charged and neutral candidates in experiment and also for mapping detector performance.

Acknowledgments

We thank Dennis Perepelitsa, Leticia Cunqueiro and Marta Verweij for comments on the draft. We also like to thank Jan Kreps and the CIIRC IT department for computing help. Antoine Miech’s LOUPE library [55] was used as the starting point for the NetVLAD layer. RKE thanks the Wayne State University computing grid for resources used in generating the training and testing datasets. RKE and JP are supported by US DOE grant No. DE-FG02-92ER40713. GP and JB are supported by project LTT18002 of the Ministry of Education, Youth and Sport of the Czech Republic. JB is supported by European Regional Development Fund-Project “Center of Advanced Applied Science” No. CZ.02.1.01/0.0/0.0/16 – 019/0000778. JS is partly supported by the European Regional Development Fund under the project IMPACT (reg. no. CZ.02.1.01/0.0/0.0/15_003/0000468) and by the French government under management of Agence Nationale de la Recherche as part of the “Investissements d’avenir” program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute).

A Fast simulation

Here we describe the fast simulation framework that was used in order to simulate detector response. Since a full detector simulation via GEANT is time consuming, one can capture major effects via parametrizing the response. To account for the tracking efficiency at STAR in proton-proton collisions, each charged and final state track is dropped with probability of 20% [63]. The TPC also produces a momentum smearing [63], which is modeled by

$$p_T = \mathcal{N}(p_T, 0.003 \cdot p_T^2). \quad (\text{A.1})$$

Regarding the finite vertex resolution, we apply a parametrization of the STAR heavy-flavor Tracker [65], by fitting the resolutions of DCA_{xy} and DCA_z dependent on the track momentum \mathbf{p}

$$DCA_{xy} = \mathcal{N}(DCA_{xy}, \sigma_{xy}(\mathbf{p})), \quad (\text{A.2})$$

$$DCA_z = \mathcal{N}(DCA_z, \sigma_z(\mathbf{p})). \quad (\text{A.3})$$

Post smearing, we apply selection criteria on the tracks similar to an experimental analysis

- Minimal smeared transverse momentum of the track is $p_T > 0.2 \text{ GeV}/c$.
- $|DCA_z| < 60 \text{ mm}$ and $|DCA_{xy}| < 20 \text{ mm}$

As mentioned before, this particular parametrization is based upon the existing STAR experiment. The upcoming sPHENIX experiment is designed for better tracking efficiency and includes specific detector prioritizing secondary vertex resolution such as the MVTX [64] which will further increase the classification performance.

References

- [1] G.F. Sterman and S. Weinberg, *Jets from quantum chromodynamics*, *Phys. Rev. Lett.* **39** (1977) 1436.
- [2] R.P. Feynman, R.D. Field and G.C. Fox, *A quantum chromodynamic approach for the large transverse momentum production of particles and jets*, *Phys. Rev. D* **18** (1978) 3320.
- [3] V.N. Gribov and L.N. Lipatov, *Deep inelastic ep scattering in perturbation theory*, *Sov. J. Nucl. Phys.* **15** (1972) 438 [*Yad. Fiz.* **15** (1972) 781].
- [4] L.N. Lipatov, *The parton model and perturbation theory*, *Sov. J. Nucl. Phys.* **20** (1975) 94 [*Yad. Fiz.* **20** (1974) 181].
- [5] Y.L. Dokshitzer, *Calculation of the structure functions for deep inelastic scattering and e^+e^- annihilation by perturbation theory in quantum chromodynamics.*, *Sov. Phys. JETP* **46** (1977) 641 [*Zh. Eksp. Teor. Fiz.* **73** (1977) 1216].
- [6] G. Altarelli and G. Parisi, *Asymptotic freedom in parton language*, *Nucl. Phys. B* **126** (1977) 298.
- [7] S. Catani and M.H. Seymour, *A general algorithm for calculating jet cross-sections in NLO QCD*, *Nucl. Phys. B* **485** (1997) 291 [Erratum *ibid.* **510** (1998) 503] [[hep-ph/9605323](#)].
- [8] B. Andersson, G. Gustafson, G. Ingelman and T. Sjöstrand, *Parton fragmentation and string dynamics*, *Phys. Rept.* **97** (1983) 31.
- [9] B.R. Webber, *A QCD model for jet fragmentation including soft gluon interference*, *Nucl. Phys. B* **238** (1984) 492.
- [10] M. Wobisch and T. Wengler, *Hadronization corrections to jet cross-sections in deep inelastic scattering*, in *Workshop on Monte Carlo generators for HERA physics (plenary starting meeting)*, (1998), pg. 270 [[hep-ph/9907280](#)].
- [11] J. Currie, E.W.N. Glover and J. Pires, *Next-to-next-to leading order QCD predictions for single jet inclusive production at the LHC*, *Phys. Rev. Lett.* **118** (2017) 072002 [[arXiv:1611.01460](#)].
- [12] Z.-B. Kang, F. Ringer and I. Vitev, *Inclusive production of small radius jets in heavy-ion collisions*, *Phys. Lett. B* **769** (2017) 242 [[arXiv:1701.05839](#)].

