The $2B^2$ Tagger

Colm Sam

Supervised by Greg Ciezarek

Summer Student Report 2019

Abstract

The oscillations of neutral B mesons between their particle and anti-particle state provides a source of CP violation. In order to study the CP violation and the asymmetries within these neutral B meson systems, the flavour of the B meson (whether it was a B or Bbar) at the time of production needs to be known, hence the need for B-tagging. Many current B-tagging algorithms are based around the usage of single tracks, meaning it does not take advantage of the geometry, kinematics and position of the rest of the event hindering their overall performance. A novel approach called The $2B^2$ Tagger is proposed here which differs from conventional techniques by focusing on the usage of two track vertices instead. Then by leveraging on proven techniques seen in the analysis of B hadrons as well as machine learning, the proposed algorithm tries to cover the pitfalls of current taggers and the initial results are promising.

Contents

1 Introduction

The experiments being carried out at the LHC have the potential to answer some of the most fundamental questions we are confronted with as human-beings let alone modern-day physicists. One question in particular, is in regard to how the big bang should have created equal amounts of matter and antimatter in the early universe. However, in today's universe, we observe an overwhelming abundance of matter over antimatter, and its the task of the LHCb experiment located at CERN to find out why. One source of this matter-antimatter asymmetry comes from neutral B mesons, which can oscillate freely between their particle and anti-particle state. In order to measure these asymmetries within the neutral B-mesons, the flavour of the B (whether it was a B or $\overline{B}$) at time of production needs to be known. The process of determining the flavour at production is called B-tagging, and in this report a novel approach called The 2B^2 Tagger will be proposed and examined.

2 Theoretical Background

Neutral B-mesons can oscillate freely between their particle and anti-particle state, which is possible via the exchange of W± bosons and can be seen in figure 2.

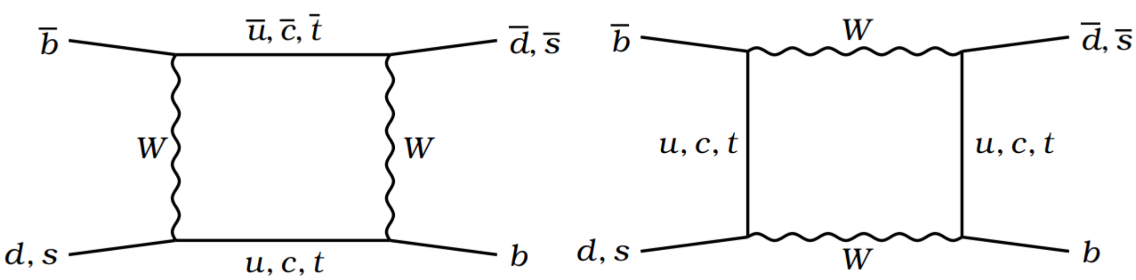

Figure 1: Feynman diagram depicting the oscillating neutral B mesons [1]

These oscillations of the neutral B mesons can provide a source of CP violation. An example of this CP violation is in the interference of the mixing and decay. This is when the decay rate of the B meson into a final state f

is different from the decay rate of the $\overline{B}$ meson into the same common final state f. This can be further seen in eq1 and figure 2.

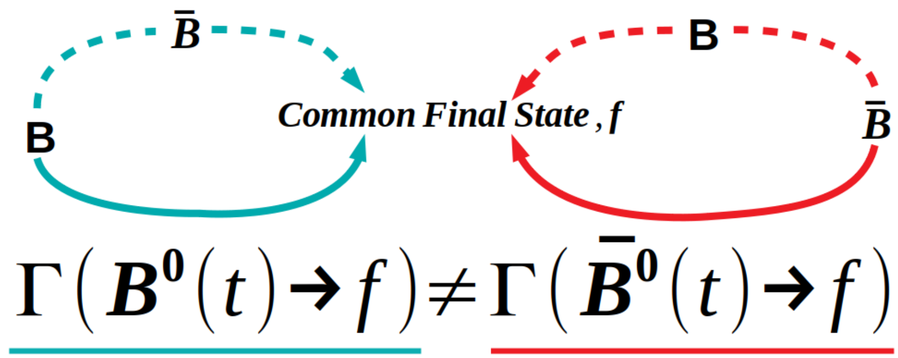

Figure 2: CP violation in the interference

The asymmetry $\mathcal{A}$ in the decay rates is given by the following:

$$\mathcal{A} = \frac{\Gamma(B^0(t) \to f) - \Gamma(\overline{B}^0(t) \to f)}{\Gamma(B^0(t) \to f) + \Gamma(\overline{B}^0(t) \to f)} \tag{1}$$

But in order to measure these asymmetries we need to have known the flavour of the B-meson (whether it was a B or $\overline{B}$) at time of production, and thus we require B-tagging.

3 Current Approaches

Current flavour tagging approaches exploit the fact that b quarks are mostly always produced in b$\bar{b}$ pairs [2]. So the B event is naturally partitioned into two-sides. Firstly you have the side containing the neutral oscillating B meson which were trying to tag as containing a b or $\bar{b}$ quark. This is known as the Signal B and this side of the B event is known as the Same Side (SS). There are some flavour tagging algorithms which try to tag the Signal B by looking at the hadronisation of the Signal B itself. These algorithms are called SS taggers and will not be discussed in detail in this report.

We then have the Opposite Side (OS), which is where the other b quark resides and can hadronise into a variety of different possible B mesons. This

B meson is known as the Tagging B. The algorithms which use the decay particles of the Tagging B to tag the Signal B are called OS taggers. The general outline of how the SS and OS taggers work can be seen in figure 3.

