Automatic Labelling of Student Results as PCD, RET, MBR and MRNM

Manape Campbell 2127079
School of Computer Science and Applied Mathematics
Faculty of Science
The University of the Witwatersrand
Johannesburg, South Africa
Email: 2127079@students.wits.ac.za
Supervised by: Dr. Ashwini Jadhav and Dr. Ritesh Ajoodha

***Abstract*—Any educational institution needs to have a preliminary knowledge of the enrolled students in order to predict their future academic success. This will help the institution to identify students who are promising and also provide more support to students who are not likely to progress. This paper aims at forecasting students first and second-year outcomes, to deduce if they are at risk of getting academically excluded or not. Seven predictive models, namely, Naive Bayes, Logistic Regression, Decision trees, Random forests, Support vector machines, Multi Layer Perceptron and Ensemble Meta-based trees were trained. Random forest trees was a good model when compared to others. It achieved an accuracy of 72% on first year data and 73.7% on second year data. Naive Bayes was the worst model with an accuracy of less than 60% on both first and second year data. The significance of this study is to promote student success in higher institutions.**

***Index Terms*—Predicting student performance, Machine learning, Outcome codes**

I. Introduction

Student performance in university courses is of great concern to the higher education managements. Every university has a large number of courses and it is difficult for students to be familiarized with all the possibilities. At the end of the study, all the courses which are compulsory needs to be completed and in addition, some of the elective courses needs to be completed for the student to be granted a degree [1].

There has been many attempts to predict student performance using characteristics of the learner [2], [3] and [4]. [2] used students' gender, educational background and family behaviours as features for predicting student performance. [4] used the students' logged data with features of how the student was interacting with the LMS while [3] used students' academic information such as marks, course code, year of study, etc. Machine learning models have been used to predict student performance. [2] used the Ensemble Meta-Based tree model, [4] used Random Forest trees, Support Vector Machines, Multi-Layer Perceptron, Naive Bayes and Logistic Regression models, [3] used the same models used by [4] but he also used Decision trees and k-Nearest Neighbours. The evaluation functions used to evaluate the models includes, confusion matrix, accuracy, precision, recall and f1-score. This authors have accurately predicted the student performance using machine learning models.

The purpose of this research is to create an auto-tagging system which examines student grades and assigns an outcome-code. The system will attempt to learn the delicate boundaries between these class labels by accessing training data assembled over many years, by the teaching and learning committee. Such committees can use the system to assist them with making decisions on how to classify student grades. Most universities have young lecturers who do not have enough experience to understand properly how to use previous students performance to make proper decisions. The system can also be used by such lecturers or staff members to predict student performance.

The seven predictive models trained in this study are Naive Bayes, Logistic Regression, Decision trees, Random Forest trees, Support Vector Machines and Ensemble Meta-Based trees. Random Forest trees was the best model when compared to the other models. It achieved an accuracy of 72% on first year data and 73.7% on second year data. All other models achieved an accuracy of less than 70% besides Ensemble Meta-Based tree, which achieved an accuracy above 70% on second year data. Naive Bayes was the least performing model with an accuracy of 53.2% on first year data and 56.3% on second year data. All the models didn't really get great metrics.

The contribution of this research is to:

1) Perform classification using some common approaches such as Naive Bayes, Random Forest trees, Support Vector Machines and Multi-Layer Perceptron to predict student performance.
2) Present a predictive model to forecast students' first and second year outcome codes. This can help the staff members and lectures to understand which students are likely to be excluded.

The rest of the paper is organized as follows. In section II we examine the related work of some of the authors who attempted to predict student performance. In section III, we examine the method that was carried out for this research as well as the data and features that were used. In section IV, we

provide the results of this study and analysis of the results. Lastly, section V concludes the research report.

II. Related Work

This section comprises of three subsections. The first subsection provides the key features for predicting student performance found in the literature. The second subsection provides the predictive models which are used for predicting student performance. The last subsection provides the evaluation techniques used in the related research.

A. Features for predicting student performance

The key features for predicting student performance used in the literature are shown in Table I, [5], [6]. To handle cases where there are a lot of features, authors have applied feature selection or feature engineering techniques to reduce the number of features [2], [1], [4], [3]. [2] used pearson correlation method to reduce the number of features. [1] and [4] also did preprocessing and applied feature selection techniques to reduce the number of features. [3] used mutual information gain to select the best features. The features that we used in this study will be presented in Methodology section.