- [13] M. Dasgupta, F.A. Dreyer, K. Hamilton, P.F. Monni and G.P. Salam, *Logarithmic accuracy of parton showers: a fixed-order study*, *JHEP* **09** (2018) 033 [Erratum *ibid.* **03** (2020) 083] [[arXiv:1805.09327](#)].
- [14] M. Dasgupta, F.A. Dreyer, K. Hamilton, P.F. Monni, G.P. Salam and G. Soyez, *Parton showers beyond leading logarithmic accuracy*, *Phys. Rev. Lett.* **125** (2020) 052002 [[arXiv:2002.11114](#)].
- [15] D. Britzger, K. Rabbertz, D. Savoie, G. Sieber and M. Wobisch, *Determination of the strong coupling constant using inclusive jet cross section data from multiple experiments*, *Eur. Phys. J. C* **79** (2019) 68 [[arXiv:1712.00480](#)].
- [16] JET collaboration, *Extracting the jet transport coefficient from jet quenching in high-energy heavy-ion collisions*, *Phys. Rev. C* **90** (2014) 014909 [[arXiv:1312.5003](#)].
- [17] JETSCAPE collaboration, *Bayesian extraction of \hat{q} with multi-stage jet evolution approach*, *PoS(HardProbes2018)048* (2019).
- [18] M. Connors, C. Nattrass, R. Reed and S. Salur, *Jet measurements in heavy ion physics*, *Rev. Mod. Phys.* **90** (2018) 025005 [[arXiv:1705.01974](#)].
- [19] G.-Y. Qin and X.-N. Wang, *Jet quenching in high-energy heavy-ion collisions*, *Int. J. Mod. Phys. E* **24** (2015) 1530014 [[arXiv:1511.00790](#)].
- [20] J.-P. Blaizot and Y. Mehtar-Tani, *Jet structure in heavy ion collisions*, *Int. J. Mod. Phys. E* **24** (2015) 1530012 [[arXiv:1503.05958](#)].
- [21] Y.L. Dokshitzer, V.A. Khoze and S.I. Troian, *On specific QCD properties of heavy quark fragmentation ('dead cone')*, *J. Phys. G* **17** (1991) 1602.
- [22] DELPHI collaboration, *Study of b-quark mass effects in multijet topologies with the DELPHI detector at LEP*, *Eur. Phys. J. C* **55** (2008) 525 [[arXiv:0804.3883](#)].
- [23] F. Maltoni, M. Selvaggi and J. Thaler, *Exposing the dead cone effect with jet substructure techniques*, *Phys. Rev. D* **94** (2016) 054015 [[arXiv:1606.03449](#)].
- [24] L. Cunqueiro and M. Płoskoń, *Searching for the dead cone effects with iterative declustering of heavy-flavor jets*, *Phys. Rev. D* **99** (2019) 074027 [[arXiv:1812.00102](#)].
- [25] ALICE collaboration, *First direct observation of the dead-cone effect*, *Nucl. Phys. A* **1005** (2021) 121905 [[arXiv:2004.05968](#)].
- [26] CMS collaboration, *Evidence of b-jet quenching in PbPb collisions at $\sqrt{s_{NN}} = 2.76$ TeV*, *Phys. Rev. Lett.* **113** (2014) 132301 [Erratum *ibid.* **115** (2015) 029903] [[arXiv:1312.4198](#)].
- [27] CMS collaboration, *Comparing transverse momentum balance of b-jet pairs in pp and PbPb collisions at $\sqrt{s_{NN}} = 5.02$ TeV*, *JHEP* **03** (2018) 181 [[arXiv:1802.00707](#)].
- [28] Z.-B. Kang, J. Reiten, I. Vitev and B. Yoon, *Light and heavy flavor dijet production and dijet mass modification in heavy ion collisions*, *Phys. Rev. D* **99** (2019) 034006 [[arXiv:1810.10007](#)].
- [29] H.T. Li and I. Vitev, *Jet splitting function in the vacuum and QCD medium*, *PoS(HardProbes2018)077* (2018) [[arXiv:1812.03348](#)].
- [30] ATLAS collaboration, *Performance of b-jet identification in the ATLAS experiment*, *2016 JINST* **11** P04008 [[arXiv:1512.01094](#)].
- [31] CMS collaboration, *Identification of b-quark jets with the CMS experiment*, *2013 JINST* **8** P04013 [[arXiv:1211.4462](#)].

