Prepared for submission to JHEP

Machine Learning Techniques for Tagging Heavy Flavor Jets at RHIC

George Halal,[a] **Helen Caines,**[b] **Rosi Reed,**[a] **and Saehanseul Oh**[b,c]

[a] *Department of Physics, Lehigh University, Bethlehem, PA 18015*
[b] *Department of Physics, Yale University, New Haven, CT 06511*
[c] *Brookhaven National Laboratory, Upton, NY 11973*

E-mail: georgech@stanford.edu, helen.caines@yale.edu, rosijreed@lehigh.edu, saehanseul.oh@yale.edu

Abstract: The properties of the Quark Gluon Plasma (QGP), a hot and dense medium made up of deconfined quarks and gluons (partons), can be studied through ultrarelativistic heavy-ion collisions. In the early stages of the collisions, high energy partons are created, which fragment into collimated sprays of hadrons, called jets. Jets are used to probe the entire evolution of the QGP that they traverse. Classifying jets based on the flavor of the parton that initiated them as heavy or light is a fundamental tool for studying the properties of the QGP as different flavors interact differently with the medium. Jets resulting from heavy ion collisions are compared to those resulting from proton-proton collisions to study their modification in the medium. Therefore, as a first step, we use jets resulting from simulated proton-proton events for the identifying their flavors, which could be extended to identifying jets resulting from heavy-ion events once heavy flavor features are added to heavy-ion event generators. We utilize different deep learning techniques and employ different strategies to minimize the misidentification probability while maintaining the efficiency of tagging heavy flavor jets at RHIC. Similar analysis has been done for LHC experiments but never at RHIC energies ($\sqrt{s} = 200$ GeV), where the ratio of the production rates of heavy flavor jets to light flavor jets is very low. In this paper, we compare and contrast the performances of the different models we have developed for tagging heavy flavor jets at RHIC energies.

Contents

1 Introduction

The only two currently operating facilities that accelerate heavy ions, such as gold and lead, to ultra-relativistic speeds and collide them are the Relativistic Heavy Ion Collider (RHIC) and the Large Hadron Collider (LHC). When heavy ions collide with each other, the quarks and gluons (partons) that are typically confined within the protons and neutrons (hadrons) that make up the nuclei become deconfined in a state called the Quark-Gluon Plasma (QGP). Studying the properties of the QGP, a hot and dense medium which is believed to have filled the universe microseconds after the Big Bang, is one of the main goals of RHIC as it helps us complete the phase diagram of Quantum Chromodynamics (QCD), the theory of strong interactions.

The highly energetic (hard) partons that scatter during the collision initiate collimated sprays of particles, called jets. These jets form before the formation of the QGP, allowing them to traverse the medium and get modified by it in a process called jet quenching. We can measure the final state particles of these jets (constituents) with detectors located around the collision and infer some of the QGP properties.

The identification (tagging) of the flavor of the parton that initiated each of the jets is of high significance for probing the QGP properties. Jets initiated by bottom (b) or charm (c) quarks, called heavy-flavor jets, are predicted to interact differently with the QGP than those initiated by up, down, or strange quarks or gluons ($udsg$), called light-flavor jets. Heavy-flavor jets are predicted to undergo different levels of collisional and radiative energy losses [8] with the QGP.

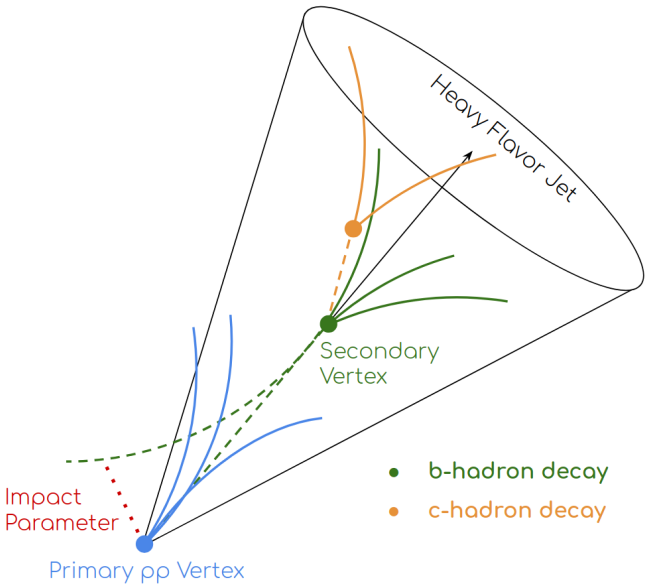

Figure 1. Illustration of a heavy-flavor jet with tracks originating from secondary and tertiary vertices due to b-hadron and c-hadron decays, respectively. It shows how these tracks have large impact parameters due to their displaced vertices. This specific jet would be tagged as bottom, due to the presence of a b-hadron, which is the heaviest of the particles it contains.

