ROMANIAN JOURNAL OF INFORMATION SCIENCE AND TECHNOLOGY
Volume **26**, Number 2, 2023, 121–136

Student Success Prediction Using Feedforward Neural Networks

Kamil YURTKAN[1,5], Ahmet ADALIER[2], and Umut TEKGÜÇ[3,4]

[1]Computer Engineering Department, Faculty of Engineering, Cyprus International University, Nicosia, Northern Cyprus via Mersin10 Turkey
[2]Computer Education and Instructional Technology Department, Faculty of Education, Cyprus International University, Nicosia, Northern Cyprus via Mersin10 Turkey
[3]Computer Programming Department, Vocational School, Baheehir Cyprus University, Nicosia, Northern Cyprus via Mersin10 Turkey
[4]Blockchain Technologies Application and Research Center, Baheehir Cyprus University, Nicosia, Northern Cyprus via Mersin10 Turkey
[5]Artificial Intelligence Application and Research Center, Cyprus International University, Nicosia, North Cyprus, Turkey
[6] E-mails: :
kyurtkan@ciu.edu.tr*, aadalier@ciu.edu.tr, umut.tekguc@baucyprus.edu.tr
* Corresponding author

Abstract. Machine learning algorithms have been used in the last decade to predict human behavior. In education, the student's behavior, and accordingly, their success prediction is also applicable in parallel with the developments in machine learning algorithms and the increased availability of the datasets. The datasets include the observations, which the machine can learn to predict student behavior. By this analysis, given the background information about a student, the features representing a student sample, and the student's possible performance may be estimated. This study's motivation is to predict a student's possible performance to give guiding service. This paper proposes a novel approach for predicting student success by using conventional feed-forward neural networks. The algorithm selects the most informative features based on the variances and uses those features to represent a student sample. The approach is tested on the Experience-API (X-API) dataset collected from Kalboard 360 e-learning system. There are 480 samples in total, with 16 features. It is shown that the improved approach achieves comparable results around 91.95% acceptable predictions by only using behavioral attributes and 93.17% acceptable prediction rates without the feature selection process, respectively.

Key-words: Data mining and analysis; educational data mining; feature selection; feed-forward neural networks; information content; natural computing; pattern recognition; student performance; student success prediction; variance.

1. Introduction

Student success prediction has been a well-studied topic in the literature in the last decade [1, 2]. These studies' main motivation is to monitor student behavior and provide accurate guidance to students in their education.

In the literature, there are improvements in prediction methods in parallel with machine learning methods' growth [3–7]. In most studies, the proposed methods seek the student attributes that can robustly identify student success. In the machine learning aspect, the feature extraction and selection parts of the systems are developed in order to find the best student features affecting their success. Although, studies were started in the early 1990s, the majority of the machine learning methods are recently proposed methods [8]. Significant examples of machine learning approaches are also given in [9–10] and [11].

Previous works define student performance as a metric of achievements in future academic events [12]. Indeed, evaluating students' academic performance has been a significant target in education in order to overcome the issues such as low performance, increased number of unsuccessful students that end with dropouts, and unexpected extensions on graduation time [8]. Student performances' predictions are mainly predicted by using previous grades and current coursework assessments [13]. However, recent works explored non-academic factors' influence on student achievements [13, 16]. These findings indicate that several factors and their combinations influence student performances. Therefore, a precise prediction requires the improvements of sophisticated methods.

Student success prediction studies accelerate after 2000s. One significant study is given in [17], where engineering students' performance is predicted using a Decision Tree. The sample size was 340, and the data were collected from their first-year exams. Their proposed model was able to achieve 60% correct predictions. In related studies given in [4–5], [18–19] and [20], the data mining is done for higher education institutions and high schools. In [21], we see another important study using final year students' marks through two different datasets. There was only one common attribute between the datasets. The majority of the students were in the last four semesters in a college. The author of [22] analyzed past research findings on students' performance prediction and assessment by applying various data mining methods. In [23], the authors achieved student performance prediction using a Decision Tree and used a classifier's neural network. The produced outcome was acceptable based on various traits to predict the student performances.