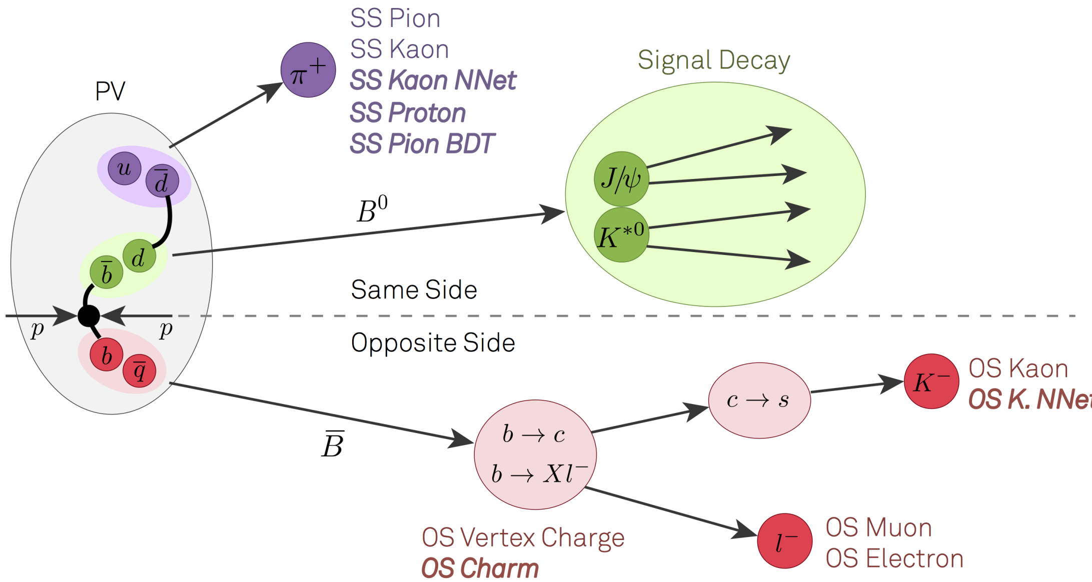

Figure 3: Diagram summarising current flavour tagging algorithms. The decay of the Signal B is seen in green. The hadronisation of the Signal B is seen in purple and the algorithms which use these particles for tagging are the SS taggers. The decay of the Tagging B is seen in red, and the algorithms which use those particles to determine the tag are called OS taggers.

3.1 Example $B^- \rightarrow \overline{D}^0 \mu^- \overline{\nu}_\mu$

To fully understand how the current approaches actually work such as the OS tagger, consider the example case seen in figure 4:

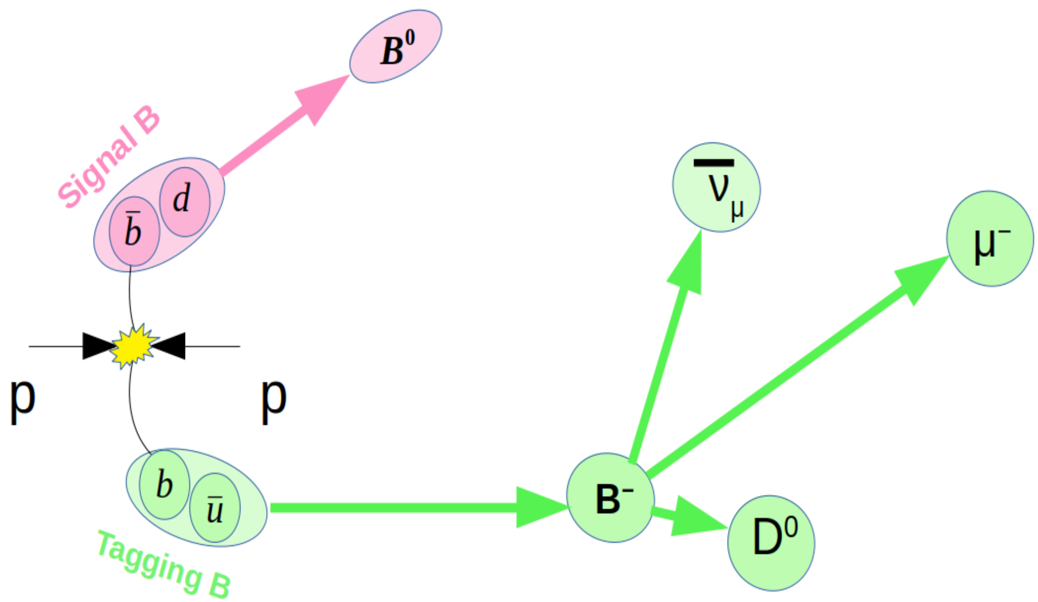

Figure 4: OS Tagger Example Case 1

In this particular example a $\bar{b}$ quark is produced on the Same Side (SS) and hadronises into the Signal B, a B^0 in this case. However, we are not further concerned with what happens to the Signal B, as the OS taggers work independently to what ensues on the Same Side. It is then clear that on the Opposite Side a b quark is produced, and in this instance hadronises into a B^- meson which is the Tagging B. This B^- subsequently decays into a D^0 meson, a muon μ^- and an anti-neutrino $\overline{\nu_\mu}$. The OS tagger is able to find the μ and sees that its negatively charged. If the μ is found to be negative, it can then be inferred that the Tagging B is also negative, and when the Tagging B is negative it must mean that it contains a b quark. Finally, under the assumption that a $b\bar{b}$ pair was originally produced at the start of the event, then if the Tagging B does in fact contain a b quark, it must mean that the Signal B contains a $\bar{b}$ quark. This process of using the charge of the decay particles of the Tagging B to infer the tag of the Signal B is how the OS Tagger works.