The main difference between heavy and light flavor jets is the distance between the position of the initial hard scattering (the primary vertex) and that of the subsequent decay of the jet's progenitor particle (the secondary or tertiary vertex). These are shown in Figure 1. b-hadrons initiated at the primary vertex decay the farthest away from it, followed by c-hadrons. While some b-hadrons later decay to c-hadrons, as shown in the Figure, jets containing both b and c hadrons are classified as b-jets. Therefore, b-jet constituents tend to originate from vertices that are the farthest away from the primary vertex.

We use machine learning techniques, a form of artificial intelligence, to significantly enhance the tagging of heavy-flavor jets at RHIC energies with a high computational efficiency. Similar techniques have been used to tag heavy-flavor jets at other experiments [10], but this is the first attempt to accomplish this at RHIC center of mass energy of $\sqrt{s} = 200$ GeV. At this energy, the ratio of light to heavy flavor jets produced varies significantly from that at LHC energies, increasing the difficulty of tagging heavy flavor jets.

In this paper, we employ different types of deep neural network models to perform the classification and compare their performances. To quantify the performance of each model, we use Receiver Operating Characteristic (ROC) curves, which plot the false positive rate (misidentification probability) as a function of the true positive rate (efficiency) of the tagging. We choose a satisfactory level of efficiency (70% in our case) and compare the misidentification probabilities of each of the models at that efficiency.

Heavy flavor features in heavy ion (AA) event generators are still under development. Therefore, we use a proton-proton (pp) event generator for the purposes of this paper. To study jet quenching, physicists compare jets in pp collisions to those in AA collisions.

Therefore, being able to tag jet flavors in pp collisions is of great significance. Moreover, AA collisions are messier than pp collisions in general; if we can't tag heavy flavor jets in pp, it is much more difficult to do so in AA. Therefore, developing a technique for tagging heavy flavor jets in pp at RHIC is an important first step towards tagging heavy flavor jets in AA as well.

2 Dataset Preparation

We generate the events using Monte Carlo simulations of proton-proton (pp) collisions at $\sqrt{s} = 200$ GeV using PYTHIA 8.235 [1]. We then cluster the final-state non-neutrino particles that are within a pseudorapidity ($\eta \equiv -\ln[\tan(\theta/2)]$) of $|\eta| < 1.0$ into jets, where θ is the angle away from the z-axis which is along the beam pipe. The anti-k_t algorithm [2] in FastJet 3.3.1 [3] is used, which clusters the particles into cones based on their momenta in the transverse plane (xy-plane), p_T. This η range is chosen based on the acceptance of the STAR and sPHENIX experiments at RHIC. We use a resolution parameter of $R = 0.4$, where $R \equiv \sqrt{\phi^2 + \eta^2}$, and ϕ is the azimuthal angle. Jets within $|\eta^{\text{jet}}| < 1.0$ are analyzed in three separate p_T bins: 10 GeV/c < p_T^{jet} < 20 GeV/c, 20 GeV/c < p_T^{jet} < 30 GeV/c, and $p_T^{\text{jet}} > 30$ GeV/c.

We employ two methods for identifying the progenitor particle of a given jet at the truth level.

Method 1: Start from the final-state jet constituents, and trace back through their decay histories up to the original (heaviest) hadron formed in the jet. Match the jet's flavor to that hadron's flavor.

Method 2: Start from the daughters of the initially hard-scattered partons, and look through their decay cascades. Match the jet's flavor to that of the heaviest of the daughter partons satisfying the condition, $\Delta R(\text{jet, parton}) < 0.4$, where $\Delta R(\text{jet, parton}) \equiv \sqrt{(\phi^{\text{jet}} - \phi^{\text{parton}})^2 + (\eta^{\text{jet}} - \eta^{\text{parton}})^2}$.

We only consider the jets for which both of these methods agree. We use the flavor of the jets tagged using these methods as the output that we train the neural network models to produce.

3 Multilayer Perceptron Model

We first try to tag the jet flavors using a traditional fully-connected feedforward neural network model, also called a Multilayer Perceptron (MLP). We start by modeling each final-state particle's track as a helix in a 0.5 T magnetic field. Then, we use several properties of these tracks as discriminators to be included as inputs to the neural network for distinguishing between heavy and light flavor jets. These are summarized in the following subsection.

Relative Production Rates	b jets	c jets	light jets
10 GeV/c < p_T^{jet} < 20 GeV/c	1	5.394	176.5
20 GeV/c < p_T^{jet} < 30 GeV/c	1	3.464	146.1
p_T^{jet} > 30 GeV/c	1	3.440	196.3

Table 1. Table showing the relative production rates for jets of different flavors for different p_T^{jet} ranges in the range $|\eta| < 1.0$ as calculated from PYTHIA with a pTHatMin 0.5 GeV/c less than lower threshold of each p_T^{jet} range.