Moreover, most of the applications are focused on predicting future course grades only, as presented in [24–25], and [26]. For example, in [27], the performances of the fifteen different algorithms on predicting the students' cumulative grade point averages are analyzed. Besides studies focusing on predicting the student requirements during education [28], on the other hand, the lack of analysis associating grade predictions with the leading key factors is still a challenge [8]. In this kind of attempt, the students' key factors are analyzed that are significantly affecting the academic performance. In [29], a related study is given in which an optimized multi-label classifier predicts the weights for the influence of key factors affecting student performances is proposed.

The attention of the researchers has grown to the topic of student success prediction as a multidisciplinary research topic and significant studies completed very recently, including the studies given in [1–3] and [30].

Overall, the significant studies in the field achieved acceptable rates; however, the features used were relatively not easy to extract. Thus, achieving acceptable student performance rates using simple and robust features is still challenging.

The main motivation of this paper is to find out the most informative features for a robust prediction system. Also, the main motivation of student success prediction systems is to predict the student success within the semester and perform actions necessary to improve the success rates. In this way, student performances can be improved. On the other hand, if there is any student that is going to be dropout because of low success rates within the semesters, this kind of prediction system will assist the management to overcome and attain the student again.

In this paper, a novel feature extraction approach is presented, resulting in selecting the students' behavioral features to predict their success in acceptable ranges. *Feed-forward neural network* (FFNN) is trained based on the features; that is, FFNN is a well-known universal function approximator expected to learn and accurately represent student distribution functions [31].

The main contribution of the paper is the approach of variance-based feature selection procedure that is applied to student success prediction problem. Moreover, the feature selection procedure outcomes the behavioral features. We believe that this is a positive addition of the paper to the literature that the most informative features in the student success prediction problem are found to be the behavioral features of the students like how many times they participate in the class work by raising their hands. This is important in another aspect that extracting all the features from the students can be challenging in practice. Thus, using the minimum number of features to perform student success prediction yields in a practically applicable approach.

The paper includes the literature review part in the introduction section. Then, the methodology is explained with Data and Pre-processing and feature selection sub-sections. The experimental setup and the results are discussed in a separate section. The discussions are followed after the presentations of the experimental results. Finally, the paper concludes with conclusions and future works.

2. Methodology

The approach applied can be considered in two main parts: the feature selection and classification. As a classifier, the Feed Forward Neural Networks (FFNN) is employed with two hidden layers. The FFNN proved its capability on numerous problems and is known as a universal function approximator [32]. The overall block diagram followed by the pseudocode explaining the steps applied are shown in Figure 1.

The students' success prediction is made using FFNN neural networks [32]. The network structure and the the learning curve are shown in Figure 2. The learning algorithm that is the conventional backpropagation algorithm that is explained in the equations below.

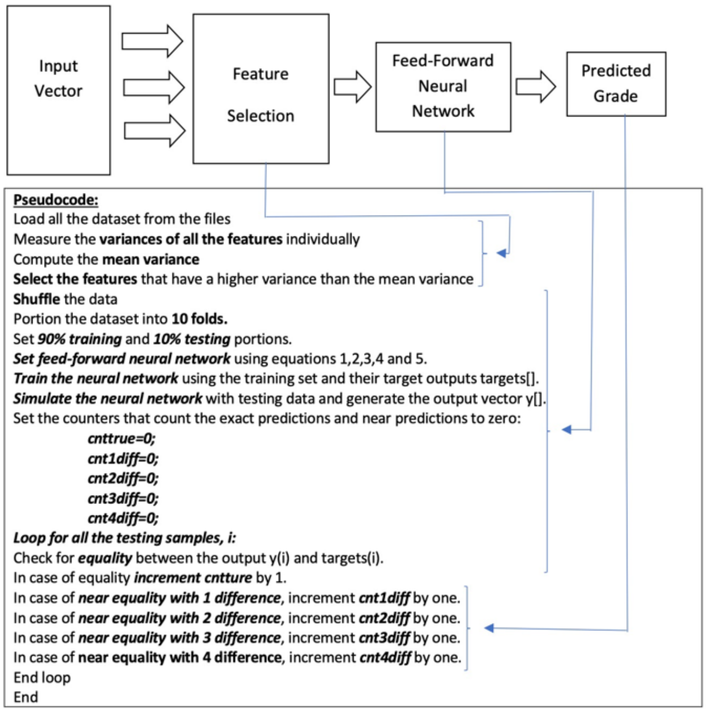

Fig. 1. Block diagram and the pseudocode of the proposed approach.

