Student Dropout Prediction During COVID-19

Sadnam Saniat
Dept. of CSE
Daffodil International University
Dhaka, Bangladesh
sadnam15-2794@diu.edu.bd

Arafat Rahman Medul
Dept. of CSE
Daffodil International University
Dhaka, Bangladesh
arafat15-1073@diu.edu.bd

MD. Omer Faruk Tusher
Dept. of CSE
Daffodil International University
Dhaka, Bangladesh
omer15-13122@diu.edu.bd

Abstract –

This study analyzes the performance of seven machine learning algorithms with different perspectives for defining data files, in the prediction of Bangladeshi student desertion. The algorithms used were: DecisionTreeClassifier, RandomForestClassifier, ExtraTreesClassifier, AdaBoostClassifier, MLPClassifier, Bagging Classifier and GaussianNB. It was found that the ExtraTreesClassifier algorithm with 05 variables randomly sampled as candidates in each division, was the best for predicting dropouts. . In a first validation sample, this approach correctly accuracy 91%.

Keyword— Dropout, students, covid-19, problems, machine learning.

I. INTRODUCTION

It is imperative to analyze the phenomenon of desertion in educational institute. According to, dropout rates become a matter of concern in education institutions and education authorities during covid-19, given that dropout rates increase social and economic gaps between social groups, and hinder the development of countries. This has led to the development of research that is looking for algorithms for predicting students dropout rates, whose situation can guide the design of support programs that will increase student retention. The most recent research in this area have focused on accurately predicting the risk of students dropping out during this covid-19 situation. Several machine learning algorithms have been used in these studies, including DecisionTreeClassifier; RandomForestClassifier; ExtraTreesClassifier; AdaBoostClassifier; MLPClassifier; BaggingClassifier; GaussianNB; In general, these techniques allow the identification of patterns and associations between the variables included in the algorithms, and dropping out. As emphasized in, a static approach to identifying students at risk is not advisable; it is therefore necessary to use dynamic models and test these using data from this current pandemic situation. Several types of variables have been used in this type of research as predictors of dropout, such as those related to financial problem, internet problem, covid-19 issue, shortage of electronics device, method of study and so on. The combination of these variables has provided important information about individual and institutional factors that increase the probability of dropping out. The present investigation, just as those mentioned previously, uses several machine learning algorithms to train an algorithm to predict student dropouts.

1. The way in which the non-dropout is defined. One issue when trying to predict future dropouts is whether active students should be included in the group of non-dropouts. Including them can add noise to the training and prediction of the algorithm, since it is not known if they will stop their study in the future. An alternative is to use only those who have completed study as non-dropouts, excluding active students. Both of these approaches are analyzed in this study.
2. The present investigation is that most studies have focused on predicting those who will drop out at the beginning stage of the pandemic situation. While the present research attempts to predict who will stop enrolling in the future. The objective is to train an algorithm that can be used to determine which students will drop out (stop enrolling in this time range), in order to take steps to address these dropouts before they occur.
3. During covid-19 situation, many types of problems could arise, our model covered most of the major problems but which also can cause dropout that cannot covered by our model.
4. We want to reduce future dropout students by predicting recent dropout student’s data. For that reason, we have to analyze recent major problems which is responsible for massive dropout students.

The main purpose of our model is to predict dropout students for reduce future dropout students.

II. METHODOLOGY

A. Data

The sample is composed of all those students who are studying in School, College and University. There were 300 records, corresponding to the School, College and University students, which initially met this criterion. Records that were incomplete or contained incorrect information were deleted, resulting in a final sample of 294 records of students. Four perspectives were used to predict dropouts.

B. Variables

There are 5 variables that are used to define a Dropout or Non Dropout. In the first variable, a student who has study at School, College and University. In the second variable, gender is used because in dropout gender play a huge rule. In third variable, we use are area, because in rural area corona effects much more than city. In fourth variable, Reason for dropout we use. In the fifth variable we use how much of them are suitable with online Study.