- [32] P. Ilten, N.L. Rodd, J. Thaler and M. Williams, *Disentangling heavy flavor at colliders*, *Phys. Rev. D* **96** (2017) 054019 [[arXiv:1702.02947](#)].
- [33] M. Voutilainen, *Heavy quark jets at the LHC*, *Int. J. Mod. Phys. A* **30** (2015) 1546008 [[arXiv:1509.05026](#)].
- [34] STAR collaboration, *Performance of heavy-flavor tagged jet identification in STAR*, in *Quark Matter 2018*, [contribution ID: 375](#), Venice, Italy (2018).
- [35] K. Jung, *Flavors in the soup: an overview of heavy-flavored jet energy loss at CMS*, Ph.D. thesis, Illinois U., Chicago, IL, U.S.A. (2016).
- [36] D. Guest, J. Collado, P. Baldi, S.-C. Hsu, G. Urban and D. Whiteson, *Jet flavor classification in high-energy physics with deep neural networks*, *Phys. Rev. D* **94** (2016) 112002 [[arXiv:1607.08633](#)].
- [37] CMS collaboration, *A deep neural network for simultaneous estimation of b jet energy and resolution*, *Comput. Softw. Big Sci.* **4** (2020) 10 [[arXiv:1912.06046](#)].
- [38] ATLAS collaboration, *Identification of jets containing b -hadrons with recurrent neural networks at the ATLAS experiment*, Tech. Rep. [ATL-PHYS-PUB-2017-003](#), CERN, Geneva, Switzerland (2017).
- [39] ATLAS collaboration, *Optimisation and performance studies of the ATLAS b -tagging algorithms for the 2017–18 LHC run*, Tech. Rep. [ATL-PHYS-PUB-2017-013](#), CERN, Geneva, Switzerland (2017).
- [40] T. Sjöstrand et al., *An introduction to PYTHIA 8.2*, *Comput. Phys. Commun.* **191** (2015) 159 [[arXiv:1410.3012](#)].
- [41] ALICE collaboration, *Groomed jet substructure measurements of charm jets tagged with D^0 mesons in pp collisions at $\sqrt{s} = 13$ TeV*, Tech. Rep. [ALICE-PUBLIC-2020-002](#), CERN, Geneva, Switzerland (2020).
- [42] STAR collaboration, *STAR detector overview*, *Nucl. Instrum. Meth. A* **499** (2003) 624.
- [43] M. Anderson et al., *The STAR time projection chamber: a unique tool for studying high multiplicity events at RHIC*, *Nucl. Instrum. Meth. A* **499** (2003) 659 [[nuc1-ex/0301015](#)].
- [44] STAR collaboration, *STAR heavy flavor tracker*, *Nucl. Phys. A* **931** (2014) 1141.
- [45] M. Cacciari, G.P. Salam and G. Soyez, *The anti- k_t jet clustering algorithm*, *JHEP* **04** (2008) 063 [[arXiv:0802.1189](#)].
- [46] M. Cacciari, G.P. Salam and G. Soyez, *FastJet user manual*, *Eur. Phys. J. C* **72** (2012) 1896 [[arXiv:1111.6097](#)].
- [47] Z.-B. Kang, K. Lee and F. Ringer, *Jet angularity measurements for single inclusive jet production*, *JHEP* **04** (2018) 110 [[arXiv:1801.00790](#)].
- [48] STAR collaboration, *Measurement of inclusive charged-particle jet production in Au+Au collisions at $\sqrt{s_{NN}} = 200$ GeV*, *Phys. Rev. C* **102** (2020) 054913 [[arXiv:2006.00582](#)].
- [49] STAR collaboration, *Measurement of groomed jet substructure observables in $p+p$ collisions at $\sqrt{s} = 200$ GeV with STAR*, *Phys. Lett. B* **811** (2020) 135846 [[arXiv:2003.02114](#)].
- [50] A.J. Larkoski, I. Moult and B. Nachman, *Jet substructure at the Large Hadron Collider: a review of recent advances in theory and machine learning*, *Phys. Rept.* **841** (2020) 1 [[arXiv:1709.04464](#)].
- [51] S. Catani, Y.L. Dokshitzer, M.H. Seymour and B.R. Webber, *Longitudinally invariant K_t clustering algorithms for hadron hadron collisions*, *Nucl. Phys. B* **406** (1993) 187.