3.2 Example $\mathrm{B}^- \to \overline{D}^0(\to K^-\mu^+\nu_\mu)\mu^-\overline{\nu}_\mu$

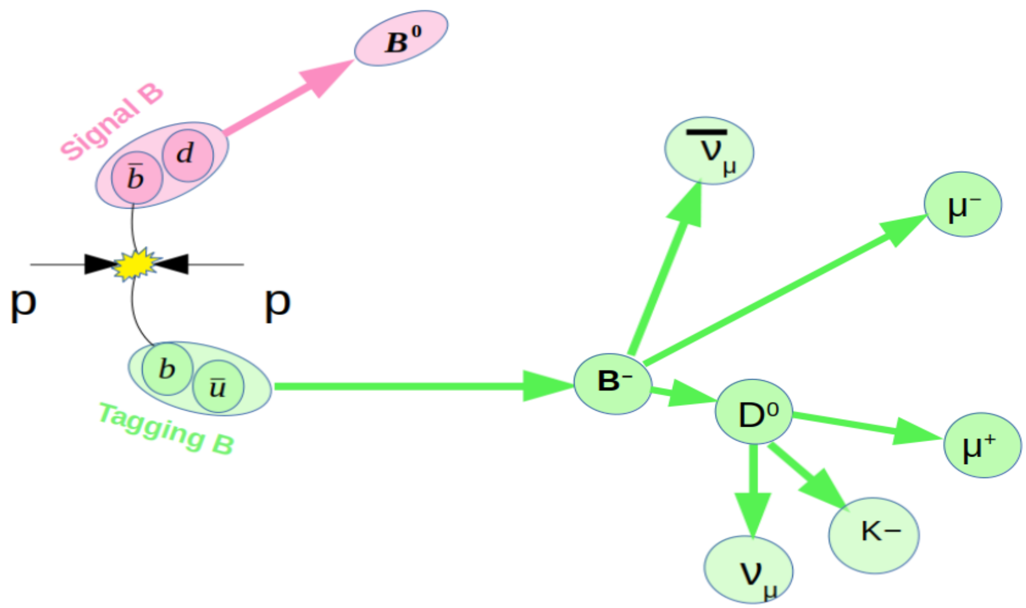

Figure 5: OS Tagger Example Case 2

Consider the same example as seen in Figure 4, but now the D^0 decays into a K^- meson, an anti-muon μ^+ and a neutrino ν_μ. As seen in figure 5, the OS tagger will now see two muons: the μ^- which decayed directly from the Tagging B and would give us the correct tag on the Signal B; and the μ^+ which decayed from the D^0 and would give us the incorrect tag. Without the kinematics and position of the rest of the decay, along with its static rule based decision making, the OS tagger finds it very hard to tell apart the muon with the correct charge for tagging (the μ^- in this case) and the muon with the wrong charge for tagging (the μ^+).

3.3 Further Failure Case of the OS Tagger

The fail case discussed along side figure 5 is just one out of a very extensive list, of possible situations where decay particles with the wrong charge for tagging are found and consequently confuse our OS taggers. But to further drive home just how complex the problem of tagging can be, theres also the case to consider when the Tagging B ends up as a neutral B^0 or B^0_s, which means it could oscillate, thus changing flavour and confusing our OS taggers even more. Then there's the worse case situation where the Tagging B or any of its decay particles are not even found. The fail cases of the OS tagger outlined above should make it clear that theres potential room for improvement, which leads us nicely to the introduction of The $2B^2$ Tagger which is discussed in the next section.

4 The $2B^2$ Tagger Approach

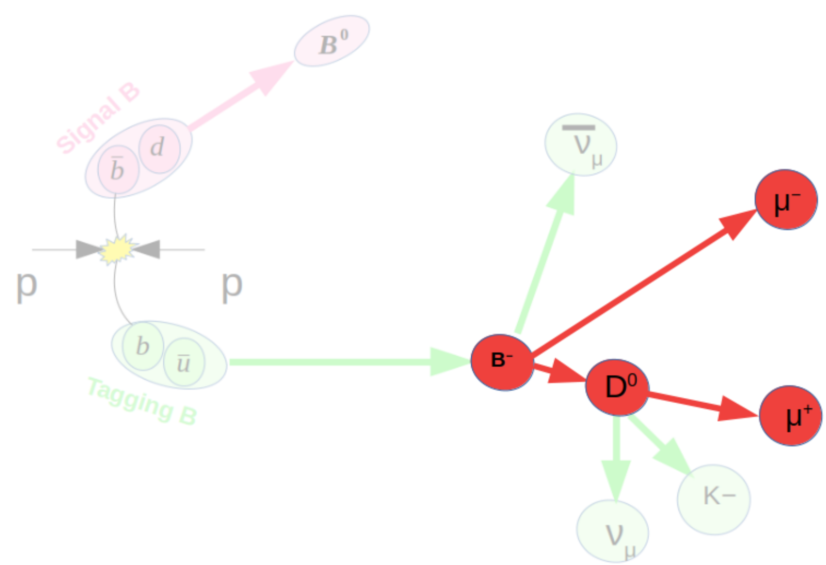

Figure 6: Example of a TwoBody (TB) highlighted in red

The $2B^2$ Tagger is an abbreviation for The Two Body B Tagger. The Two Body aspect is emphasised as unlike current approaches to OS tagging which is generally based around the usage of single tracks, the $2B^2$ Tagger is centred around the usage of two body vertices, ie, two tracks that are measured to be consistent with originating from a single point. An example of a two body (TB) which our tagger could use can be seen in figure 6 in red, which is made up of two tracks: the μ^- track which came directly from the B^- and the μ^+ which decayed from the D^0. The $2B^2$ Tagger could then find the rest of the charged tracks which make up the decay of the Tagging B, in this case it would be the addition of the K^-. The $2B^2$ Tagger would then have a three body system, the K^- track along with the μ^- and μ^+ tracks, giving the system the correct overall charge to help produce the correct tagging decision. How exactly The $2B^2$ Tagger starts with a TB such as the one seen in figure 6 and ends up with the correct tagging decision will be explained in the following sections.

4.1 Data set, Classifier and Training Set up

The data used for the training and testing of the taggers is Monte Carlo (MC) $B^{\pm} \to J/\psi K^{\pm}$ simulated data. This means that the Signal B is either a B^+ or a B^-, rather than a neutral B meson. Like the OS taggers, The $2B^2$ tagger works independently of what happens to the Signal B, and for that reason the Signal B is a charged non-oscillating B meson so that its easier to check if the tag is correct.

The classifier used was Microsoft's Light Gradient Boosting Machine (LGBM) [3]. The train and test data sets were generated completely independently from each other, this was done to stop different TB candidates from the same event being in both the training subset and the testing subset, and would have consequently caused the classifier to overfit otherwise. The classifier is trained via a k-fold cross validation which is then predicted and tested on the unseen test subset.