1. Dividing the data: For predictive purposes the data must be divided into at least two parts, one to train the model and the other to evaluate its predictive capacity. The first part was made up of data for 80% of the subjects, chosen at random, and the second part by the remaining 20%.
2. Training: The parameters of the seven methods were estimated in this phase. Using this library, the models are calibrated by evaluating different parameters in each method by default. To determine which parameters contribute to better prediction, at first we use one-hot-encoding then separate dummies variable then covered to numerical value after that smote to test prediction. This procedure consists of taking 80% of the sample to estimate the algorithm with specific parameters and 20%.

ExtraaTreeClassifier:

$$\text{Information Gain} = \text{Entropy}(S) - [(\text{Weighted Avg}) * \text{Entropy (each feature)}]$$

Entropy: Entropy is a metric to measure the impurity in a given attribute. It specifies randomness in data. Entropy can be calculated as:

$$\text{Entropy}(s) = -P(\text{yes}) \log 2$$
$$P(\text{yes}) - P(\text{no}) \log 2 \, P(\text{no})$$

Where,

0 **S= Total number of samples** 0 **P (yes) = probability of yes** 0 **P (no) = probability of no**

Gini Index,

$$\text{Gini Index} = 1 - \sum_j P_j^2$$

A too-large tree increases the risk of overfitting, and a small tree may not capture all the important features of the dataset. Therefore, a technique that decreases the size of the learning tree without reducing accuracy is known as Pruning. There are mainly two types of tree **pruning** technology used:

0 Cost Complexity Pruning 0 Reduced Error Pruning.

In Random Forest the parameter tested was mtry. This is the number of variables randomly sampled as candidates in each division.

The training algorithm for random forests applies the general technique of bootstrap aggregating, or bagging, to tree learners. Given a training set $X = x_1, ..., x_n$ with responses $Y = y_1, ..., y_n$, bagging repeatedly (B times) selects a random sample with replacement of the training set and fits trees to these samples:

For $b = 1... B$:

1. Sample, with replacement, n training examples from X, Y; call these Xb, Yb.
2. Train a classification or regression tree fb on Xb, Yb.

After training, predictions for unseen samples x' can be made by averaging the predictions from all the individual regression trees on x':

Or by taking the majority vote in the case of classification trees.

This bootstrapping procedure leads to better model performance because it decreases the variance of the model, without increasing the bias. This means that while the predictions of a single tree are highly sensitive to noise in its training set, the average of many trees is not, as long as the trees are not correlated. Simply training many trees on a single training set would give strongly correlated trees (or even the same tree many times, if the training algorithm is deterministic); bootstrap sampling is a way of decorrelating the trees by showing them different training sets.

Additionally, an estimate of the uncertainty of the prediction can be made as the standard deviation of the predictions from all the individual regression trees on x':

The number of samples/trees, B, is a free parameter. Typically, a few hundred to several thousand trees are used, depending on the size and nature of the training set. An optimal number of trees B can be found using cross-validation, or by observing the out-of-bag error: the mean prediction error on each training sample x_i, using only the trees that did not have x_i in their bootstrap sample. The training and test error tend to level off after some number of trees have been fit.

3. Validation: In this step, the predictive capacity of the estimated seven algorithm was evaluated.

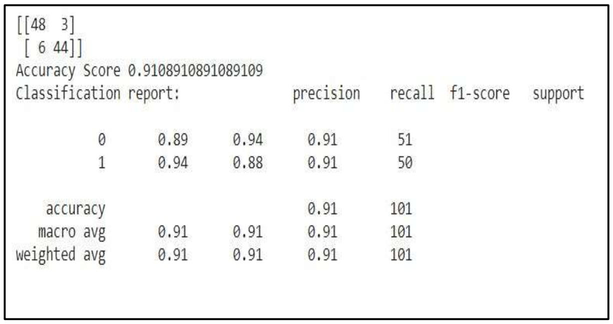
Subsequently, adjustment indicators were estimated: the probability of correctly detecting dropouts, the probability of correctly detecting non-dropouts. The probability of correctly detecting no dropouts and the specificity are included simply for informative purposes, since the best model is the one that simultaneously maximizes the probability of correctly detecting dropouts.

III. RESULT AND ANALYSIS

Tables 1, 2, 3, 4, 5, 6, 7 present the performance metrics for the seven algorithms with the parameters that generated the highest prediction value in each of the seven perspectives.

In Table 1, 3, 5 it can be seen that prediction values in the three algorithms are high. The best algorithms are the Random Forest and Extra Tree Classifier. Which has the best performance. Where accuracy score 91%.