3.1 Discriminators

IP1, IP2, IP3 We trace back each particle's track to find its distance of closest approach to the primary vertex, also called the impact parameter (IP). An example of this is shown in red in Figure 1. The sign of the IP is the same as the sign of the dot product between a vector along the jet axis and a vector pointing from the primary vertex to the point on the particle track that is closest to the primary vertex. We sort the IP values of the jet constituents in descending order and use the largest three as discriminators.

DV1, DV2 We also sort the distances between the displaced vertices (DVs) of the final-state jet constituents and the primary vertex in descending order and use the largest two as discriminators.

$N_{constituents}$, N_{DV} Other useful discriminators are the total number of constituents within each jet and the number of jet constituents with DVs.

Lepton Constituents Finally, the presence of an electron or a muon within a jet can signify that the jet originated from a heavy flavor hadron as those tend to undergo semi-leptonic decays.

3.2 Architecture

We use the nine discriminators discussed above as inputs to a multilayer perceptron (MLP) consisting of two hidden dense layers with 125 neurons each and a final output layer with three neurons. The three neurons stand for b-jets, c-jets, and $udsg$-jets. The inputs are scaled in order to have similar ranges and thus be viewed on an equal footing by the neural network. A dropout layer is added after every hidden layer, which randomly drops 25% of the nodes in that layer during the training to prevent the model from overtraining itself on the training dataset.

This model is implemented in Keras [4] with the TensorFlow [5] back-end. The model checks the validation loss value after every epoch and stops the training process after the validation loss hasn't decreased in 20 epochs. This is called EarlyStopping and is used instead of training the model for a specified number of epochs. The Rectified Linear Unit (ReLU) activation function [6] is utilized in each layer, except for the output layer which is softmaxed. The model uses the Adam optimizer [7] and the categorical cross entropy loss function during the training.

Misidentification probability with 70% efficiency	b jets	c jets	light jets
10 GeV/c < p_T^{jet} < 20 GeV/c	68.6%	84.6%	0.133%
20 GeV/c < p_T^{jet} < 30 GeV/c	43.0%	85.5%	0.0848%
p_T^{jet} > 30 GeV/c	46.6%	92.1%	0.154%

Table 2. Table showing the misidentification probability of tagging jets of different flavors for different p_T^{jet} ranges with a 70% efficiency using the fully-connected feedforward neural network model.

3.3 Performance

For each of the three flavor categories (light, bottom, and charm) and each of the three p_T^{jet} ranges (10-20, 20-30, >30), we use 200,000 jets for training and 100,000 jets for testing this model. We train on equal amounts of jets for each flavor in order to not introduce any biases in the network's prediction. However, to make the result more realistic, one needs to consider amounts of jets corresponding to the relative production rates of bottom, charm, and light jets at RHIC when colliding protons at $\sqrt{s} = 200$ GeV.

To determine those rates, we run PYTHIA at this energy and count the number of jets produced for each flavor in the pseudorapidity range $|\eta| < 1.0$. For the 10 GeV/c < p_T^{jet} < 20 GeV/c, 20 GeV/c < p_T^{jet} < 30 GeV/c, and p_T^{jet} > 30 GeV/c ranges, we use pTHatMin 9.5 GeV/c, 19.5 GeV/c, and 29.5 GeV/c in PYTHIA, respectively. The resulting rates are summarized in Table 1.

When the predictions are scaled according to the relative production rates listed in Table 1, the resulting ROC curve, which is explained in Section 1, is shown in Figure 2. For a 70% efficiency, the misidentification probabilities of tagging bottom, charm, and light jets are listed in Table 2. Note the difference in the misidentification probabilities of bottom and charm jets to those of light jets. This is due to the high imbalance in the relative production rates.

3.4 Lepton Selection

To improve the model's performance, we thought of a solution that would balance out the relative production rates, since the relative weighting is what exacerbates the performance. We only consider jets that have at least one muon constituent with a $p_T^{\mu} > 1.3$ GeV/c. Due to the semi-leptonic decay channel of heavy flavor hadrons, we expect that more heavy-flavor jets than light-flavor jets will have this requirement, therefore balancing out the jet samples.

Relative Production Rates	b jets	c jets	light jets
10 GeV/c < p_T^{jet} < 20 GeV/c	4.25	9.17	1
20 GeV/c < p_T^{jet} < 30 GeV/c	3.54	3.65	1
p_T^{jet} > 30 GeV/c	1.86	1.64	1

Table 3. Table showing the relative production rates for jets of different flavors with a high p_T muon for different p_T^{jet} ranges in the range $|\eta| < 1.0$ as calculated from PYTHIA with a pTHatMin 0.5 GeV/c less than lower threshold of each p_T^{jet} range.