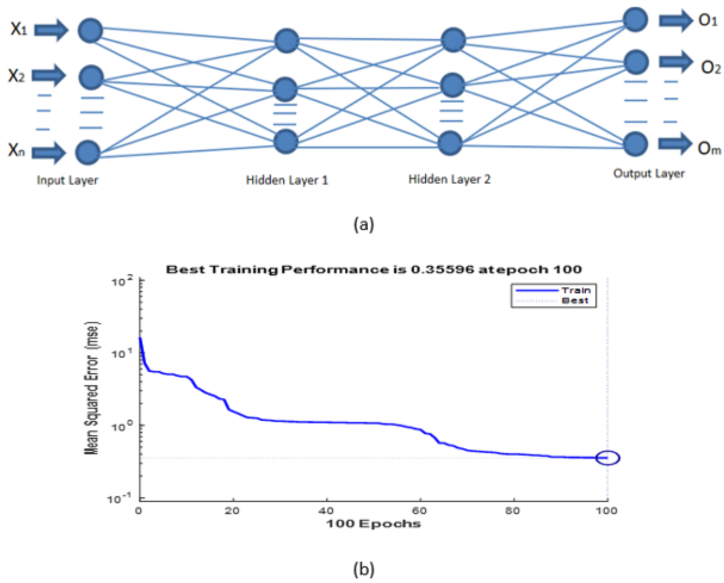

Fig. 2. Feed-Forward Neural Network classifier with two hidden layers employed (a) and the training performance graph (b). n is the feature vector's dimension equal to 17 before feature selection and 4 after the feature selection. m denotes the number of output units that is 8.

Considering the dataset having (x_i, y_i), pairs where x_i is the input and y_i is the desired output of the neural network for the input x_i. With the size N, it is denoted by $X = (x_1, y_1), , (x_N, y_N)$.

A feedforward neural network runs a backpropagation algorithm to adjust the weights to satisfy input-output relations. Considering a feed-forward neural networks whose parameters are denoted by θ, the following parameters are of interest. The weight between node j in layer l_k and node i in layer l_{k1} is denoted by $w^k ij$. Another parameter is the bias (threshold) that is b_i^k, representing the bias for node i in layer l_k.

$E(X, \theta)$, is the error function that defines the error between the desired output yi and the actual calculated output yi' . This is for the neural network on input xi. The error function is calculated for a set of input-output pairs (x_i, y_i), where $(x_i, y_i) \in X$ and the parameters θ.

Backpropagation training requires gradient descent that the calculation of the gradient of the error function $E(X, \theta)$. This calculation involves the current weights $w^k ij$ and biases b_i^k. Then, also using the learning rate α, that is a relatively small (between 0 and 1) step size, every iteration of gradient descent updates the weights and biases according to equation 1 as follows.

$$\theta^{t+1} = \theta^t - \alpha \frac{\partial E(X, \theta^t)}{\partial \theta} \tag{1}$$

Obviously, if the desired output and the actual calculated output are the same, there is no weight update. Here, θ^t denotes the parameters of the neural network at iteration t in gradient descent [33, 34].

3. Dataset

The dataset used in our study is the X-API dataset described in [35], the Experience-API (X-API) dataset collected from Kalboard 360 e-learning system. The student performances are obtained based on two courses, and eight different grades are reported. The samples include a total of 480 individual student attributes. The attribute list is given in Table 1.

As shown in Table 1, there are 17 attributes in total. 16 of those are used for representing a sample, and one student grade is used as the target. A sample represents a student's extracted features for the selected course. GradeID field is the target field in this list. Therefore, the samples are categorized according to their GradeID's. There are ten different GradeID values, resulting in a ten-class classification problem.

The system is set up to predict student performance. This is achieved by estimating the GradeID of the student according to the attributes trained. In other words, according to the previous observations, the system is experiencing and trying to predict the GradeID of an unknown sample when these attributes are provided. All the features are firstly used in our experiments. Then, the selected features are used to represent a student sample.