The outcome codes or the number of class labels used in the related work differs. [2] used 3 class labels; [7] used 4 class labels; [1] did a comparison of 5, 3 and 2 class labels, where using 2 class labels was the best case; [4] and [8] used 2 class labels and lastly [3] used 10 class labels. There is no strong difference between students with small differences in grades [1]. In our research, we used 6 class labels or outcome codes which are PCD, RET, MBR, MRNM, MFC and MBZ.

Features	Description
CGPA	Cumulative grade point average.
Internal assessment	Student's behaviour in the class such as exam marks, lab tests, class tests, attendance, and discussions.
Student's demographic	Student's characteristics that include gender, age, salary income, and family background.
External assessments	Factors coming from student's environment such as student's behaviour out of the class such as extracurricular activities, high school background, social interaction network, and psychometric.
Extracurricular activities and High school background	Activities which are not essential for normal class activities and pre-admission courses for universities.
Social interaction network	Interacting among students or instructors by using social networks.
Psychometric	These attributes can be collected after enrolment such as attitude, motivation, personality, and learning strategies.

TABLE I
FEATURES FOR PREDICTING STUDENT PERFORMANCE.

B. Models for predicting student performance

There are many different models in the related work and one cannot choose the best model, because they differ in many aspects such as learning rate, amount of training data, classification speed, robustness, etc. The common models used in the literature are: Support Vector Machines (SVM), Naive Bayes (NB), Multi-Layer Perceptron (MLP), Decision trees (DT), Random Forest trees (RFT), Ensemble Meta-Based tree (EMT), Logistic Regression (LR) and K-Nearest Neighbours (KNN).

SVM was used to predict student performance [1], [4], [9], [3]. SVM performed better in the dataset used by [1] with an accuracy of 89%. [2] used EMT which also performed good with an accuracy of 98.5%. [7] used CB with different models for clustering based, i.e., the author did a comparison of CB-MLP, CB-NB, CB-EMT and CB-J48. KNN and RFT were also used for predicting student performance [3] and [9]. DT is the most popular model used for predicting student performance in the literature [2], [7], [1], [9], [8]. LR is only used by three authors [3], [4], [8].

Table II summarizes the models used in the literature and their accuracies. In this study, we used NB, SVM, LR, DT, RFT, MLP and EMT.

C. Evaluation Techniques

The evaluation functions used in the literature include confusion matrix, accuracy, Recall, precision and f1-score. All the authors have evaluated their learnt models using all or some of the evaluation functions mentioned above [10], [2], [4], [1], [9], [6], [8], [3], [11]. We used confusion matrix and accuracy score to evaluate the performance of the models. More about the evaluation techniques will be explained in the next section.

III. Methodology

In this section, we describe the research aims, research motivation, research hypothesis and also the method that was employed for this research as well as the data, features, models and evaluation functions that were used for this research.

A. Research Aims

The main aim of this research is to create an auto-tagging system which examines student grades and assigns an outcome code to the grades, by establishing some criteria to assess class labels using the provided training data.

B. Research Motivation

In wits university, most students fail first year and repeat several times before they graduate. Some students get expelled from the university for poor performance. Students who are likely to get an outcome-code of MRNM, which indicates that the student has been excluded and will have to appeal for re-admission; can be provided with extra support or consider changing courses and try to take courses which the individual might be good at. Most universities have young lecturers who do not have enough experience to understand properly how to use previous student performance to make proper decisions. The system can also be used by such lecturers or staff members to predict student performance.

Author	Data	Features	Models	Accuracy
[2]	student educational data.	gender, academic information, students' behaviours, family behaviours.	EMT.	ACC = 98.5% achieved by J48 + NB Tree.
[7]	historical student records from MIS department of balqa applied university.	gender, age, address, graduating year, student's grades.	CB-MLP, CB-NB, CB-J48, CB-EMT.	ACC = 96.96% achieved by CB-EMT.
[1]	information system of Masaryk university.	biographical information, academic information, social behaviour.	NB, SVM, IBL, CR, OneR, DT, J48.	ACC = 89% achieved by SVM.
[4]	student log data from DEEDS LMS.	marks, features of student interaction with the LMS.	RFT, SVM, MLP, NB, LR.	ACC = 97.4% achieved by RFT.
[9]	2 datasets including student interaction with an e-learning platform.	biographical data, student academic data, student behaviour.	SVM, NB, KNN, RFT, DT.	ACC = 84% achieved by RFT.
[8]	student data collected at wits university.	biographical information, academic information, university registration information.	DT, FNN, NB, LR.	ACC = 82.3% achieved by FNN.
[3]	synthetic data generated which reflects data from wits university.	year of study, course code, final mark, final grade and encrypted student number.	DT, RFT, SVM, LR, KNN, NB.	ACC = 98% achieved by DT.