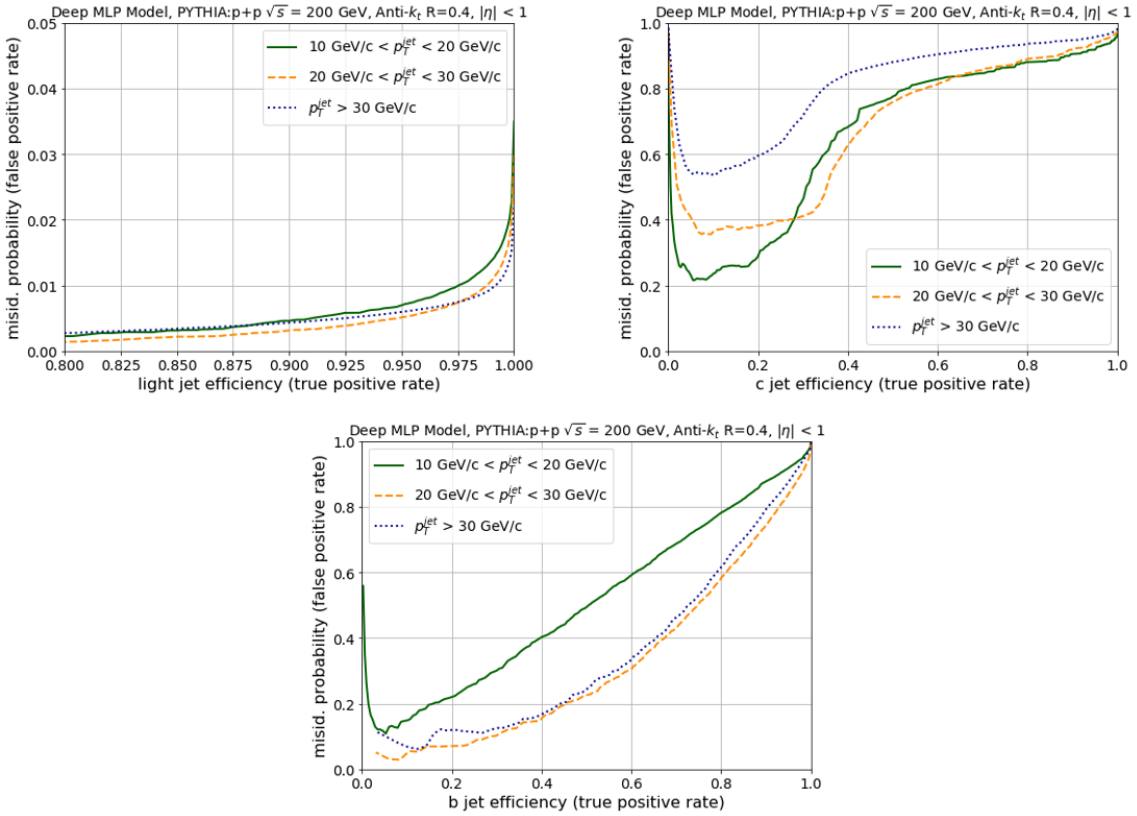

Figure 2. ROC curves showing the misidentification probability of bottom, charm, and light jet samples, weighted according to Table 1, as a function of the efficiency of tagging them using the MLP model.

The approximate relative jet production rates with this requirement for bottom, charm, and light jets can be found in Table 3. These are calculated as explained in Section 3.3. For the same motivation, we repeat the analysis for jets with at least one electron constituent that has a $p_T^e > 5.0$ GeV/c. The approximate relative jet production rates with this requirement for bottom, charm, and light jets can be found in Table 4. The p_T threshold values are chosen to comply with the values of the trigger thresholds of the STAR experiment's Muon Telescope Detector (MTD) and high tower triggers, respectively.

The same model is used for these jet samples with seven instead of nine inputs, where

Relative Production Rates	b jets	c jets	light jets
10 GeV/c < p_T^{jet} < 20 GeV/c	1.08	1	1.12
20 GeV/c < p_T^{jet} < 30 GeV/c	1.18	1	3.31
p_T^{jet} > 30 GeV/c	1.20	1	9.47

Table 4. Table showing the relative production rates for jets of different flavors with a high p_T electron for different p_T^{jet} ranges in the range $|\eta| < 1.0$ as calculated from PYTHIA with a pTHatMin 0.5 GeV/c less than lower threshold of each p_T^{jet} range.