Table 1

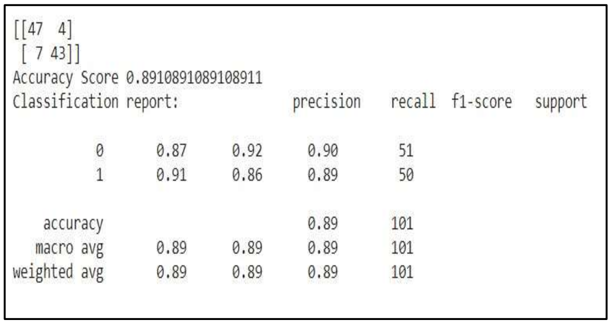


```
[[48  3]
 [ 6 44]]
Accuracy Score 0.9108910891089109
Classification report:             precision    recall  f1-score   support

           0       0.89      0.94      0.91        51
           1       0.94      0.88      0.91        50

    accuracy                           0.91       101
   macro avg       0.91      0.91      0.91       101
weighted avg       0.91      0.91      0.91       101
```

Table 2

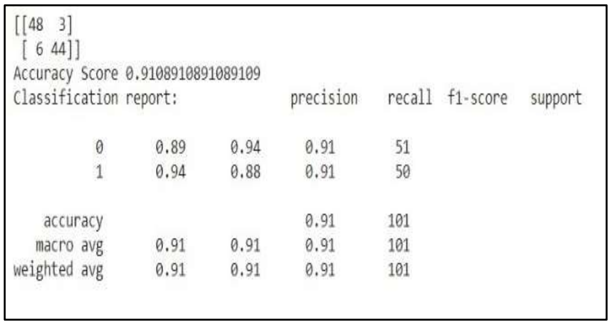


```
[[47  4]
 [ 7 43]]
Accuracy Score 0.8910891089108911
Classification report:             precision    recall  f1-score   support

           0       0.87      0.92      0.90        51
           1       0.91      0.86      0.89        50

    accuracy                           0.89       101
   macro avg       0.89      0.89      0.89       101
weighted avg       0.89      0.89      0.89       101
```

Table 3

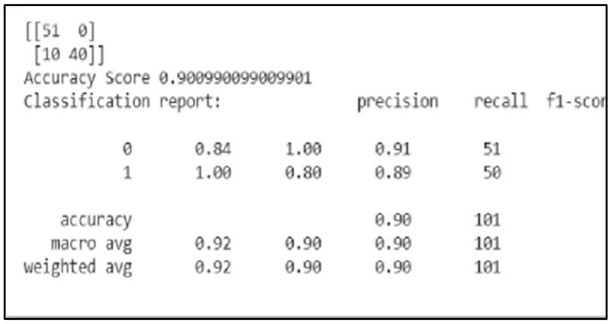


```
[[48  3]
 [ 6 44]]
Accuracy Score 0.9108910891089109
Classification report:             precision    recall  f1-score   support

           0       0.89      0.94      0.91        51
           1       0.94      0.88      0.91        50

    accuracy                           0.91       101
   macro avg       0.91      0.91      0.91       101
weighted avg       0.91      0.91      0.91       101
```

Table 4

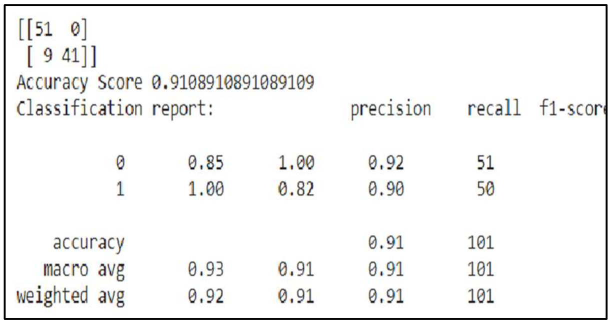


```
[[51  0]
 [10 40]]
Accuracy Score 0.900990099009901
Classification report:              precision    recall  f1-scor

           0       0.84      1.00      0.91        51
           1       1.00      0.80      0.89        50

    accuracy                           0.90       101
   macro avg       0.92      0.90      0.90       101
weighted avg       0.92      0.90      0.90       101
```