TABLE II
RELATED WORK. WHERE FNN: FEED FORWARD NEURAL NETWORKS, MLP: MULTIPLE LAYER PERCEPTRON, SVM: SUPPORT VECTOR MACHINES, LR: LOGISTIC REGRESSION, DT: DECISION TREES, RFT: RANDOM FOREST TREES, EMT: ENSEMBLE META-BASED TREE, NB: NAÏVE BAYES, IBL: INSTANCE BASED LEARNING, CR: CLASSIFICATION RULES, ONER: ONE RULE, KNN: K-NEAREST NEIGHBOURS, CB-MLP: CLUSTERING-BASED MLP, CB-NB: CLUSTERING-BASED NB, CB-J48: CLUSTERING-BASED J48 DECISION TREE, CB-EMT: CLUSTERING-BASED EMT.

C. Research Hypothesis

To predict student performance, we will make use of ML models as informed by the literature. The problem can be reduced to this question: given student historic data and student marks, can we predict the outcome-code of the student? Our goal is to train a machine learning model using student historic data that has been assembled over many years, by the teaching and learning committee. Based on the question above, we can give the following hypothesis.

Hypothesis: By using appropriate ML models, we can be able to classify the student grades into possible outcome-codes which includes PCD, RET, MBR and MRNM.

D. Data Collection and Processing

The study makes use of the data that was collected from Wits university. The data is a synthetic data generated which reflects data from wits university. The data contains 14 326 samples and 42 features including the outcome-codes for first year, second year and third year. We then removed features which were not relevant to the study, i.e., third year features like third year outcome-codes, age at third year, etc. The data was cleaned and processed. Some null values and potential outliers were removed. Null values were replaced by the mean. After cleaning the data, we had 8 057 samples. We then selected 6 outcome codes to predict for first year and second year datasets. The data was split into first year data and second year data. For first year data, we selected 26 features from the main data and for second year data we selected 30 features from the main data as informed by the literature. The data was then split into 80% train and 20% test for both first year and second year data.

E. Feature Selection

After diving the data into first year data and second year, we applied mutual information gain(also called entropy) as found in [12] to select k best features to use for predicting student performance. We used the sklearn software when it comes to processing the data and selecting the k best features. For first year data, we selected 15 best features and for second year data, we selected 20 best features. Fig. 1 and Fig. 2 shows the best features which were selected for first year data and second year data respectively.

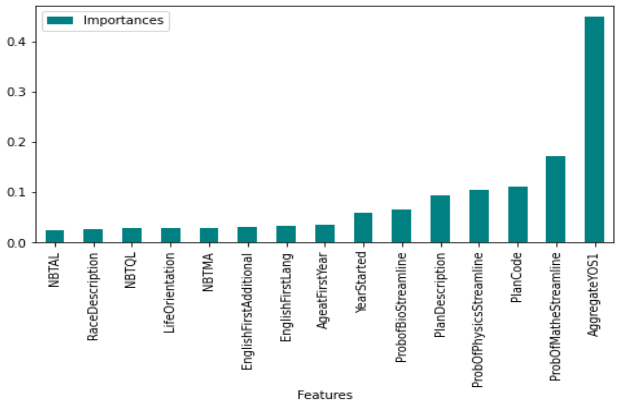

Fig. 1. First year data features

Data used in the study was imbalanced. The class distribution for both first year and second year data was not uniform. For first year data, we had 3 890 for PCD, 1 418 for RET, 949 for MBR, 333 for MBZ, 128 for MFC and 95 for MRNM. For second year data, 2 705 for PCD, 765 for RET, 253 for MBR, 62 for MBZ, 62 for MFC and 42 for MRNM. If we train ML models on this datasets, some ML models will be biased towards the majority class [3]. To solve this problem, we used SMOTE (Synthetic Minority Oversampling Technique)

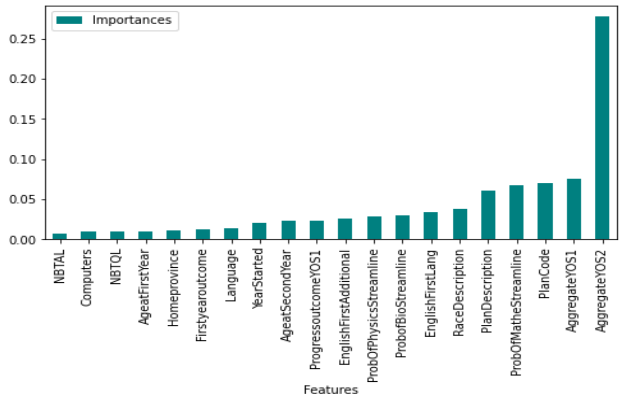

Fig. 2. Second year data features

method [3]. Take note that oversampling was applied on training data not test data.