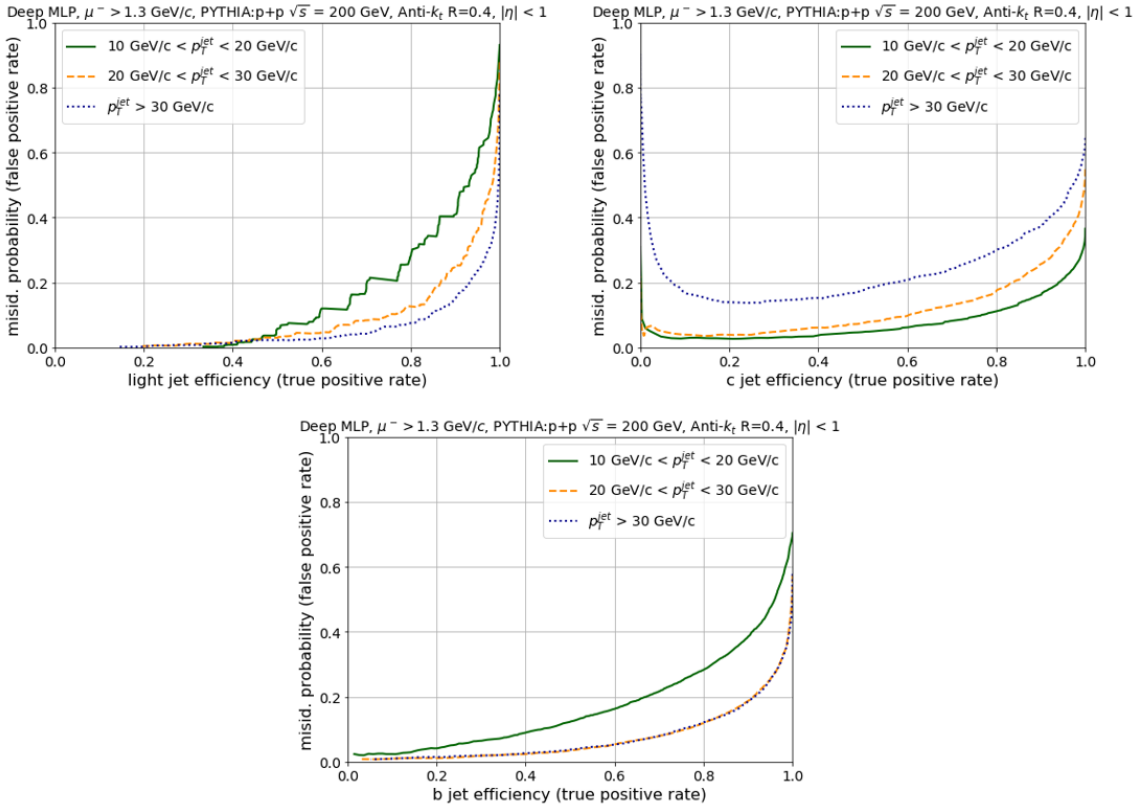

Figure 3. ROC curves showing the misidentification probability of bottom, charm, and light jet samples with at least one muon constituent with $p_T > 1.3$ GeV/c, weighted according to Table 4, as a function of the efficiency of tagging them using the MLP model.

the lepton constituent inputs are removed. The model's performance when trained on jets with the muon and electron requirements is shown in the ROC curves found in Figures 3 and 4, respectively. For a 70% efficiency, the misidentification probabilities of tagging bottom, charm, and light jets can be found in Table 5 for jets with the muon requirement and in Table 6 for jets with the electron requirement. Note the significant improvement in performance of the model with this dataset compared with the dataset that lacks these requirements.

However, by making these selections, we may be biasing the jet samples used to study the QGP properties since the jet substructure is important to consider when studying jet

Misidentification probability with 70% efficiency	b jets	c jets	light jets
10 GeV/c < p_T^{jet} < 20 GeV/c	21.7%	8.23%	17.8%
20 GeV/c < p_T^{jet} < 30 GeV/c	7.86%	13.0%	8.34%
p_T^{jet} > 30 GeV/c	7.96%	24.2%	4.21%

Table 5. Table showing the misidentification probability of tagging jets of different flavors with a high p_T muon for different p_T^{jet} ranges with a 70% efficiency using the fully-connected feedforward neural network model.