Table 5

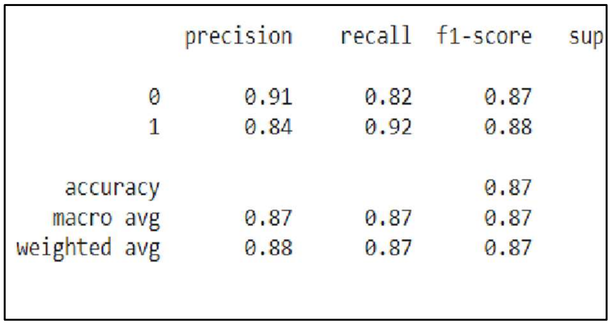


```
[[51  0]
 [ 9 41]]
Accuracy Score 0.9108910891089109
Classification report:              precision    recall  f1-score

           0       0.85      1.00      0.92        51
           1       1.00      0.82      0.90        50

    accuracy                           0.91       101
   macro avg       0.93      0.91      0.91       101
weighted avg       0.92      0.91      0.91       101
```

Table 6

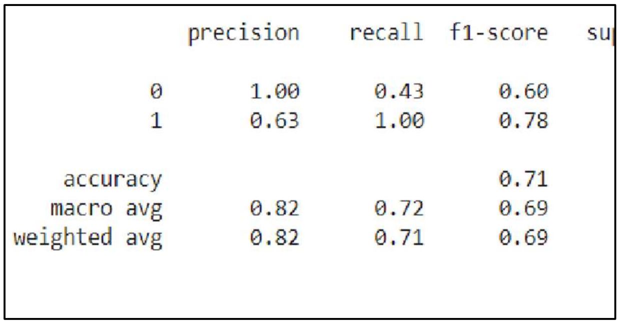


```
              precision    recall  f1-score   sup

           0       0.91      0.82      0.87
           1       0.84      0.92      0.88

    accuracy                           0.87
   macro avg       0.87      0.87      0.87
weighted avg       0.88      0.87      0.87
```

Table 7



```
              precision    recall  f1-score   su

           0       1.00      0.43      0.60
           1       0.63      1.00      0.78

    accuracy                           0.71
   macro avg       0.82      0.72      0.69
weighted avg       0.82      0.71      0.69
```

Survey Results

1. **Reasons for Enrolling in an Online Program**
   Students enrolled in the HRE Online program because of the flexibility of schedule, the convenience and effectiveness of taking online classes, the good fit with their goals, for professional development, to obtain an advanced degree in the field, and also because of the strong reputation of the University of Illinois.

2. **Reasons for Leaving an Online Program** Students reported leaving the online program for a variety of reasons. There did not appear to be a dominant reason for dropping out of the program. Their reasons for leaving the program were organized into personal, job-related, and program-related reasons:

   a. Personal Reasons Financial difficulties or the long-term financial investment not worth the benefit Lack of time to complete the assignments, which took more time compared to traditional courses Schedule conflicts Family problems

   b. Program-related Reasons Too many low level assignments Too difficult working on the group assignments Lack of one-to-one interaction with the instructors and students The academic program was too difficult/demanding Lack of interest in the material or the program didn't meet expectations

   c. Technology-related Reasons The learning environment was too de-personalized Not enough support from the technical staff The technology overwhelmed the content Lack of technical preparation for the program.[1]

IV. CONCLUSION

After analyzing the results, it was determined that the best algorithm for classifying dropouts is the Random Forest. The ideal perspective for building the algorithm is to use information for students' situation.

In addition to yielding the best metrics, the Random Forest with the perspective discussed previously shows a smaller gap between these metrics, and more appropriate behavior through time.

The results also suggest that, to train the dropout prediction algorithm, it is convenient to exclude active students, who may add noise because it is not known beforehand if they will dropout or graduate in the future. In essence, the problem is that they may have a dropout pattern, but they have not been classified as such. However, these research focused only on predicting those students who stopped studying in corona pandemic.

V. REFERENCES

[1]. **Pedro A. Willging & Scott D. Johnson.** *FACTORS THAT INFLUENCE STUDENTS' DECISION TO DROPOUT OF ONLINE COURSES*, Illinois, Journal of Asynchronous Learning Networks, v13 n3 p115-127, Oct 2009.