4.2 First Stage: The Two Body's (TBs)

For any given event, The $2B^2$ Tagger works by reconstructing many TB candidates. In order for this approach to work, we need to find the TB that originated from the Tagging B, this was done by following the strategy of the LHCb topological trigger MVA [4]. This involved feeding in all the reconstructed TBs into a Multi Variant Analysis (MVA) (or in other words, a Machine Learning (ML) algorithm) and have the MVA filter out the TBs that don't come from the decay of the Tagging B as illustrated in figure 7.
Features of the TBs used to train the MVA try and capture the geometry and kinematics of the TBs, a handful of the features used can be seen in the pair plot in figure 8. The whole list of features used can be seen in the Appendix. For each TB, the MVA outputs a score from zero to one on the likeliness that the TB originated from the Tagging B. The output of the MVA can be

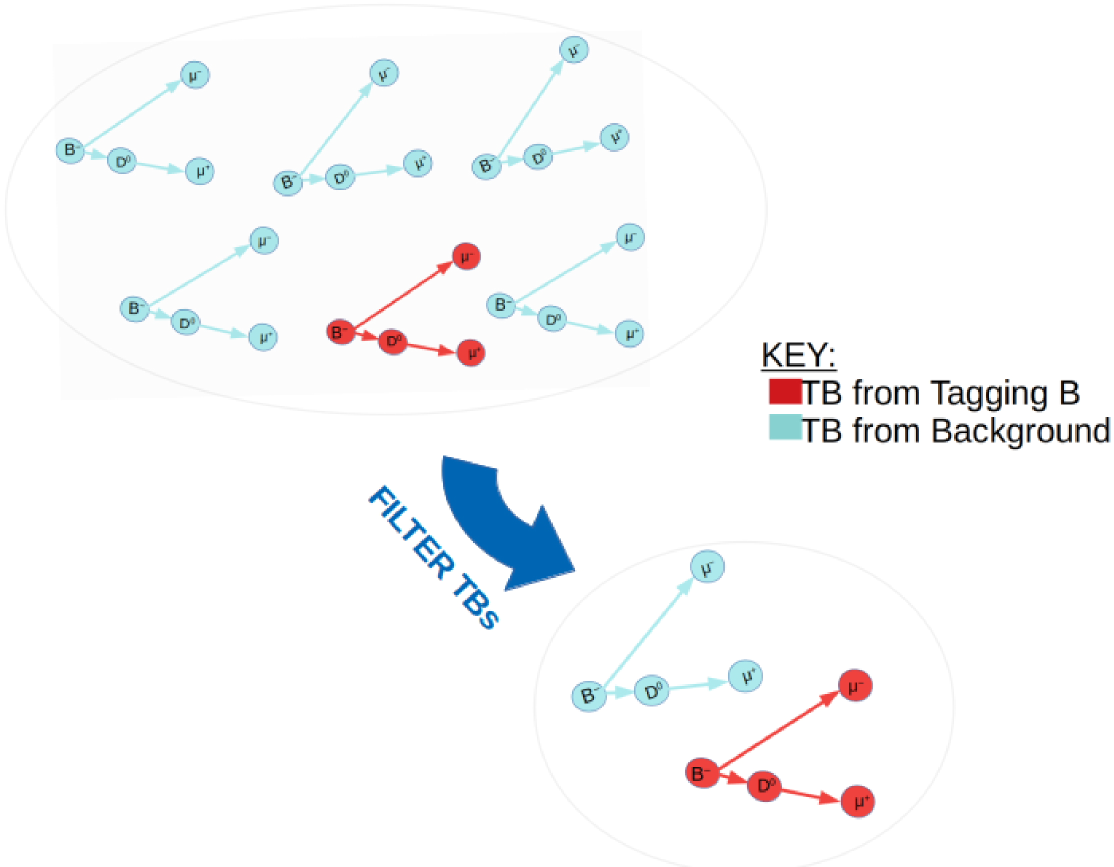

Figure 7: Schematic overview of the First Stage. For any given event, potential TB candidates are found and reconstructed. TB candidates are pairs of tracks within the event which are measured to be consistent with originating from a single point and possibly come from the decay of the Tagging B. In order for The 2B^2 Tagger to work properly we need to find the TB candidate which originated from the Tagging B. Thus as seen in figure 7, all the reconstructed TB candidates are passed through the First Stage MVA to filter out the TB candidates which the MVA thinks are least likely to originate from the Tagging B.