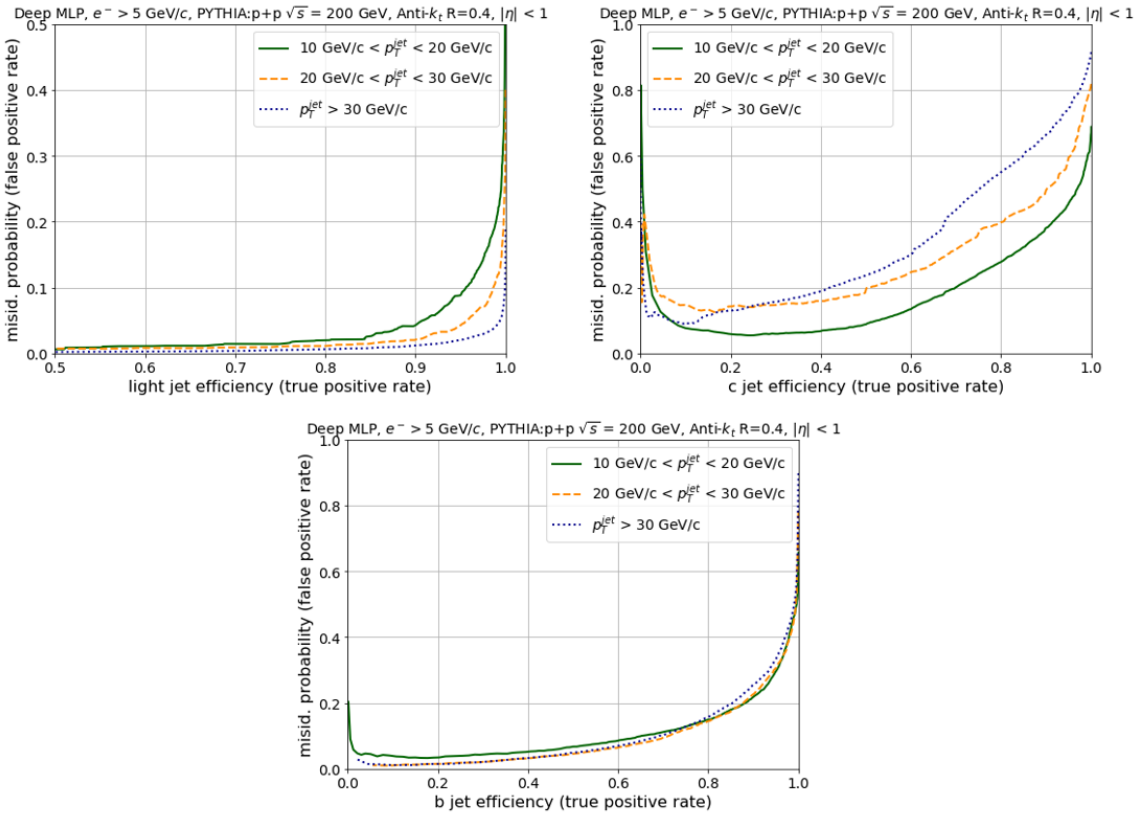

Figure 4. ROC curves showing the misidentification probability of bottom, charm, and light jet samples with at least one electron constituent with $p_T > 5$ GeV/c, weighted according to Table 4, as a function of the efficiency of tagging them using the MLP model.

quenching. For this reason, we propose a new deep learning model in the following section that doesn't require these selections.

4 Concatenated Model

We propose a concatenated deep learning model that makes use of Long Short-Term Memory (LSTM) [9] layers, a type of Recurrent Neural Networks (RNNs). Among other capabilities, RNNs can process an arbitrarily long sequence of inputs and output a fixed dimensional vector for each sequence. They are mainly used for speech recognition, trans-

Misidentification probability with 70% efficiency	b jets	c jets	light jets
10 GeV/c < p_T^{jet} < 20 GeV/c	11.1%	19.98%	1.44%
20 GeV/c < p_T^{jet} < 30 GeV/c	9.40%	31.6%	0.930%
p_T^{jet} > 30 GeV/c	10.2%	44.4%	0.395%

Table 6. Table showing the misidentification probability of tagging jets of different flavors with a high p_T electron for different p_T^{jet} ranges with a 70% efficiency using the fully-connected feedforward neural network model.

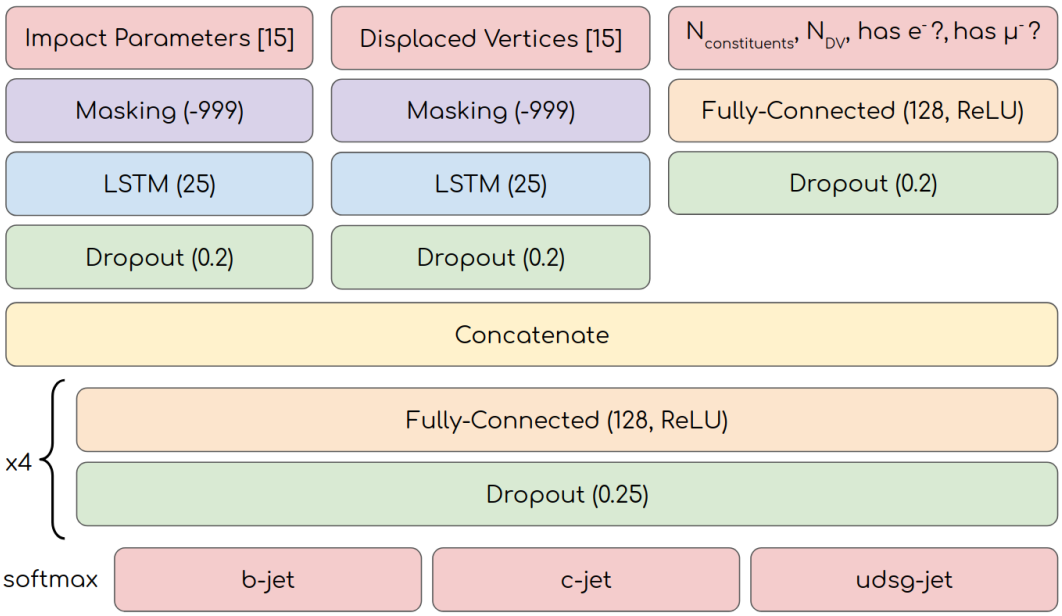

Figure 5. Diagram showing the neural network architecture starting with three branches for the inputs at the top that merge into one network with three outputs at the bottom for classification.