F. Models

1) Naive Bayes (NB): Naive Bayes is a probabilistic algorithm based on Bayes theorem. It is naive in the sense that each feature makes an equal and independent contribution in determining the probability of the target class [4]. It is easy to build and particularly useful for very large high dimensional datasets [4]. Bayes theorem provides a way of calculating the posterior probability $P(c|x)$ from $P(c)$, $P(x)$ and $P(x|c)$.

2) Logistic Regression (LR): Logistic Regression predicts the probability of the dependable outcome as a function of the independent variables [12]. It is considered as the building blocks of neural networks.

3) Support Vector Machines (SVM): Support Vector Machines is a supervised ML model to solve classification and regression problems. It has demonstrated efficiency in solving a variety of linear and non-linear problems. The idea of SVM lies in creating a hyperplane that distinctly categorizes the data into classes. SVM works well for multi-domain applications with a large dataset; however, the model has a high computational cost [5]. Essentially, it is a constrained optimization problem where the margin is maximized subject to the constraint that it perfectly classifies the data.

4) Decision Trees (DT): Decision Tree is a supervised classification model. We have variants of DT models, i.e., ID3, J48 and C4.5. In DT learning, ID3 is an algorithm used to generate a DT from the data. The decision tree technique involves constructing a tree to model the classification process. Once a tree is built, it is applied to each tuple in the data and results in classification for that tuple [10]. The ID3 algorithm is based on Information Entropy.

5) Multi-Layer Perceptron (MLP): Multi-Layer Perceptron is a supervised learning model that is based on the perceptron. MLP is composed of three or more node layers, including the input/output layer and one or many hidden layers [4]. Input units receives information to be processed, output units where the results of the processing are found, and hidden units learn the non linear patterns. The training phase of MLP consists of adjusting the model parameters (biases and weights) through a back and forth mechanism (feed-forward pass followed by back-forward pass) with respect to the prediction error [4].

6) Random Forest Trees (RFT): Random Forest Tree is an ensemble of Decision Trees bundled together. The training of these bundles of trees consists of executing the bagging process on a data of N entities. This process consists of sampling a set of N training samples with replacement. Then using these samples to train a decision tree. This process needs to be repeated T times. [4]

7) Ensemble Meta-Based Trees (EMT): Ensemble Meta-Based trees is a general meta approach to ML that looks for better predictive performance by combining predictions from multiple models. In this study we used stacking ensemble learning method that runs different ML models on the same data and returns the predicted class that is voted the most or averages the probabilities of the classes. In our case, we used RFT, MLP and Decision trees for EMT.

G. Evaluation Techniques

1) Confusion Matrix: This is a classification model assessment approach that uses an $N \times N$ matrix, with N being the number of needed classes [3]. The machine learning model's projected values are compared to the actual target values. This enables us to comprehend the categorization method's performance and faults [3]. Fig. 3 illustrates the confusion matrix of binary classification.

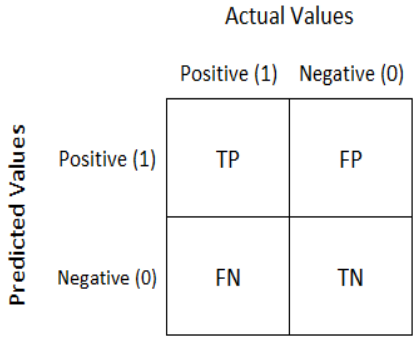

Fig. 3. Confusion Matrix for Binary classification.

2) Accuracy: Accuracy is the percentage of instances which the model predicted them correctly. To calculate the accuracy of the model, we use Eq. 1,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

IV. Results and Discussion

This section provides the results of the models. We used the sklearn software to train the models. We performed 10-fold cross validation on the training data. Random Forest trees is the only model which achieved an accuracy of above 70% on both first and second year datasets. The notebooks used for

experiments can be found on this github repository [13]. Table III shows the accuracies which were achieved by the models.

