Prediction of Student Success by Machine Learning

Melek Ertan
Owaym Khan
Middle East Technical University, Informatics Institute
melekertan95@gmail.com
owaymkhan@gmail.com

Abstract

Education is an important part of our lives and even during pandemic we continued getting the education through online or offline resources. Moreover, the recent advances has been influential in education field and it has become essential for the institutions to adapt to these changes. The aim of this paper is to present the results of a project done by the researchers given about. We have used an public dataset which has been used in different articles. The aim of the project was to answer the questions if it is possible to predict the students' success based on the features collected through Open University (OU) and if the demographics of the students can be used to predict their achievement. The dataset consists of 7 different courses which are presented to the students via online platforms. We used this dataset and tried to predict students' success. The model was trained by using Python 3.0. We have implemented different types of algorithms to see how the performance changes. The algorithms used were Random Forest, Decision Tree, SVM, FeedForward Neural Network, K Nearest Neighbour, Gradient Boosting (XGBoost) and Naïve Bayes. Random forest was the best performing algorithm while the performance of Naïve Bayes was poor. The dataset had many missing values and as we also know the student's success is dependent on many different factors. We hope that further research on student success prediction will contribute to the improvement or development of educational technology systems.

1 Introduction

The advancements in technology have become vital for the survival of many people all around the world. The outbreak of the COVID-19 pandemic also contributed and also forced many people to depend on online systems and technology to carry out a great number of tasks. Education as an indispensable part of our lives, has been affected by this new trend, too. Almost all the students and teachers shifted to online education platforms which they may have never used. Even if online education systems have been around for quite a long time, they were not preferred by the users as much were chosen now. This sudden change in the education required both the students and teachers to adapt to a totally different way of teaching. Thus, it came up with some advantages as well as disadvantages. As we all know, following the progress of the student and detection of failure are the among things the teachers want to do. Online education and machine learning algorithms provides us the opportunity to make use of the online platforms to have an idea about the prediction of student success and failure by using the data collected from online learning tools. Hopefully, these systems will provide important information to improve the quality of education for parents, students, teachers and administrators.

The paper is categorized into four sections including the introduction. The following section is the methodology, and followed by the third section which is the result. Finally, in the following two sections, the result will be discussed and the research would be concluded.

1.1 Background

Our project focuses on datasets provided by Open University to make a prediction of a student's success or failure in a module taken in the university. In this study we implemented and experimented with many techniques on the real-world data. CF techniques are one of the most popular techniques for predicting student's performance (Sarwar et al., 1998) so we decided not to use that technique and instead focused on techniques that had been experimented on a little to none. There were many datasets available for which we could have used for this project but they were either small and less in numbers or they were experimented on too many times that we would add no value to it. Extensive efforts have been made to predict student's performance or grades for different reasons such as; resource allocation, determining low performing students early on, etc. This research aims to predict a student's likelihood to pass or fail a module in the earliest stage possible without taking into consideration the student's number of attempts in a module except the first assessment as there would be ample time for improvement for said student.

Questions that we would try to answer in this research is (a) whether a student's success could be predicted with certain techniques based on few data features of a huge dataset and (b) whether a student's demographics information and the first module assessment datasets can be used to predict a student's performance.

2 Methodology

As we are taking a conducted research as our reference in this project, we have looked at the techniques and methods used (Kuzilek et al., 2017; Kuzilek et al, 2015). The study and many other studies have shown some machine learning algorithms perform better for different reasons. Decision trees are very easy and fast. It can also cope with different loss functions. However, one big disadvantage of the algorithm is that some small changes can lead to huge changes in the hypothesis. Thus, we have also included random forest which mostly seems to work better in classification. We have also included Support Vector Machine (SVM), Feed Forward Neural Network and K Nearest Neighbor, Gradient Boosting (XGBoost) and Naive Bayes. We wanted to use different algorithms so that we can see which one performs better.