lation, and image captioning, where the sequences in those cases consist of words. LSTMs are useful because they have a long term memory, so they use previous information in the sequence to help with their subsequent predictions. This is possible due to internal gating mechanisms that enable LSTMs to regulate how much to remember and forget long and short term information.

As before, we model each final-state particle's track as a helix in a 0.5 T magnetic field. We then make use of the different track properties of the different jet flavors as discriminators to be used as inputs to the proposed concatenated model. These are summarized in the following subsection.

4.1 Discriminators

The discriminators used as inputs for training this neural network model are divided into three groups:

Impact Parameters We sort the impact parameter (IP) values of the jet constituents in descending order, as explained in Section 3.1. If a jet has more than 15 constituents, we only consider the 15 with the largest IP values. If a jet has less than 15 values, we pad the array of IP values with the value -999 until the array is of size 15. The -999 values are masked later, as explained in Section 4.2.

Displaced Vertices We sort the displaced vertex position (DV) values of the jet constituents in descending order, as explained in Section 3.1. If a jet has more than 15 constituents, we only consider the 15 with the largest DV values. If a jet has less than 15

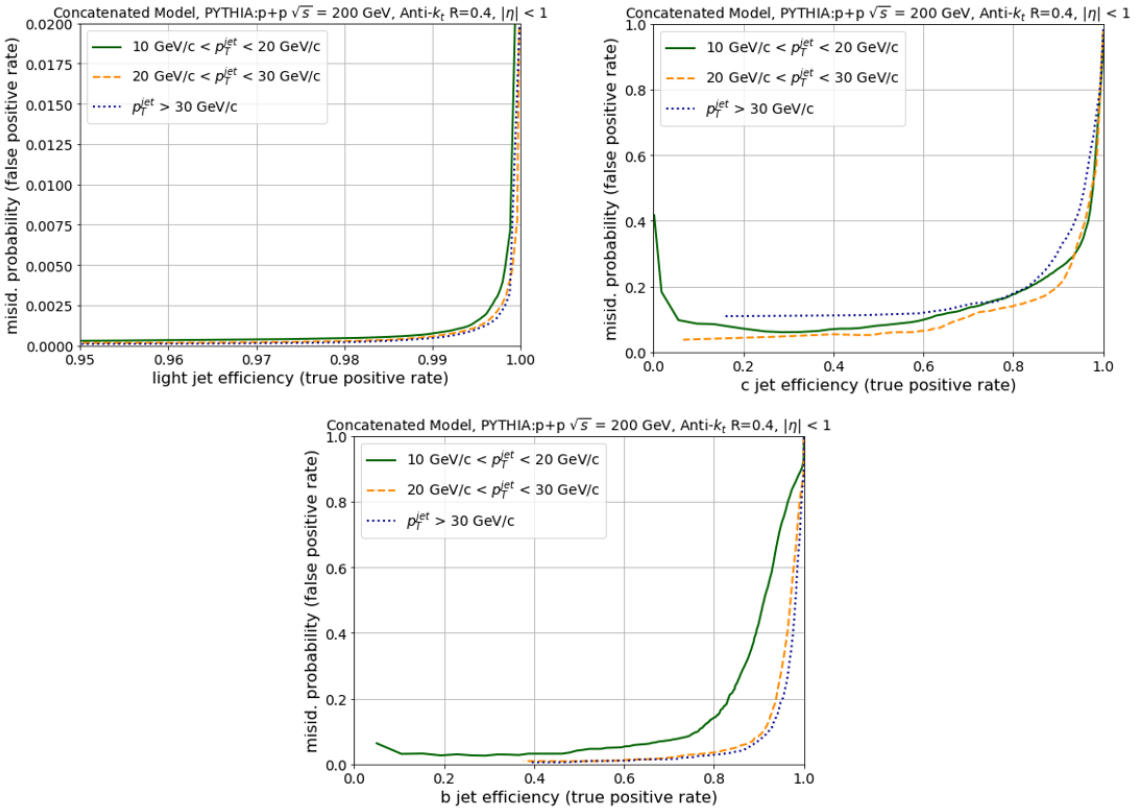

Figure 6. ROC curves showing the misidentification probability of bottom, charm, and light jet samples, weighted according to Table 1, as a function of the efficiency of tagging them using the concatenated model.

values, we pad the array of DV values with the value −999 until the array is of size 15. Again, the −999 values are masked later, as explained in Section 4.2.