- [52] Y.L. Dokshitzer, G.D. Leder, S. Moretti and B.R. Webber, *Better jet clustering algorithms*, *JHEP* **08** (1997) 001 [[hep-ph/9707323](#)].
- [53] A. Andreassen, I. Feige, C. Frye and M.D. Schwartz, *JUNIPR: a framework for unsupervised machine learning in particle physics*, *Eur. Phys. J. C* **79** (2019) 102 [[arXiv:1804.09720](#)].
- [54] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla and J. Sivic, *NetVLAD: CNN architecture for weakly supervised place recognition*, *IEEE Trans. Pattern Anal. Machine Intell.* **40** (2018) 1437 [[arXiv:1511.07247](#)].
- [55] A. Miech, I. Laptev and J. Sivic, *Learnable pooling with context gating for video classification*, [arXiv:1706.06905](#).
- [56] K. He, X. Zhang, S. Ren and J. Sun, *Deep residual learning for image recognition*, [arXiv:1512.03385](#).
- [57] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, *Dropout: a simple way to prevent neural networks from overfitting*, *J. Mach. Learn. Res.* **15** (2014) 1929.
- [58] I. Loshchilov and F. Hutter, *SGDR: Stochastic Gradient Descent with Restarts*, [arXiv:1608.03983](#).
- [59] M. Segù, A. Loquercio and D. Scaramuzza, *A general framework for uncertainty estimation in deep learning*, [arXiv:1907.06890](#).
- [60] C. Englert, P. Galler, P. Harris and M. Spannowsky, *Machine learning uncertainties with adversarial neural networks*, *Eur. Phys. J. C* **79** (2019) 4 [[arXiv:1807.08763](#)].
- [61] G. Kasieczka, M. Luchmann, F. Otterpohl and T. Plehn, *Per-object systematics using deep-learned calibration*, *SciPost Phys.* **9** (2020) 089 [[arXiv:2003.11099](#)].
- [62] PHENIX collaboration, *An upgrade proposal from the PHENIX collaboration*, [arXiv:1501.06197](#).
- [63] J. Rusnak, *Jet reconstruction in Au+Au collisions at RHIC*, Ph.D. thesis, Nuclear Physics Institute, The Czech Academy of Sciences, Rez, Czech Republic (2017).
- [64] *A monolithic active pixel sensor detector for the sPHENIX experiment*, <https://p25ext.lanl.gov/maps/mvtx/Proposals/sPHENIX-MVTX-Preproposal-022017-final.pdf>.
- [65] J. Schambach et al., *The STAR Heavy Flavor Tracker (HFT)*, in *20th International Conference on Particles and Nuclei*, (2014), pg. 659 [*Proc. PANIC* (2014) 660].

IDENTIFYING HEAVY-FLAVOR JETS USING VECTORS OF LOCALLY AGGREGATED DESCRIPTORS

G. Ponimatkin, ponimatkin@ujf.cas.cz, Nuclear Physics Institute of the Czech Academy of Sciences, Prague, Czech republic

INTRODUCTION

The heavy-flavor jets which arise from hard scattered heavy quarks play an important role in many physics processes at collider experiments. Thus, the ability to identify such jets with high precision is crucial to many measurements, for example in studies of boosted objects at the LHC or studies of mass dependence of energy loss. At RHIC energies successful identification of heavy-flavor jets down to the low transverse momentum (p_T) can help with searches for the dead-cone effect [1] via Lund plane formalism [2, 3] and its modification via quark-gluon plasma medium [4] created in heavy-ion collisions.

Machine learning is an established way for solving such classification problems. The previous research on jet flavor identification utilized jet images [5] (for W/Z boson initiated jets) or p_T/DCA (distance of closest approach to the primary vertex) ordered sequences of particles [6] (for heavy flavor jet identification). The p_T/DCA ordering of particles in a jet is not physical and hence other methods are needed. One such method, called NetVLAD [7] is an efficient way for solving computer vision tasks that are using sets of vectors as an input.