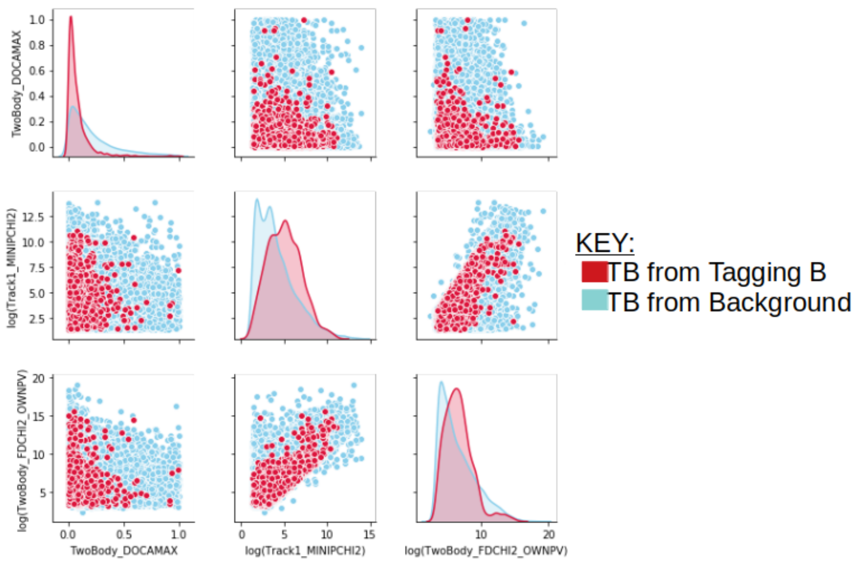

Figure 8: Pairplot of some of the features used in TB MVA

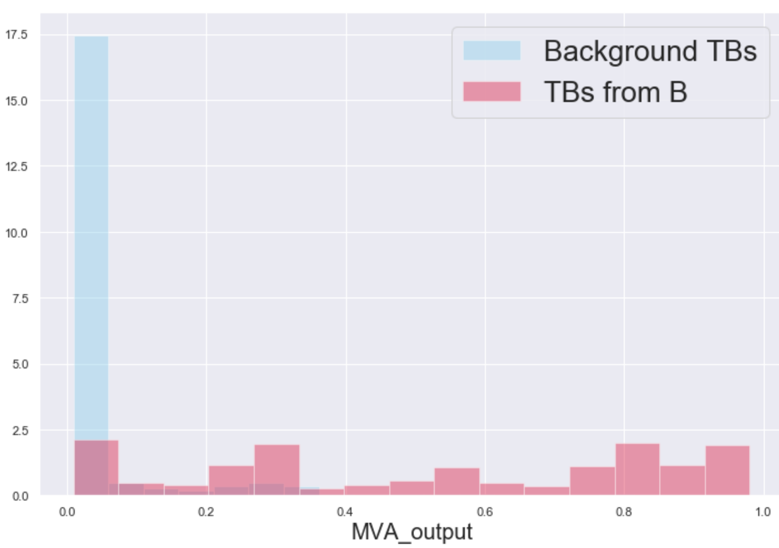

Figure 9: Distribution of the MVA output of the First Stage classifier, which achieved an ROC AUC score of 0.95.

4.3 Second Stage: The Extra Tracks (ETs)

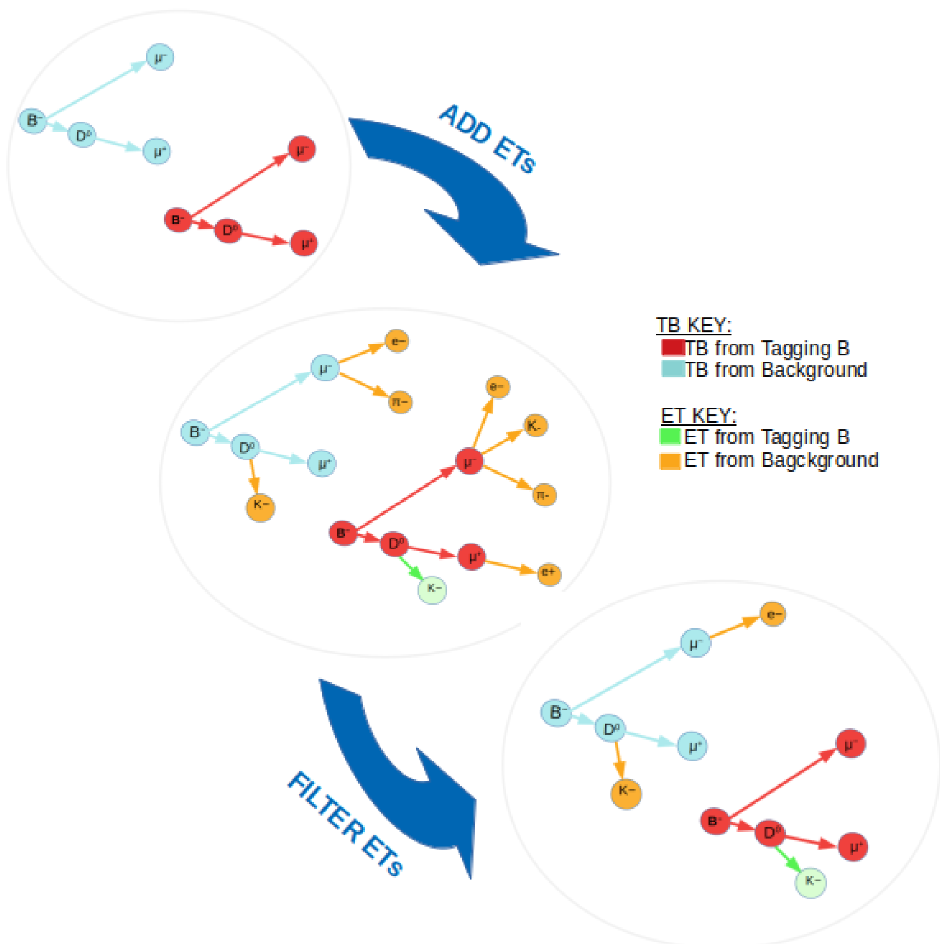

Figure 10: Schematic overview of the Second Stage. The TB candidates which have made it through from the First Stage have extra tracks (ETs) reconstructed and added to them. Then similar to what happens in the First Stage, the ETs are passed through a MVA to filter out the ETs which are less likely to have originated from the Tagging B. The ETs are added in order to find as much of the decay chain of the Tagging B as possible, and thus giving our tagger the best chance to produce the correct tag.

The TBs which have a MVA score above the desired threshold from the First Stage get to move onto the Second Stage. In the Second Stage, each TB is taken and extra tracks (ETs) are added to them, this is to try and find as much of the decay chain as possible, helping the Tagger produce the correct tag. Then similar to what is done with the filtering of the TBs in the First Stage, the ETs that have been added to the TBs are filtered using an (Isolation) MVA [5][6] to try and find the ETs which are associated with the decay of the Tagging B. The Second Stage is illustrated in figure 10.

The ET MVA learns which ETs originate from the decay of the Tagging B by being fed features describing the kinematics and geometry of the ET itself, the features used in the First Stage to describe the TB from which the ET is associated with, is also fed into the Second Stage MVA as well. Examples of some of the features that try and capture the characteristics of the ETs in the Second Stage include the track reconstruction quality, transverse momentum and vertex fit quality of ET to the TB. A pair plot of a few of these features can be seen in figure 11, the whole set of features used can be seen in the Appendix.