First of all, the attributes that are verbal in the dataset are numerated and quantized. This is achieved by quantizing the levels observed in every attribute. As a result of this quantization, we have all integer data in the data set. Then, a vector with integer component values is constructed to represent a student observation, being the feature vectors. The feature vectors are then set as the inputs of the machine learning part. Finally, the total samples are divided into training and test sets, where the training set includes 90% of the overall samples.

$$\overline{FV_j} = [f_1 \quad f_2 \quad f_3 \ldots\ldots f_{k1}] \tag{2}$$

$$\overline{FV_j} = [f_1 \quad f_2 \quad f_3 \ldots\ldots f_{k2}] \tag{3}$$

Furthermore, the system selects the best features with a feature selection procedure. The aim is to find out which attributes are more informative and contribute to student performance. In order to find the informative features, we employ a variance-based approach.

Table 1. Comparison of the proposed approach with other similar ones

No	Attribute	Type	No of Quantization Levels
1	Gender	Attribute Data – Demographic	***2***
2	NationalITy	Attribute Data – Demographic	***14***
3	PlaceofBirth	Attribute Data – Identity	***14***
4	StageID	Attribute Data	***3***
5	GradeID	Attribute Data	***10***
6	SectionID	Attribute Data	***3***
7	Topic	Attribute Data	***12***
8	Semester	Attribute Data	***2***
9	Relation	Attribute Data	***2***
10	raisedhands	Behavioral	***Numerical att.***
11	VisITedResources	Behavioral	***Numerical att.***
12	AnnouncementsView	Behavioral	***Numerical att.***
13	Discussion	Behavioral	***Numerical att.***
14	ParentAnsweringSurvey	Attribute Data	***2***
15	ParentschoolSatisfaction	Attribute Data	***2***
16	StudentAbsenceDays	Attribute Data	***2***
17	Class	Attribute Data	***3***

The variance-based feature selection procedure starts with finding the variances of every attribute throughout the training set. Then, the average of the variances is computed. After that, the variances above the average variance are selected, and corresponding features are used. In our experiments, there is 91.95% acceptable prediction rate after the feature selection. The selected features, which are "raisedhands", "VisitedResources", "AnnouncementsView", "Discussion" then used in feature vector descriptions. The feature vectors are represented in equations 2 and 3, before and after feature selection, where j is from 1 to 480 and f represents every single feature. FV represents the overall feature vectors in the equations 2 and 3. The number of features are reduced from 17 to 4, therefore, k1 and k2 that are the sizes of the vector lengths in the equations 2 and 3 are 17 and 4 respectively.

$$FM = \begin{bmatrix} \overline{FV_1} \\ \overline{FV_2} \\ \ldots\ldots \\ \overline{FV_{n1}} \end{bmatrix} \tag{4}$$

$$FM = \begin{bmatrix} \overline{FV_1} \\ \overline{FV_2} \\ \dots\dots \\ \overline{FV_{n2}} \end{bmatrix} \tag{5}$$

The equations 4 and 5 are showing how the overall dataset is organized before and after the feature selection, where k1 and k2 are the number of attributes before and after the feature selection, respectively. n_1 and n_2 are equal limits that are the number of samples used in the experiments. FM represents the overall matrix of data, where each row represents a sample student's feature vector, FV, and each column represents a feature. After the feature selection process, the number of features is reduced from k_1 to k_2, as shown in equations 2 and 3. The experimental results section discusses the limits and number of features in detail.

4. Preprocessing

The dataset undergoes into pre-processing stage before running the classifier. First of all, verbal attributes are converted into numerical information. Then, they are quantized. The quantization is done according to the number of different values involved. For example, the attribute gender has two possible values. Therefore, it is quantized in two levels, 1 and 2. The quantized information is then sent to the classifier. Figure 3 shows the quantization of verbal attributes.