Models	1^{st} **year**	2^{nd} **year**
DT	63.5%	64.3%
EMT	68.2%	71.7%
LR	61.8%	62.6%
MLP	65.8%	66.6%
NB	53.2%	56.3%
RFT	72%	73.7%
SVC	64.4%	66.2%

TABLE III
ACCURACY OF THE MODELS ON BOTH FIRST AND SECOND YEAR DATA

From the results, we can see that Naive Bayes (NB) was the least performing model when compared to all other models on both first year and second year data. It is followed by Logistic Regression (LR) with an accuracy of 61.8% on first year data and 62.6% on second year data. The best model when compared to all other models is Random Forest (RFT) which achieved an accuracy of 72% on first year data and 73.7% on second year data. The trend of the accuracy is the same on both first and second year data, i.e., we have NB, followed by LR, DT, SVC, MLP, EMT and lastly RFT (from worst to best model). In general the models did not successfully predict the student performance despite Random Forests achieving an accuracy of over 70%.

Fig. 4 and 5 shows the confusion matrix for first and second year data achieved by Naive Bayes. From the confusion matrix, we can see that for all the classes, we have the majority of the samples predicted correctly (diagonally). However for example, for RET, the model did not learn to distinguish between RET and MBR because we have 87 samples misclassified as MBR and 92 samples correctly predicted as RET in first year confusion matrix. The same still applies to second year confusion matrix.

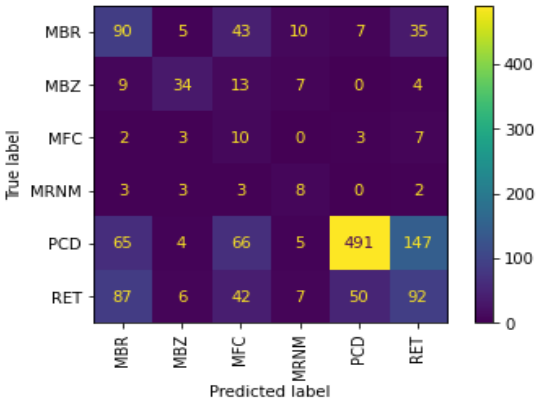

Fig. 4. Naive Bayes confusion matrix for first year data

Fig. 6 and 7 shows the confusion matrix for first and second year data achieved by Random Forests. If we compare the confusion matrix of random forests to that of Naive Bayes, we can see that for random forests, for all classes, the majority

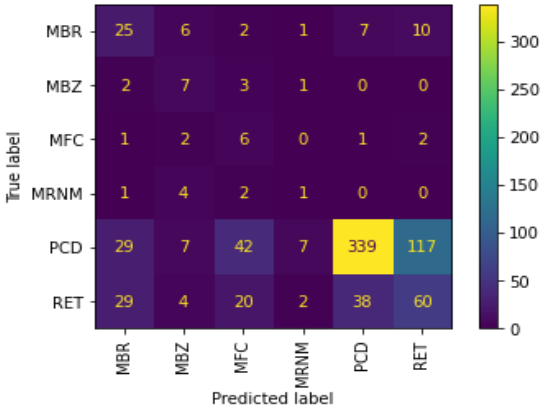

Fig. 5. Naive Bayes confusion matrix for second year

of the samples are predicted correctly (diagonally) better than Naives Bayes. But in this case, when looking at class label RET, we can see that the model managed to distinguish RET and MBR better than Naive Bayes. But the model still finds it hard to distinguish between PCD and RET. It misclassifies 79 samples which are RET as PCD in first year confusion matrix. Random forests is not by chance that it performed better because in the related work, some authors have used the model to predict student performance and it got best accuracy when compared to other models [14], [9], [4]. The rest of the confusion matrices for the other models can be seen in the Appendix section.

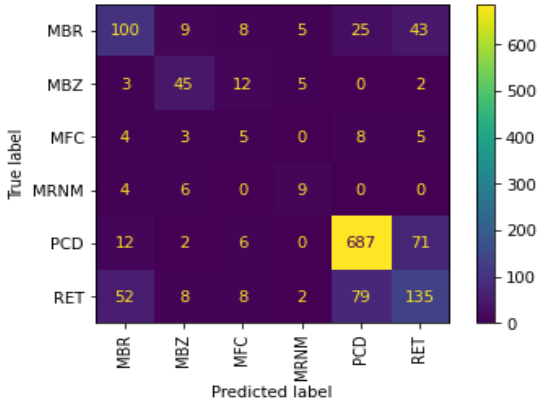

Fig. 6. Random Forests confusion matrix for first year data

Naives Bayes is naive because it makes an assumption that the features are independent when making classification which is not the case in our data and this results in poor performance of the model [3]. Another problem is that of zero frequency when the model sees a sample in test data which it has never seen in training data. It assigns it to a small probability using laplace smoothing which may also be the reason why the model performed poorly. On the other hand, we have Logistic Regression which is the second least performing model. This has to do with the fact that LR requires hand-crafted features. Unlike MLP, it requires features to be