2.1 Dataset

As mentioned before, the dataset is taken from an open source which is called Open University (OU). The dataset contains assessment results and logs of interactions with the VLE of 32,593 students of seven different courses on the system. The dataset is stored as 'CVS' files. The materials used in the system is delivered to the students through VLE. Each interaction is stored and recorded by the university. The dataset can be divided into three different data types, respectively: demographics, performance, and learning behavior. Moreover, demographics represent basic information about the student such as name, age, region, and previous education, etc. Moreover, performance refers to student's results and success during their studies. Learning behavior, on the other hand, contains a log of student activities in the VLE. Lastly, the dataset contains data from the years 2013 and 2014. There are some unique values and irrelevant features which will not make any difference in prediction so therefore, they have to be eliminated.

2.2 Tools

We have used python 3.0 to train our model for machine learning part by implementing different techniques. For performance measure we have taken precision, accuracy and F1 scores as our references.

2.3 Preprocessing

As we all know, preprocessing is very significant for machine learning because the way we represent the data will affect our model and the results we have. The features we have included in the dataset will have a minor or a major impact on the training model. As a result, we have examined the data to see what kind of representation would work the best in machine learning. We have discussed with my partner about different types of preprocessing steps and which features to include. Some of the features we had in the data set were unique such as student ID. This feature is unique and it will probably do not have any affect on the student's achievement. Thus, we decided to drop it. The next preprocessing we have implemented was encoding. After dropping the student ID, we had 2 numerical and 35 categorical data. Yet, we need to turn the categorical data into numerical data so that we can use them in our model. We have used different encoding for different data groups because of their properties. Code_module, code_presentation, gender, region, disability sections in the data were encoded with one hot encoding due to the reason that these data points are not ordinal or they do not have any order or priority over other subsections of the data points. The other categorical data types had an order or a certain listing. Therefore, we decided to use ordinal encoding for them. We tried to use standardization for the numerical data points, but they did not have a significant impact on the performance. As a result, we removed scaling from our preprocessing tasks. We have replaced the categorical data in the dataset with their encodings.

2.4 Training the model

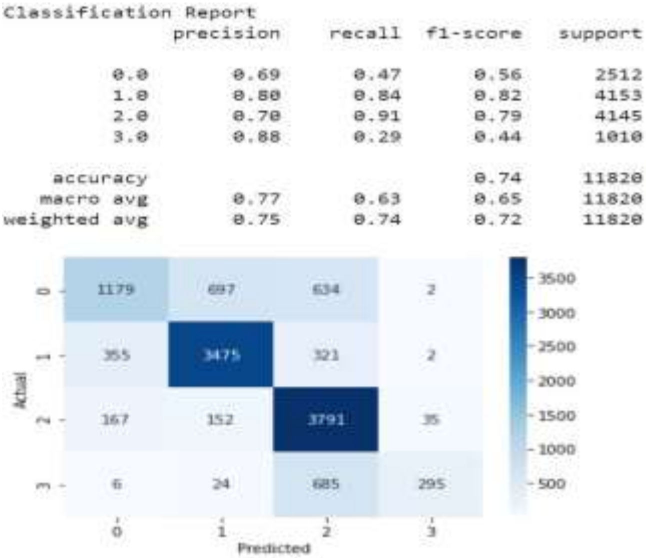
We trained our data by splitting it into train set and test set. The ratio of the train and test set was 70% to 30%. To be able to increase the performance of our model we have used 10 fold cross validation. The aim of cross validation is to use different order of the dataset to see how the model predicts in each. It is highly preferred in machine learning algorithms.

3 Results

In order to gain the best prediction rate, we had to use different techniques namely; Random Forest, Decision Tree, SVM, FeedForward Neural Network, K Nearest Neighbour, Gradient Boosting (XGBoost) and Naïve Bayes. And as mentioned earlier, the ratio of the train and test is set at 70% and 30% respectively. The accuracy of the result can be solidified by the precision and the recall value. Additionally, cross validation has been carried out to prevent under and overfitting and to ensure that the model performs better on data that it has not seen before, making it more generalized. Below we will show the result and it will be followed with a discussion on the results.