	0	1	2	3	4	5	6	7
gender		F	M					
NationalITy		Egypt	Iran	Iraq	Jordan	KW	lebanon	Lybia
PlaceofBirth		Egypt	Iran	Iraq	Jordan	KuwaIT	lebanon	Lybia
StageID		lowerlevel	HighSchoc	MiddleSchool				
GradeID		G-02	G-04	G-05	G-06	G-07	G-08	G-09
SectionID		A	B	C				
Topic		Arabic	Biology	Chemistry	English	French	Geology	History
Semester		F	S					
Relation		Father	Mum					
raisedhands								
VisITedResources								
AnnouncementsView								
Discussion								
ParentAnsweringSur		No	Yes					
ParentschoolSatisfac		Bad	Good					
StudentAbsenceDays		Above-7	Under-7					
Class		H	L	M				

Fig. 3. Pre-processing: example quantization of verbal attributes. One full set of features here corresponds to FVin the equations (2),(3),(4) and (5). The size is k1 that is 17. After feature selection the size is k2 that is 4.

The number of quantized levels according to the attributes is listed in Table 2.

5. Feature Selection

The dataset includes 17 attributes in total to describe the student's current situation as a sample. The attributes included are listed in Table 1. In addition, there are statistical attributes like the number of days absent and the behavioral attributes that describe the student as a total sample. Noting that extraction of all the features may not be applicable all the time, the feature selection stage is proposed in our study. The main motivation of the feature selection is to decrease the number of features while conserving the system's accuracy. Besides, it will make the feature extraction stage practicable.

The feature selection approach proposed in our study is based on the variance of the features. Starting from the idea that the within-class variance should be the minimum in a classification problem, whereas between-class variance should be the maximum [36]. Accordingly, we measure every attribute's variance on the whole dataset and employ a variance-based feature selection procedure.

Table 2. List of attributes with quantization levels

Verbal Attributes	No of Quantization Levels	Non-verbal Attributes
gender	2	raisedhands
NationalITy	14	VisITedResources
PlaceofBirth	14	AnnouncementsView
StageID	3	Discussion
GradeID	10	
SectionID	3	
Topic	12	
Semester	2	
Relation	2	
ParentAnsweringSurvey	2	
ParentschoolSatisfaction	2	
StudentAbsenceDays	2	
Class	3	

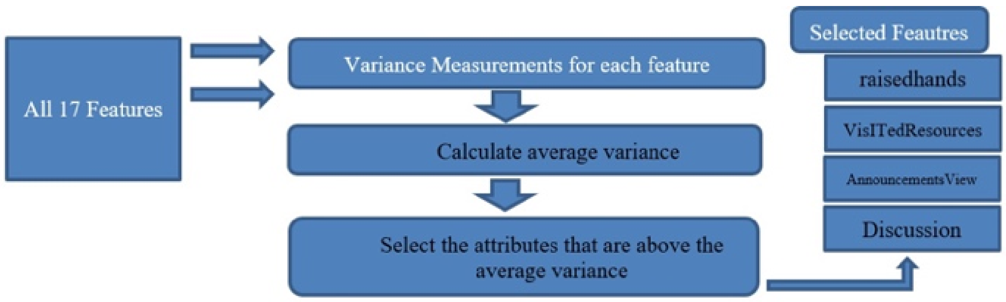

Fig. 4. Feature selection procedure.

The feature selection approach starts with finding the variances of every attribute among all the samples. Then, the average variance is calculated. After that, the attributes having the variance above the average variance are selected. These attributes are "raisedhands", "VisITedResources", "AnnouncementsView" and "Discussion". Interestingly, all the attributes having high variance are from behavioral attributes. The feature selection procedure is depicted in Figure 4. As machine learning algorithms include complex calculations, the visualizations of the selected features, which are the most informative features, are important [37] and [38]. Feature spaces for the selected features are plotted in Figures 5 and 6, by using the features "raisedhands", "VisITedResources", "AnnouncementsView" and "VisITedResources", "AnnouncementsView" and "Discussion" respectively.

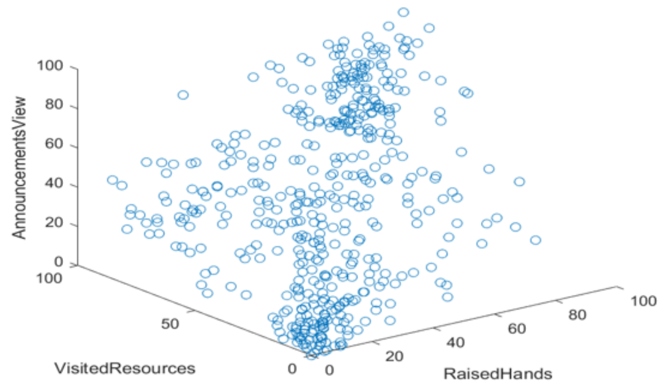

Fig. 5. Feature visualization: "raisedhands", "VisITedResources", "AnnouncementsView".