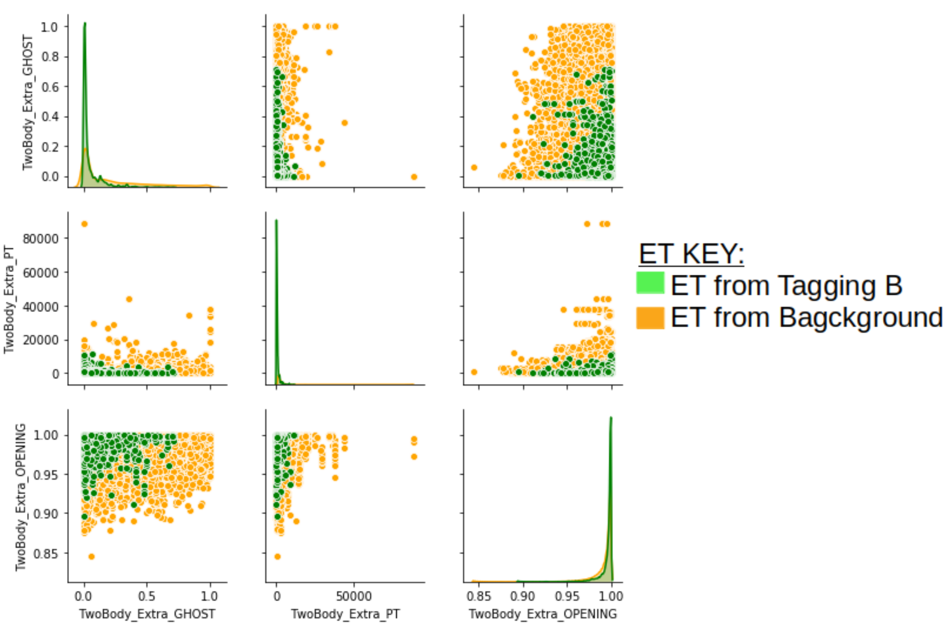

Figure 11: Pairplot of some of the features used in the ET MVA

Similarly to the Fist Stage, the Second Stage MVA outputs a score from 0 to 1 for any given ET, where 1 is when the MVA is most confident that the ET originates from the decay of the Tagging B, and 0 is when its most confident that the ET does not originate from the Tagging B. Setting the decision threshold (ie, any ET that scores above this is deemed to be associated with the Tagging B) to 0.1, the MVA is able to reject 88% of the background (ETs not from B) while keeping 85% of the signal (ETs from B). The output of the Second Stage MVA can be seen in figure 12.

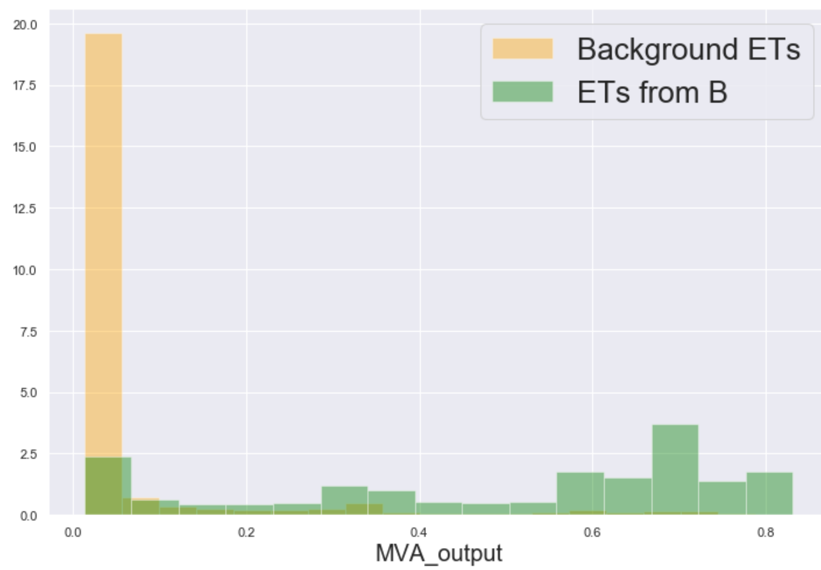

Figure 12: Distribution of the MVA output of the Second Stage classifier, achieving an ROC AUC score of 0.92

4.4 Line of Flight (LOF)

Once most of the background ETs are filtered out in the Second Stage, we can then start to utilise the most important new feature which comes from starting off with a two track vertex. This feature allows the estimation of the Tagging B rest frame using the Line of Flight technique (LOF) [5] [6]. Using the LOF, we can then calculate the energy of the decay chain particles within the B centre of mass (COM) frame. To see the discriminative power contained in having the COM energies of the decay chain particles, refer back to one of the fail cases of the OS tagger seen in figure 5, where the Tagging B decays in the following mode $B^- \rightarrow \overline{D}^0(\rightarrow K^-\mu^+\nu_\mu) + \mu^- + \overline{\nu}_\mu$. The OS tagger fails as it sees two muons with two different charges and thus two different possible tagging outcomes, and as the OS tagger cant tell the difference between the two muons, the tagger struggles to perform. However with the introduction of the muons COM energy, which can bee seen in figure 13, it clearly shows that the muons which have the wrong charge for tagging have a lower COM energy than the muons which have the right charge for tagging, and thus using this information will help the overall tagging performance.

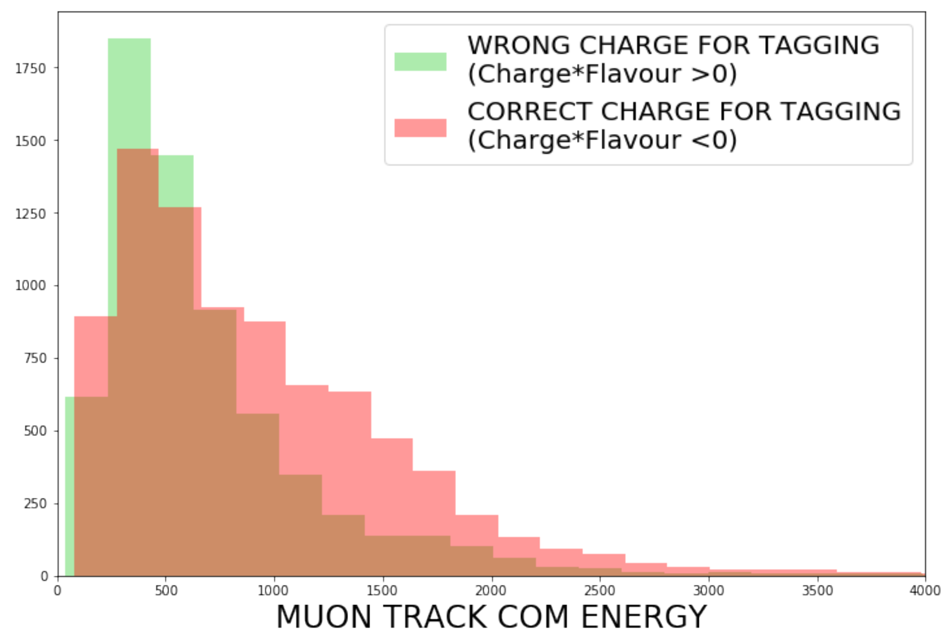

Figure 13: The distribution of the COM energies of muons using the Line of Flight (LOF) technique. It shows the discriminative power that the COM energies of the muon hold, as the muons with higher COM energy (distribution in dark pink) are more likely to have the right charge for tagging and thus helps in the failure case seen in Section 3.2

4.5 Third Stage: Tagging

Once the First Stage has picked out promising TB candidates along with their corresponding promising ETs selected by the Second Stage, its time to put them all together to produce a final tag for each event. A final tag is produced by combing together the kinematics, geometry, particle identification (PID) and LOF information of the TBs and ETs, which is then fed into a single MVA classifier which outputs a final tag per TB.