TRAINING DATASET

Training data for pp collisions at $\sqrt{s} = 200$ GeV is simulated with the PYTHIA8 [8, 9] event generator at the center of mass energy $\sqrt{s} = 200$ GeV. We generate 2 datasets that contain light jets (those initialized by uds -quarks and gluons) and heavy-flavor jets (those coming from the fragmentation of c and b quarks). First dataset respects the realistic jet cross-section ratios and the second one with balanced number of jets represents an idealized benchmark case. In order to bring in the effects of the finite detector resolution, the transverse momentum and vertexing information of the particles are smeared using a Gaussian distribution via $p_T = \mathcal{N}(p_T, \sigma(p_T))$ [10], where $\sigma(p_T)$ and $DCA = \mathcal{N}(DCA, \sigma(\mathbf{p}))$ [11] with $\sigma(p_T)$ being the transverse momentum resolution and $\sigma(\mathbf{p})$ is vertex tracker resolution. After smearing, only the charged particles (protons, pions and kaons) with $0.2 < p_T < 30$ GeV/c and pseudorapidity $|\eta| < 1.0$ are accepted, since only charged particles leave hits in the vertex tracker.

JETVLAD MODEL AND ITS PERFORMANCE

We introduce the JetVLAD model, which takes a set of tracks, that are produced by the anti- k_T [14] jet reconstruction algorithm within the FastJet package [12, 13] and predicts if the jet originates from a light or heavy-flavor parton. Initially 4 combinations of the input variables were considered - track-

ing (p_T, η, φ), vertexing (DCA_{xy}, DCA_z), tracking and vertexing ($p_T, \eta, \varphi, DCA_{xy}, DCA_z$) and finally tracking and fragmentation ($p_T, \eta, \varphi, z, \Delta R, z(\Delta R)^2$). Here φ represent azimuthal angle of the track, DCA_{xy} and DCA_z are xy and z projections of the DCA , ΔR is the distance between track and jet axis and z is a total jet momentum fraction carried away by the track.

The tracking and vertexing inputs were chosen to yield optimal classification performance. The resulting model possesses 111608 of free parameters and only two hyper-parameters (number of layers in the model and number of clusters in the NetVLAD layer), which makes running this model computationally very efficient. The evaluation is done using standard metrics, namely the efficiency:

$$\text{Eff} = \frac{\text{TP}}{\text{P}}, \quad (1)$$

i.e. ratio of true-positives (correctly identified heavy-flavor jets) over positives (total number of heavy-flavor jets in the testing sample). This metric shows percentage of heavy-flavor jets algorithm can retrieve from the sample. Next metric is purity, given by

$$\text{Pur} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (2)$$

i.e. it is ratio of true-positives over true-positives and false-positives (light flavor jets that were identified as heavy-flavor) in the accepted sample. As the name suggest, this metric shows the background contamination in the selected sample. The last metric is rejection

$$\text{Rej} = \frac{\text{N}}{\text{FP}}, \quad (3)$$

where N is a total number of negatives in the sample (i.e light flavor jets) and FP is a number of false positive jets. This metric tells how many background jets are rejected per one false positive jet accepted. The results for selected working points and jet p_T ranges can be seen in the Table . Efficiency and rejection plots that were used for optimal input combination can be seen on Figure 1. The efficiency and purity plots for jets across different p_T ranges can be seen on Figure 2.

TAB. 1. JetVLAD classification performance in purity and rejection for different jet p_T ranges for the cross-section weighted (balanced) datasets with two working points based on efficiencies of 80% and 50%, respectively.

Range in jet p_T [GeV/c]	Tagging Efficiency	Signal Purity	Background Rejection
[5 - 10]	80%	83% (99%)	223 (268)
	50%	88% (99%)	540 (579)
[25 - 40]	80%	81% (99%)	322 (366)
	50%	85% (99%)	677 (740)

Figure 1 shows the background rejection as a function of the classification efficiency for different inputs.