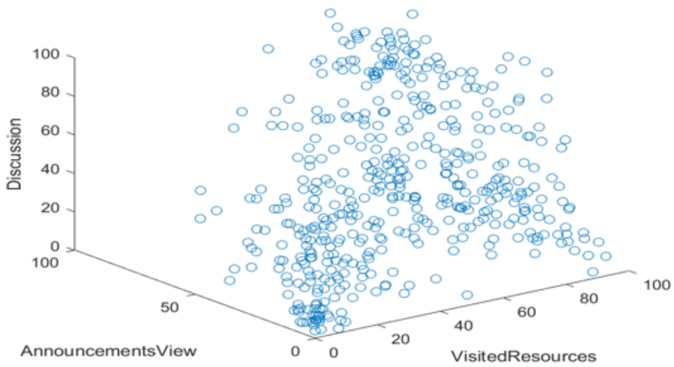

Fig. 6. Feature visualization: "VisITedResources", "AnnouncementsView" and "Discussion".

6. Experimental Results

The student success prediction approach is tested on the X-API dataset using the Matlab simulator. The Matlab code and the dataset of the algorithm can be accessed via the following link: https://github.com/umuttekgucbau/StdSuccessPredictionFNN. Initially, the preprocessing of the attributes is done. Then, the pre-processed data is used to train the feed-forward neural network classifier, as shown in Figure 2. Parameter settings play a role in the performance of the neural network [38–42] and we employ feed forward neural network with the default settings on

Matlab. The settings are shown in Figure 7. The test results are examined in two parts. Firstly, the system is tested by using all the attributes. Then the selected attributes are used.

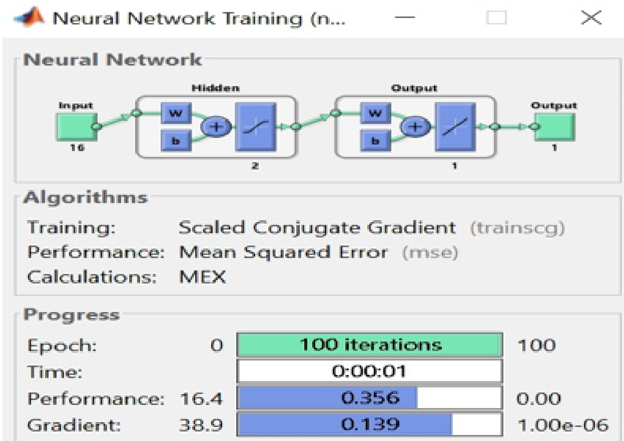

Fig. 7. Default settings used in experiments.

The feed-forward neural network is trained with 90% of the data. 10% are used for testing. It results in 440 samples in training and 41 samples in testing. In order to validate the test results, cross-validation is applied. The overall data matrix is randomly re-arranged, and then train and test subdivisions are made for every single experiment. For the validation process, 10-fold cross-validation is applied. That is, the data are split into 10 different folds and every time one single fold is tested. Then the results for every experiment are averaged.

Learning algorithms can be compared with respect to their average performances. This can be defined as the expected value of prediction error over training sets. K-fold cross-validation is a technique selected in this manner to validate the results of our neural network [43].

Table 3. Part one simulation results

Exp no	Exact	1diff	2diff	3diff	4diff
1	24	15	2	0	0
2	25	11	2	1	1
3	27	12	2	0	0
4	21	19	0	1	0
5	23	14	4	0	0
6	23	16	1	0	0
7	26	15	0	0	0
8	25	14	1	0	0
9	27	11	2	1	0
10	24	10	3	0	4
Average	**24.5**	**13.7**	**1.7**	**0.3**	**0.5**

10-fold cross-validation is applied for every experiment. After one experiment, the overall data is randomized, then train and test divisions are redone. After that, the experiments are completed with 10-fold cross-validation. So, randomization and cross-validation processes are sequentially applied.