The MVA outputs a 1 if it thinks the Signal B was a B meson at time of production or it outputs a 0 if it thinks it was a $\overline{B}$. However, sometimes even after the filtering of the TBs in the First Stage, there can sometimes be more than one TB candidate per event. In these cases we combine the tags of multiple TBs using a weighted mean to then finally output a tag per event. Where the weights in this case are the TB scores outputted by the MVA in the First Stage. This is to try and prioritise the tags of the TBs that are most likely to originate from the Tagging B. The final per event tagger tested on the unseen test subset achieved an ROC AUC score of 0.628, an overall accuracy of 60% and an effective tagging power of 2.2% [2]. The MVA output for the test data can be seen in figure 14.

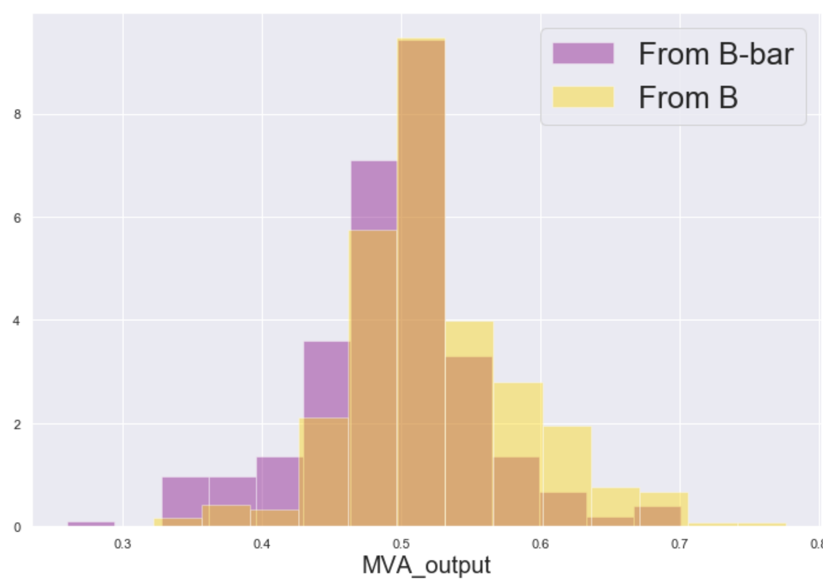

Figure 14: Distribution for the MVA output of the per event tagger, which achieves an ROC AUC of 0.628 and overall accuracy of 60%.

5 Conclusion

A new approach to B-tagging was proposed and discussed, which differs from current techniques such as the OS tagger by focusing on two track vertices rather than single tracks. The $2B^2$ Tagger uses a combination of machine learning and proven techniques from the analysis of B hadrons, to take advantage of the position and kinematics of the rest of the event which the OS taggers fail to do. The initial results are promising, where the final tagger was able to achieve an ROC AUC of 0.628, an overall accuracy of 60% and an effective tagging power of 2.2%. Furthermore, the overall classifiers performance was limited by the availability of a larger data set to train on. There's also plenty of room for performance gain from the optimisation of the different thresholds, hyperparamter tuning of LightGBM and feature engineering. The $2B^2$ Tagger also needs to be tested on real data, but hopefully due to its mostly uncorrelated approach when compared with other taggers, it should be able to add something.

Acknowledgements

Foremost, I would like to express sincere gratitude to my supervisor Greg Ciezarek for his continuous guidance, knowledge and patience throughout the whole of this project, it was a real pleasure working with you. Besides my supervisor, I would like to thank CERN, the LHCb collaboration and the Summer Student Team for making the Summer Student programme possible and making it as great as it is, with special thanks to Adriana Bejaoui for really making my time at CERN possible with the endless support she gave me in the problems I faced upon arriving. I would also like to give thanks to everyone in my office for creating such a lovely working environment, it was a real blessing working alongside you all and I thoroughly enjoyed our time spent together. Lastly but not least, I would like to mention my fellow Summer Students, I am truly grateful to have been constantly surrounded by such brilliant, smart and passionate people and I owe many of you a great deal thanks for making this summer as special as it was.

References

[1] Kevin Heinicke. Optimization of Flavour Tagging Algorithms for the LHCb Experiment, 30. September 2016.

[2] Katharina Kreplin. A Novel Flavour Tagging Algorithm using Machine Learning Techniques and a Precision Measurement of the B0–B0 Oscillation Frequency at the LHCb Experiment: Section 3.42 on derivation of tagging power.

[3] Guolin Ke. Lightgbm, light gradient boosting machine. https://github.com/microsoft/LightGBM, 2017.

[4] Tatiana Likhomanenko. Lhcb topological trigger reoptimization. 2015.

[5] Greg Ciezarek. Semileptonic b baryon decays talk on behalf of the lhcb.

[6] Greg Ciezarek. Searches for lepton number violation, and flavour violation beyond the Yukawa couplings at LHCb, 20. August 2014.