3.1 Random Forest

We chose Random Forest due to its high accuracy rate of prediction and it can handle missing values. It also has the power to handle a large dataset with higher dimensionality which is applicable to our project. Below is the result:



```
Classification Report
              precision    recall  f1-score   support

         0.0       0.69      0.47      0.56      2512
         1.0       0.80      0.84      0.82      4153
         2.0       0.70      0.91      0.79      4145
         3.0       0.88      0.29      0.44      1010

    accuracy                           0.74     11820
   macro avg       0.77      0.63      0.65     11820
weighted avg       0.75      0.74      0.72     11820
```

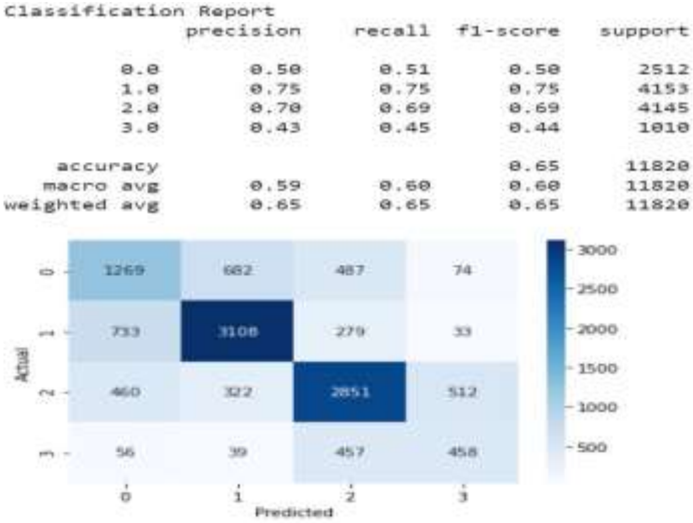
	0	1	2	3
0	1179	697	634	2
1	355	3475	321	2
2	167	152	3791	35
3	6	24	685	295

Fig.1 Random Forest

As can be seen in the graph, the accuracy is at 0.74 or 74% accuracy rate.

3.2 Decision Tree

Decision trees are one of the best forms of learning algorithms which can boost predictive models with accuracy and can be used to solve classification problem. Below is the result:



```
Classification Report
              precision    recall  f1-score   support

         0.0       0.50      0.51      0.50      2512
         1.0       0.75      0.75      0.75      4153
         2.0       0.70      0.69      0.69      4145
         3.0       0.43      0.45      0.44      1010

    accuracy                           0.65     11820
   macro avg       0.59      0.60      0.60     11820
weighted avg       0.65      0.65      0.65     11820
```

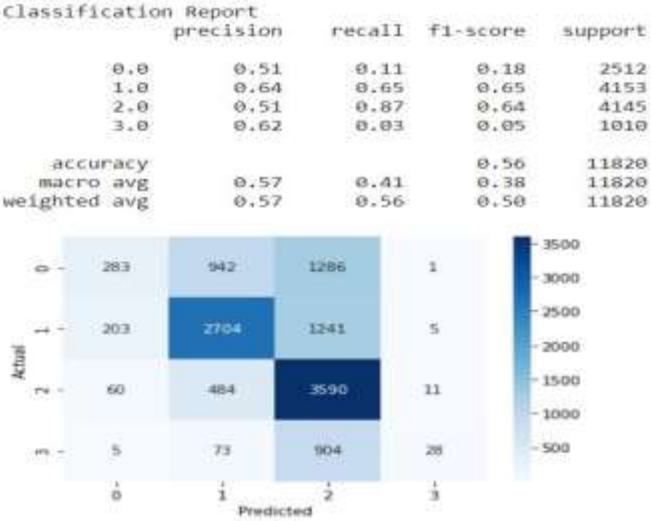
	0	1	2	3
0	1269	682	487	74
1	733	3108	279	33
2	460	322	2851	512
3	56	39	457	458

Fig.2 Decision Tree

The rate of prediction and its accuracy is at 0.65 with this technique. Much lower than the rate predicted by Random Forest.