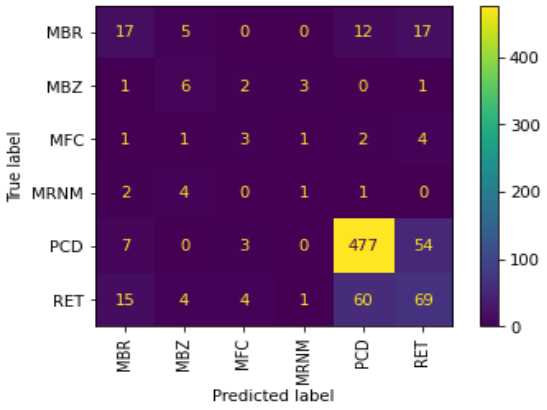

Fig. 7. Random Forests confusion matrix for second year

engineered because it does not learn the features by itself. On the other hand we have EMT which is the second best model which achieved 71.7% on second year data. EMT is a powerful model because it combines different classifiers (in our case its RFT, DT and MLP) and votes which class is predicted the most when compared to other classes or it averages the probabilities achieved by RFT, DT and MLP. Random Forests are robust to outliers since they get averaged out by the aggregation of multiple tree output. It also performs well with non-linear data. There is a low risk of overfitting, as the results are calculated based on the output of multiple decision trees. That is why Random Forest trees achieved better than Decision trees.

The limitations we encountered during the research was that synthetic data may not express the real life systems and observations, which distort the results [14]. Another problem to raise is that the data which we had, all the rows or samples contained null values for at least one feature. We replaced the null values with the mean of their respective columns. But such a replacement can hinder the accuracy of the models. To account for this, domain expertise is required. Class imbalance was one other problem which we encountered. In the Methodology section, we explained that we used SMOTE oversampling technique to increase samples of the minority classes. The ratio between PCD and MRNM classes was too high. So, this means that most of the training samples in MRNM were synthetically generated by SMOTE which also poses another problem.

V. Conclusion

The first goal of the study was to create an auto-tagging system which examines student grades and assigns an outcome-code for first and second year students. The second goal is to Perform classification using some common approaches such as Naive Bayes, Random Forest trees, Support Vector Machines and Multi-Layer Perceptron to predict student performance.

We used synthetically generated data to predict student performance. We used 6 outcome codes which are PCD, MBR, MFC, MRNM, MBZ and RET. We trained Naive Bayes, Support Vector Machines, Ensemble Meta-Based trees, Decision trees, Random Forests, Logistic Regression and Multi-Layer Perceptron. We saw that Random Forests was the best classifier when compared to the other models with an accuracy of above 70% on both first year and second year data. We managed to predict student performance using machine learning models even though the models did not get an accuracy of above 80%.

Future work can include researching ways on how to handle cases of class imbalance and handling null values when it comes to predicting student performance. In the literature I realised that many authors have used machine learning models to predict student performance but I rarely saw authors who tried deep learning methods to predict student performance. Researching the performance of deep neural networks on student success is one of the future works I could consider.

References

[1] H. Bydovska and L. Popelínskỳ, "Predicting student performance in higher education," in *2013 24th International workshop on database and expert systems applications*. IEEE, 2013, pp. 141–145.

[2] A. Almasri, E. Celebi, and R. S. Alkhawaldeh, "Emt: Ensemble meta-based tree model for predicting student performance," *Scientific Programming*, vol. 2019, 2019.

[3] M. Mudau, R. Ajoodha, and K. Padayachee, "Student outcomes prediction by used of machine learning."

[4] G. B. Brahim, "Predicting student performance from online engagement activities using novel statistical features," *Arabian Journal for Science and Engineering*, pp. 1–19, 2022.

[5] A. M. Shahiri, W. Husain *et al.*, "A review on predicting student's performance using data mining techniques," *Procedia Computer Science*, vol. 72, pp. 414–422, 2015.

[6] M. El Zeweidy, E. Osman, and A. P. M. E. Elhennawy, "A comparative analysis of techniques for predicting academic performance," *Journal of the ACS*, vol. 7, 2013.

[7] A. Almasri, R. S. Alkhawaldeh, and E. Çelebi, "Clustering-based emt model for predicting student performance," *Arabian Journal for Science and Engineering*, vol. 45, no. 12, pp. 10 067–10 078, 2020.

[8] H. Mncube, "Predicting student performance using enrollment figures and biographical information."

[9] N. Chettaoui, A. Atia, and M. S. Bouhlel, "Predicting student performance in an embodied learning environment," in *2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*. IEEE, 2021, pp. 1–7.