Appendix

TB Vertex Features	TB Track Features	ET Features	Label
TwoBody_DIRA_OWNPV	Track1_PT	-	TwoBody_FromSameB
TwoBody_DOCAMAX	Track2_PT	-	-
TwoBody_ENDVERTEX_CHI2	log(Track1_MINIPCHI2)	-	-
log(TwoBody_FDCHI2_OWNPV)	log(Track2_MINIPCHI2)	-	-
TwoBody_FD_OWNPV	Track1_Charge	-	-
TwoBody_M	Track2_Charge	-	-
TwoBody_IPCHI2_OWNPV	Track1_PX	-	-
TwoBody_Mcorr	Track1_PY	-	-
TwoBody_PT	Track1_PZ	-	-
TwoBody_n_Extra	Track2_PX	-	-
TwoBody_M	Track2_PY	-	-
TwoBody_PE	Track2_PZ	-	-
TwoBody_PX	Track1_ProbNNe	-	-
TwoBody_PY	Track1_ProbNNk	-	-
TwoBody_PZ	Track1_ProbNNp	-	-
TwoBody_ENDVERTEX_X	Track1_ProbNNpi	-	-
TwoBody_ENDVERTEX_Y	Track1_ProbNNmu	-	-
TwoBody_ENDVERTEX_Z	Track1_ProbNNghost	-	-
TwoBody_OWNPV_X	Track2_ProbNNe	-	-
TwoBody_OWNPV_Y	Track2_ProbNNk	-	-
TwoBody_OWNPV_Z	Track2_ProbNNp	-	-
-	Track2_ProbNNpi	-	-
-	Track2_ProbNNmu	-	-
-	Track2_ProbNNghost	-	-

Table 1: The variables used in the First Stage process. The variables in the first column contain the features which describe the properties of the Two Body TB) vertex, ie, the point where the two tracks meet. The second column contains the variables which describe the two tracks which make up each TB. As the First Stage is only concerned with filtering out the TBs that don't originate from the Tagging B, information regarding the extra tracks (ETs) is not needed and thus there's no features contained in the third column. The fourth column contains the label, which is the variable which the First Stage ML classifier is trying to predict once its been trained on the features in columns one and two.

TB Vertex Features	TB Track Features	ET Features	Label
TwoBody_DIRA_OWNPV	Track1_PT	TwoBody_Extra_VERTEXCHI2	TwoBody_Extra_FromSameB
TwoBody_DOCAMAX	Track2_PT	TwoBody_Extra_MINIPCHI2	-
TwoBody_ENDVERTEX_CHI2	log(Track1_MINIPCHI2)	TwoBody_Extra_OPENING	-
log(TwoBody_FDCHI2_OWNPV)	log(Track2_MINIPCHI2)	TwoBody_Extra_PT	-
TwoBody_FD_OWNPV	Track1_Charge	TwoBody_Extra_FLIGHT	-
TwoBody_M	Track2_Charge	TwoBody_Extra_TRACKCHI2	-
TwoBody_IPCHI2_OWNPV	Track1_PX	TwoBody_Extra_GHOST	-
TwoBody_Mcorr	Track1_PY	TwoBody_Extra_TYPE	-
TwoBody_PT	Track1_PZ	-	-
TwoBody_n_Extra	Track2_PX	-	-
TwoBody_M	Track2_PY	-	-
TwoBody_PE	Track2_PZ	-	-
TwoBody_PX	Track1_ProbNNe	-	-
TwoBody_PY	Track1_ProbNNk	-	-
TwoBody_PZ	Track1_ProbNNp	-	-
TwoBody_ENDVERTEX_X	Track1_ProbNNpi	-	-
TwoBody_ENDVERTEX_Y	Track1_ProbNNmu	-	-
TwoBody_ENDVERTEX_Z	Track1_ProbNNghost	-	-
TwoBody_OWNPV_X	Track2_ProbNNe	-	-
TwoBody_OWNPV_Y	Track2_ProbNNk	-	-
TwoBody_OWNPV_Z	Track2_ProbNNp	-	-
-	Track2_ProbNNpi	-	-
-	Track2_ProbNNmu	-	-
-	Track2_ProbNNghost	-	-

Table 2: Variables used in the Second Stage process. The variables include all the features used in the First Stage plus additional features which describe the properties of the ETs which can e seen in the third column. The label is also different compared to the one seen in 1 as the Second Stage looks at filtering the ETs which don't come from the Tagging B and and thus a new label is required as the First Stage looks just at the TBs.

TB Vertex Features	TB Track Features	ET Features	Label
TwoBody_DIRA_OWNPV	Track1_PT	TwoBody_Extra_VERTEXCHI2	SignalB_ID
TwoBody_DOCAMAX	Track2_PT	TwoBody_Extra_MINIPCHI2	-
TwoBody_ENDVERTEX_CHI2	log(Track1_MINIPCHI2)	TwoBody_Extra_OPENING	-
log(TwoBody_FDCHI2_OWNPV)	log(Track2_MINIPCHI2)	TwoBody_Extra_PT	-
TwoBody_FD_OWNPV	Track1_Charge	TwoBody_Extra_FLIGHT	-
TwoBody_M	Track2_Charge	TwoBody_Extra_TRACKCHI2	-
TwoBody_IPCHI2_OWNPV	Track1_PX	TwoBody_Extra_GHOST	-
TwoBody_Mcorr	Track1_PY	TwoBody_Extra_TYPE	-
TwoBody_PT	Track1_PZ	TwoBody_Extra_Erest	-
TwoBody_n_Extra	Track2_PX	-	-
TwoBody_M	Track2_PY	-	-
TwoBody_PE	Track2_PZ	-	-
TwoBody_PX	Track1_ProbNNe	-	-
TwoBody_PY	Track1_ProbNNk	-	-
TwoBody_PZ	Track1_ProbNNp	-	-
TwoBody_ENDVERTEX_X	Track1_ProbNNpi	-	-
TwoBody_ENDVERTEX_Y	Track1_ProbNNmu	-	-
TwoBody_ENDVERTEX_Z	Track1_ProbNNghost	-	-
TwoBody_OWNPV_X	Track2_ProbNNe	-	-
TwoBody_OWNPV_Y	Track2_ProbNNk	-	-
TwoBody_OWNPV_Z	Track2_ProbNNp	-	-
MissingM2	Track2_ProbNNpi	-	-
-	Track2_ProbNNmu	-	-
-	Track2_ProbNNghost	-	-
-	Track1_Erest	-	-
-	Track2_Erest	-	-

Table 3: Variables used in the Third Stage. Similar to the list of variables used in the Second Stage seen in 2 but includes the features from the Line of Flight technique as well as a different label which contains the tagging information.