3.3 SVM

One of the reason we chose SVM is due to its performance and well generalization especially on high dimensional data. Fig 3. shows the result when SVM was implemented on the dataset.



```
Classification Report
              precision    recall  f1-score   support

         0.0       0.51      0.11      0.18      2512
         1.0       0.64      0.65      0.65      4153
         2.0       0.51      0.87      0.64      4145
         3.0       0.62      0.03      0.05      1010

    accuracy                           0.56     11820
   macro avg       0.57      0.41      0.38     11820
weighted avg       0.57      0.56      0.50     11820
```

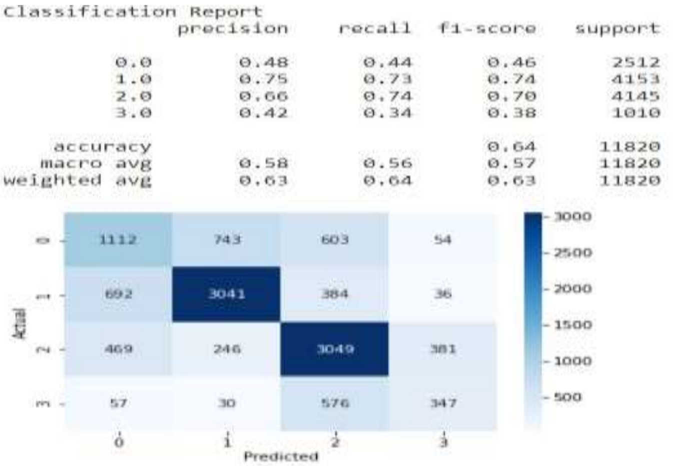
	0	1	2	3
0	283	942	1286	1
1	203	2704	1241	5
2	60	484	3590	11
3	5	73	904	28

Fig.3 SVM

The accuracy is much lower than Random Forest and also Decision Tree, at 0.56 or 56%.

3.4 FeedForward Neural Network

Using this technique, the model has the ability to capture more complex representations or datasets. When a dataset is simple, using this technique can lead to overfitting but this dataset is not simple as it contains complex features and is high dimensional. Below is the result of the accuracy of the prediction.



```
Classification Report
              precision    recall  f1-score   support

         0.0       0.48      0.44      0.46      2512
         1.0       0.75      0.73      0.74      4153
         2.0       0.66      0.74      0.70      4145
         3.0       0.42      0.34      0.38      1010

    accuracy                           0.64     11820
   macro avg       0.58      0.56      0.57     11820
weighted avg       0.63      0.64      0.63     11820
```

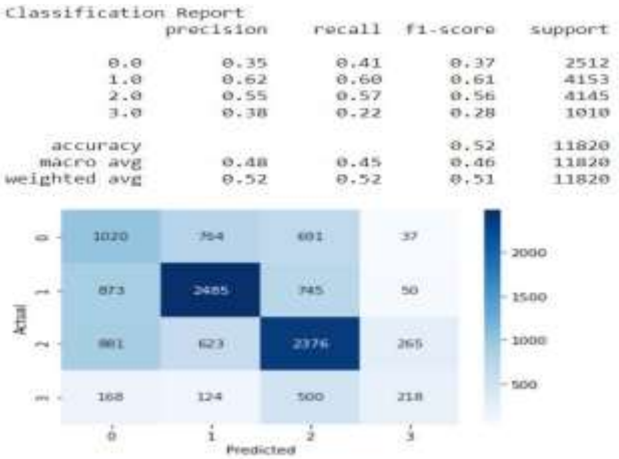
	0	1	2	3
0	1112	743	603	54
1	692	3041	384	36
2	469	246	3049	381
3	57	30	576	347

Fig.4 FeedForward NN

In comparison, the accuracy is at an acceptable level when compared with the previous results. 64% when compared to 65%, 56% and 74% puts it at the third highest in terms of accuracy.

3.5 K Nearest Neighbour

This technique has a drawback and the drawback is especially valid in this project due to the fact that it becomes significantly slower in relation to the size of the dataset. But since many datasets has been removed and only certain features has been selected, this will not affect the speed and additionally, the speed does not affect the accuracy.