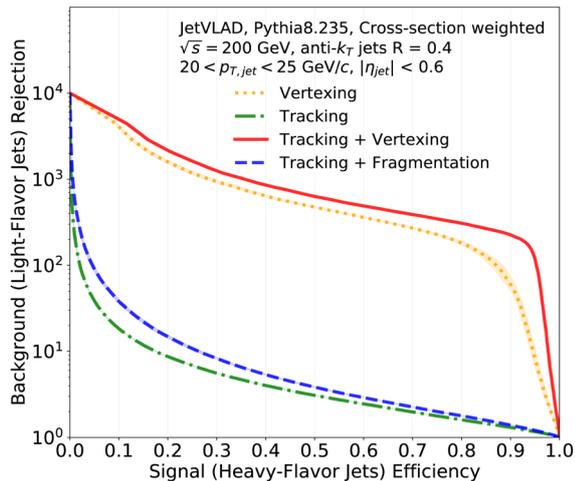


Fig. 1. Rejection vs efficiency plots for different input variables. The combination of tracking and vertexing information provides the best performance.

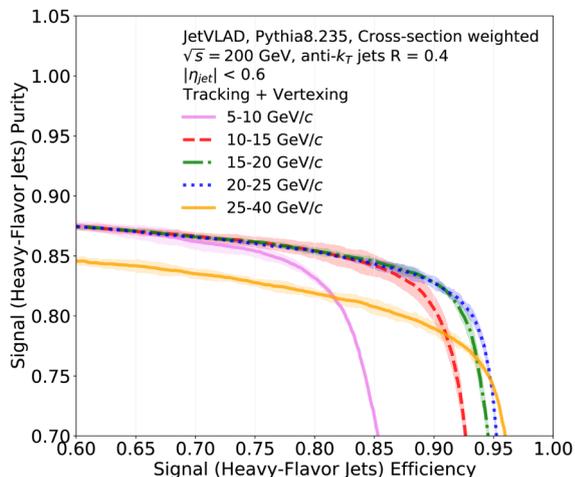


Fig. 2. Purity vs efficiency plots for different jet p_T ranges.

One can see that tracking and vertexing provides the best performance.

It is interesting to note, that vertexing alone provides a good model performance, but adding tracking information increases overall performance. Figure 2 shows that the model performance is changing slowly for different p_T ranges. The drop in the purity for the highest momentum range is due to the difference in jet kinematics, since at those momenta heavy-flavor jets are starting to resemble light jets.

CONCLUSIONS

We propose a novel jet tagging method suitable for RHIC energies which utilizes a set of tracks as an input making this tagger directly applicable to experiments for existing vertexing detectors and jet algorithms. The resulting method is computationally efficient and has a small number of hyper-parameters. The proposed method achieves 83% purity, background rejection rate of more than 200 and efficiency of 80% at lowest jet momenta. Such a performance enables

indentation of heavy-flavor jets at RHIC energies and provides the opportunity for new flavor tagged measurements.

ACKNOWLEDGMENT: This work is supported by the project LTT18002 of the Ministry of Education, Youth and Sport of the Czech Republic.

REFERENCES

1. Yu. L. Dokshitzer, D. E. Kharzeev, Phys. Lett. **B519**,199 (2001).
2. F. A. Dreyer, G. P. Salam, G. Soyez, JHEP **12**, 064 (2018).
3. L. Cunqueiro, M. Ploskon, Phys. Rev. **D 99**, 074027 (2019).
4. H. A. Andrews et al., J. Phys. **G 47**, 065102 (2020).
5. L. Oliveira, M. Kagan, L. Mackey, B. Nachman, A. Schwartzman, JHEP **07**, 069 (2016).
6. D. Guest, J. Collado, P. Baldi, S.-C. Hsu, G. Urban, D. Whiteson, Phys. Rev. **D 94**, 112002 (2016).
7. R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, J. Sivic, IEEE Transactions on Pattern Analysis and Machine Intelligence **40**, 1437 (2018).
8. T. Sjostrand, S. Mrenna, P. Skands, JHEP **05**, 026 (2006).
9. T. Sjostrand et al., Comput. Phys. Commun. **191** 159 (2015).
10. J. Rusnak: Jet Reconstruction in Au+Au collisions at RHIC (PhD thesis, Nuclear Physics Institute, The Czech Academy of Sciences - 2017).
11. STAR Collaboration, Phys. Rev. **C 99**, 034908 (2019).
12. M. Cacciari, G.P. Salam, G. Soyez, Eur. Phys. J. **C72**, 1896 (2012).
13. M. Cacciari, G.P. Salam, Phys. Lett. **B 641**, 57 (2006).
14. M. Cacciari, G.P. Salam, G. Soyez, JHEP **0804**, 063 (2008).