[10] K. Adhatrao, A. Gaykar, A. Dhawan, R. Jha, and V. Honrao, "Predicting students' performance using id3 and c4.5 classification algorithms," *arXiv preprint arXiv:1310.2071*, 2013.

[11] E. Osmanbegovic and M. Suljic, "Data mining approach for predicting student performance," *Economic Review: Journal of Economics and Business*, vol. 10, no. 1, pp. 3–12, 2012.

[12] L. Bokgoshi, A. Jadhav, and R. Ajoodha, "Predicting students that are at risk of not graduating in record time."

[13] "Github repository for research experiments." [Online]. Available: https://github.com/campbell184/Research-Project.git

[14] S. C. Kubayi, A. Jadhav, and R. Ajoodha, "A machine learning approach for predicting students' second-year outcomes," in *Proceedings of International Conference on Communication and Computational Technologies*. Springer, 2023, pp. 535–547.

VI. Appendix

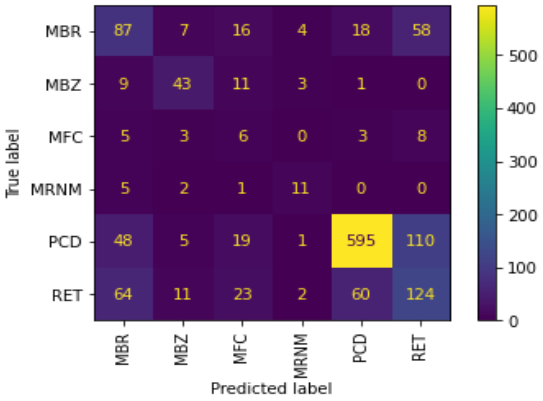