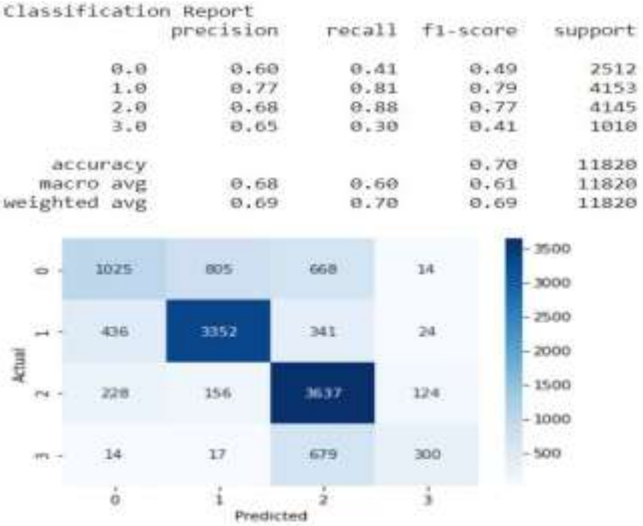
```
Classification Report
              precision    recall  f1-score   support

         0.0       0.35      0.41      0.37      2512
         1.0       0.62      0.60      0.61      4153
         2.0       0.55      0.57      0.56      4145
         3.0       0.38      0.22      0.28      1010

    accuracy                           0.52     11820
   macro avg       0.48      0.45      0.46     11820
weighted avg       0.52      0.52      0.51     11820
```

Fig. 5 K Nearest Neighbour

Fig. 5 shows the accuracy reaching a mere 52% in terms of its accuracy, being the lowest out of all the previous models.

3.6 Gradient Boosting (XGBoost)

This particular technique helps to reduce variance and bias. It delivers high performance and accuracy as compared to other techniques. This particular technique is a regularized version of Gradient Boosting Machine.



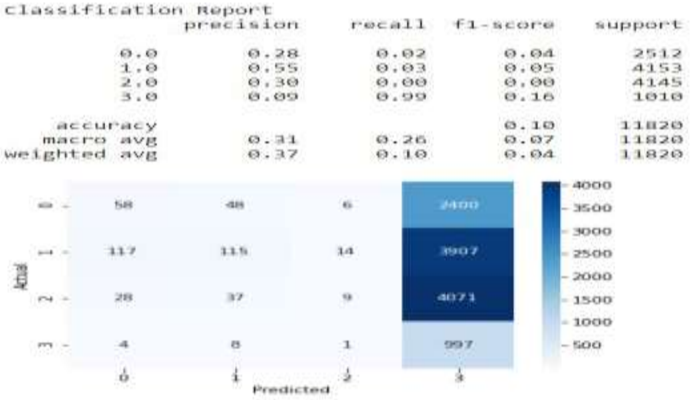
```
Classification Report
              precision    recall  f1-score   support

         0.0       0.60      0.41      0.49      2512
         1.0       0.77      0.81      0.79      4153
         2.0       0.68      0.88      0.77      4145
         3.0       0.65      0.30      0.41      1010

    accuracy                           0.70     11820
   macro avg       0.68      0.60      0.61     11820
weighted avg       0.69      0.70      0.69     11820
```

Fig. 6 Gradient Boosting

This technique yields a 70% accuracy rate, which makes this technique highly accurate when compared with previous results.

3.7 Naive Bayes

Naive Bayes is suitable for multi-class prediction problems and would have been suitable for this project especially, but on the other hand the probability output of Naive Bayes algorithm should not be taken too seriously due to its assumption of independent predictors.



```
Classification Report
              precision    recall  f1-score   support

         0.0       0.28      0.02      0.04      2512
         1.0       0.55      0.03      0.05      4153
         2.0       0.30      0.00      0.00      4145
         3.0       0.09      0.99      0.16      1010

    accuracy                           0.10     11820
   macro avg       0.31      0.26      0.07     11820
weighted avg       0.37      0.10      0.04     11820
```

Fig. 7 Naive Bayes

Yielding a mere 10% puts this predictor model at the lowest ranking in result-wise as it performed terribly due to the complex approach required for this research and dataset.