$N_{constituents}$, N_{DV}, e^- constituent, μ^- constituent Finally, our inputs also include the total number of jet constituents, the total number of jet constituents with displaced vertices, and two Booleans indicating whether a jet has an electron or a muon as part of its constituent.

4.2 Architecture

The network starts out with three branches for the inputs, as shown in Figure 5, divided as described in the previous subsection. The branches are then merged into one network that outputs the flavor of the jet.

Masking layers are applied to the first two branches to mask the padded −999 values in the input arrays of size 15 that are added when a jet has less than 15 constituents. Those layers tell the network to disregard those values as placeholders. LSTM layers are applied after the masking layers of the first two branches to learn from the sequences of impact parameters and displaced vertices of the jet constituents, which are described in the

Misidentification probability with 70% efficiency	b jets	c jets	light jets
10 GeV/c < p_T^{jet} < 20 GeV/c	7.29%	13.3%	4.46×10^{-3}%
20 GeV/c < p_T^{jet} < 30 GeV/c	2.05%	10.9%	4.22×10^{-3}%
p_T^{jet} > 30 GeV/c	1.55%	14.8%	2.51×10^{-3}%

Table 7. Table showing the misidentification probability of tagging jets of different flavors for different p_T^{jet} ranges with a 70% efficiency using the concatenated model.

previous section. A fully-connected feedforward layer is applied to the last branch to learn from the rest of the parameters.

The three branches are then concatenated. Four fully-connected layers follow, ending with a softmaxed output layer similar to that from the previous model. We use a dropout layer after every layer, which randomly drops out nodes during the training in order to avoid overtraining.

This model is trained using Keras [4] with the TensorFlow [5] back-end. EarlyStopping is implemented to stop the training process after the model's validation loss stops decreasing for 20 epochs as described in Section 3.2. We use the Rectified Linear Unit (ReLU) activation function [6] for each layer, except for the output layer. Finally, the model utilizes the Adam optimizer [7] and the categorical cross entropy loss function during the training as before.

4.3 Performance

We train this model on 200,000 jets of each flavor and p_T^{jet} range, and we test it on 100,000 jets of each flavor and p_T^{jet} range to create the ROC curves. As before, the training is done on equal amounts of jets for each flavor in order to not introduce any biases in the network's prediction. However, we scale the predictions, as explained in Section 3.3, according to the relative production rates listed in Table 1.

The model's performance is displayed in Figure 6 as a ROC curve. The misidentification probabilities that we get for a 70% efficiency using this model are summarized in Table 7. By comparing these results to those of the previous model without the lepton selection, we note a significant enhancement in performance.

5 Conclusion and Future Outlook

We found that the relative production rates at RHIC energies for the jets of different flavors make it difficult to tag heavy flavor jets with a traditional fully-connected feedforward neural network model. In this paper, we propose the solution to trigger on jets with a high p_T muon or electron. This balances out the relative jet production rates due to the semi-leptonic decay channel of heavy-flavor hadrons. However, since the substructure of the jets is important for studying QGP properties, we introduce a new model with LSTM layers in order to not bias our jet samples. We try this model on an unbiased jet sample and show that it outperforms the previous model.

One can extend this model by introducing STAR or sPHENIX detector effects in order to use it for tagging heavy flavor jets at those experiments. Also, once the heavy flavor features are introduced into heavy-ion event generators, we could generate heavy-ion collisions to train the concatenated model on and study its performance.

Acknowledgments

GH thanks the Eckardt Scholars Program for supporting this work through the Eckardt Scholars Senior Research Project Grant.

References

[1] T. Sjostrand, S. Mrenna, and P. Z. Skands *Comput. Phys. Commun.* **178** (2008) 852-867, arXiv:0710.3820.

[2] M. Cacciari, G. P. Salam, and G. Soyez *JHEP* **04** (2008) 063, arXiv:0802.1189.

[3] M. Cacciari, G. P. Salam, and G. Soyez *Eur. Phys. J.* **C72** (2012) 1896, arXiv:1111.6097.

[4] F. Chollet, **https://github.com/fchollet/keras**, 2017.

[5] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. A. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zhang *CoRR* **abs/1605.08695** (2016), arXiv:1605.08695.

[6] V. Nair and G. E. Hinton in *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 807-814, 2010.

[7] D. Kingma and J. Ba, arXiv:1412.6980.

[8] S. Chatrchyan et al. (CMS collaboration) *Phys. Rev. Lett.* **113** (2014) 132301

[9] S. Hochreiter and J. Schmidhuber *Neural Computation* **9** (1997) 1735-1780, ISSN 0899-7667

[10] CMS collaboration, *Identification of heavy-flavour jets with the CMS detector in pp collisions at 13 TeV*, *JINST* **13** (2018) P05011–P05011