Fig. 8. Decision trees confusion matrix for first year data

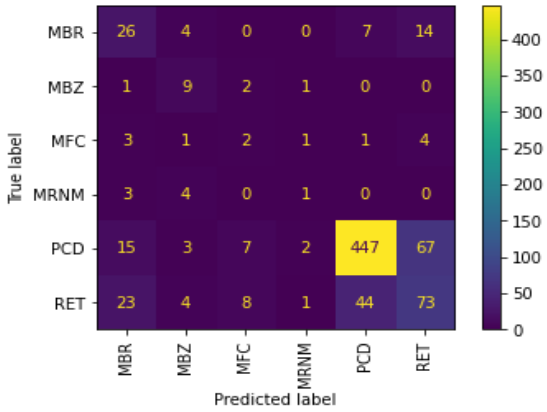

Fig. 11. EMT confusion matrix for second year

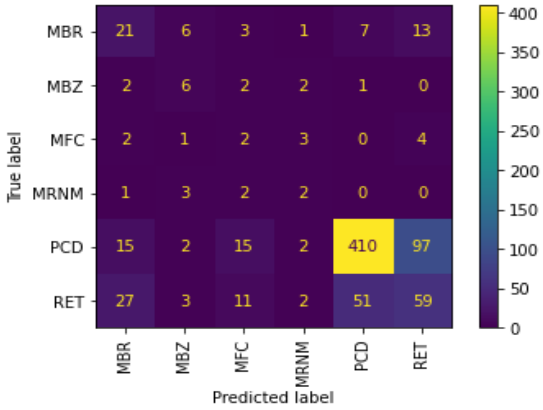

Fig. 9. Decision trees confusion matrix for second year

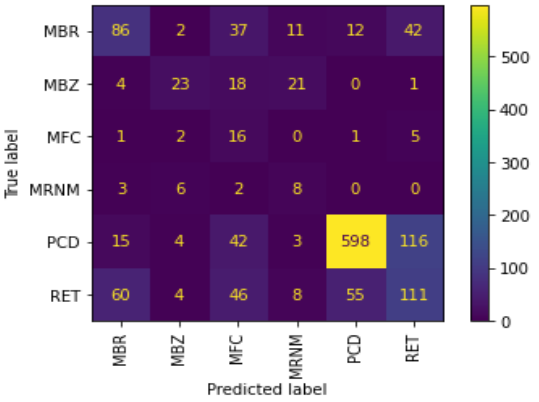

Fig. 12. Logistic Regression confusion matrix for first year data

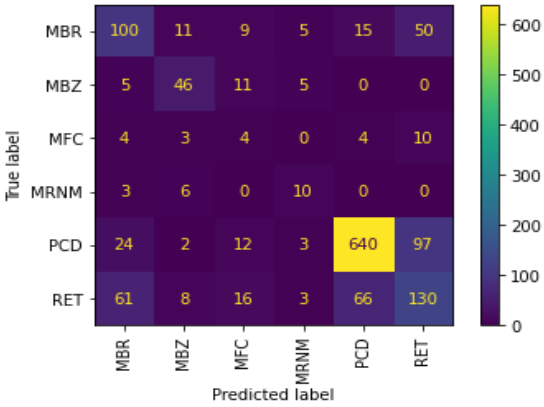

Fig. 10. EMT confusion matrix for first year data

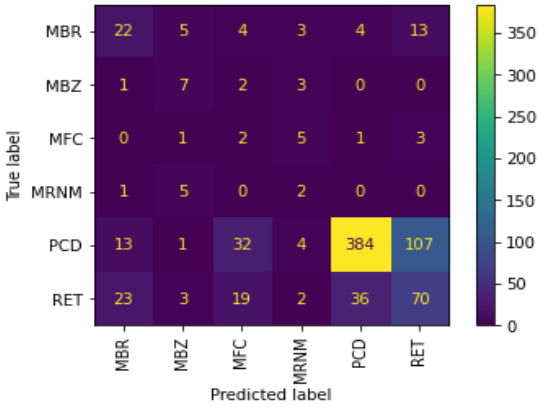

Fig. 13. Logistic Regression confusion matrix for second year

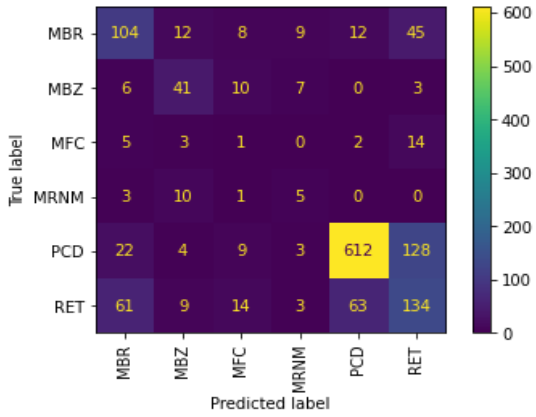

Fig. 14. MLP confusion matrix for first year data

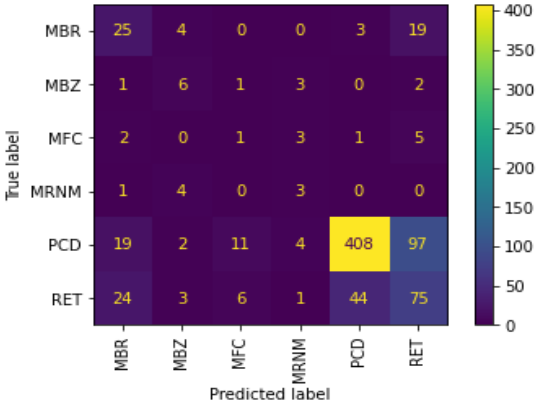

Fig. 15. MLP confusion matrix for second year

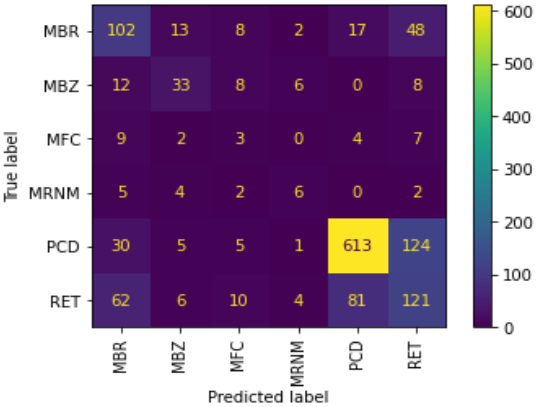

Fig. 16. SVM confusion matrix for first year data

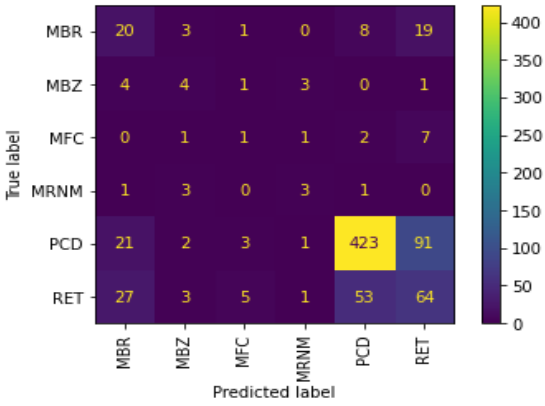

Fig. 17. SVM confusion matrix for second year