4 Discussion

In this section we will discuss the findings given in the results. The algorithms which were implemented on the dataset before were Bayesian classifier, Classification and regression tree (CART), k Nearest Neighbours (k-NN), and k-NN with VLE. Different from the algorithms given we have tried to implement Random Forest, Decision Trees, SVM, Gradient Boosting and Feedforward Neural Network. In a similar study by Huang (2013), Naïve Bayes performed the best among the algorithms that they have used which included SVM. In the light of this study, we expected Naïve Bayes to perform relatively better than the result we had in our dataset. This actually gives us the idea that even if the task and the dataset is similar, the nature of the dataset and the data points might lead to very different results even if we use a similar type of algorithm. Therefore, it is very important to decide which features to use in the training. The representation of the data, optimization and the generalization has strong connections and a small change in one of these affects the others performance. The result of the random forest and decision trees algorithm was not surprising for us. Because of the randomness nature of the random forest algorithm, it performs better than decision trees.

The second most accurate technique used was XGBoost. Gradient boost is also an ensemble method. It also tries to solve the slower computation problem by adding parallel computation into the model. This method also performed very well as the second best algorithm for our dataset.

When it comes to SVM and its performance, the accuracy of the algorithm was 0.56. A few of the possible reasons for its lower performance compared to Random Forest and XGBoost is because SVM is known to perform poor in large dataset and when there is noise in the data. There is also no clear cut separation between the classes since this is a student success prediction which can be affected by many factors.

On the other hand, there are few reasons to keep in mind in terms of the overall results we have in our model. Including the assessment results into the prediction algorithm helps us to predict the success better, but it can be also seen as redundant for a machine learning model for the following reasons. The tutor, student, administrator, or parents can also make an easy prediction on the student performance by looking at the previous grades. This does not seem like a complex task. The only advantage of creating a machine learning system for this type of prediction can be done due to its speed and capability of handling larger dataset in a shorter amount of time. For systems such as online education platform which tries to serve for great deal of numbers of students with a customized manner, these models can be effective. However, in an institution level where there is face to face education, they may not be preferred. Another important point that we need to consider is that student performance depends on many different factors and individual differences as well as their attitude and motivation towards learning can affect their success in a taken course as well as the type of the course. The dataset deals with different types of courses and distinct students. Thus, it may not be very easy or accurate to predict their performance on the course since too many factors play small or big role in the education process. If our data had more about the students' previous success in similar courses etc. we could have made a better prediction for their success in a given course.

5 Conclusion

All in all, in this project we have tried to use an open source dataset to predict students' success. The dataset was large and it has been used in different studies. The results showed that Random Forest performed the best and Gradient Boost and Decision Trees followed Random Forest respectively in terms of their accuracy rates. Especially Naïve Bayes performed very poorly with its accuracy of 10%.

Further research can be conducted in the light of the results given for the model by using different classifiers as well as regression models. Since online education is becoming popular day by day, such machine learning model will gain importance in our world. These models will also help institutions, teachers, parents and students in terms of keeping the track of the students' improvement throughout the year as well as preventing possible failures.

References

(Sarwar et al., 1998) Sarwar, B. M., Konstan, J. A., Borchers, A., Herlocker, J., Miller, B., and Riedl, J. 1998. Using filtering agents to improve prediction quality in the grouplens research collaborative filtering system. In Proceedings of the 1998 ACM conference on Computer supported cooperative work. ACM, 345–354.

(Kuzilek et al., 2017) Kuzilek, J., Hlosta, M. and Zdrahal, Z., 2017. Open university learning analytics dataset. *Scientific data*, *4*(1), pp.1-8.

(Kuzilek et al., 2015) Kuzilek, J., Hlosta, M., Herrmannova, D., Zdrahal, Z., Vaclavek, J. and Wolff, A., 2015. OU Analyse: analysing at-risk students at The Open University. *Learning Analytics Review*, pp.1-16.