Table 4. Predicted categories for part one simulations

Category	Percentage
Exact (%)	59.7561
1diff (%)	33.41463
2diff (%)	4.146341
3diff (%)	0.731707
4diff (%)	1.219512
Acceptable Predictions (exact+1diff) (%)	**93.17073**

The neural network results are rounded into integers and then grouped into four categories to evaluate the results. The categories include the exact grade predictions, one, two, three, or four nearest grade predictions, respectively. The approximation classes mean that the neural network's prediction is very close to the student's exact grade. For example, if a sample is classified with one nearest class prediction, it is either one less or one more than the exact grade. This is a very useful prediction in order to comment and guide the students throughout. Therefore, we give utmost importance to exact grade predictions and one nearest grade prediction. Therefore, we consider these predictions as acceptable predictions. The test results of the first part are provided in Table 3 accordingly. Table 4 shows the percentages of the categories predicted by the neural network.

Similarly, the experiments are accomplished by representing the samples with the selected features. Then, again, the validation process is followed. Test results by using the selected features are provided in Tables 5 and 6.

Table 5. Part two simulation results

Exp no	Exact	1diff	2diff	3diff	4diff
1	22	19	0	0	0
2	20	17	4	0	0
3	27	9	4	0	1
4	23	10	5	0	2
5	26	11	1	3	0
6	23	18	0	0	0
7	24	12	4	1	0
8	27	11	3	0	0
9	20	19	0	2	0
10	21	18	2	0	0
Average	**23.3**	**14.4**	**2.3**	**0.6**	**0.3**

Table 6. Predicted categories for part two simulations

Category	Percentage
Exact (%)	56.82927
1diff (%)	35.12195
2diff (%)	5.609756
3diff (%)	1.463415
4diff (%)	0.731707
Acceptable Predictions (exact+1diff) (%)	**91.95122**

7. Application

The experimental results show that the system can predict the exact student grades with 56% of accuracy. This is a good performance if we consider the quantization level of the grades into 10 different levels. As our application domain is student performance, accurate predictions with very near values are also valuable besides the exact prediction. Therefore, if we evaluate the system in this sense, its performance is around 91.95%, which is acceptable. For example, a student's grade will be predicted as one grade below or above, in the worst case with this percentage. Overall, the system provides acceptable predictions that can guide the teachers throughout the semester, according to the students' ongoing attributes. Besides, with accurate predictions, instructors will manage the resources and instructions more efficiently.

The comparison of the proposed approach is made with two other approaches on the same dataset. The comparison table is given in Table 7. As it is seen in Table 7, the proposed system achieves comparable rates. On the other hand, the system performances that are reported for the proposed model are based on the selected behavioral features. This will substantially provide an advantage to the instructors and managers in order to be able to track these student features accurately.

Table 7. Comparison table

Author(s)	Methodology	Accuracy
Amrieh, E. A., Hamtini, T. & Aljarah, I. (2015)[44]	FFNN, Naive Bayes, Decision Tree+filter-based Feature Selection	73.8% (high, medium, low-level accuracy)
Datta, B.A. & Dhakane, V. N. (2020)[45]	FFNN, SVM, Random Forrest, J48, Naive Bayes	76.07% (exact prediction)
Proposed Approach	FFNN + Variance based Feature Selection	91.95% (exact+one grade difference accuracy)

As listed in Table 7, the studies proposing improved classifiers to predict student performance have achieved promising results. However, exploring the key factors associated with student performance is still a challenge for researchers. In this sense, the paper is contributing to the field by proposing a feature selection procedure that tries to detect the most informative features. A recent study about finding the key factors is given in [29]. They proposed an optimized multi-label classifier. The classifier relates the predicted performances with the influence of various factors associated. Comparatively with the proposed approach, in [29] higher education was the target. The approach also exploits the students' pre-university performance, demonstrating the practicality of the approach in marking multiple factors affecting student performance. In our study, the key factors are analyzed for the single courses, and behavioral attributes are highlighted as a result of our feature selection procedure.

Also, it can be observed from the results that the features that survive after the selection procedure are the already numerical features. It seems that the system is eliminating the verbal features after the quantization. On the other hand, this observation is not a fact that the feature selection procedure always eliminates verbal features. When we analyze the eliminated features, we conclude that, for example, features that are quantized for 14 different levels are also below the average variance and eliminated. This observation is not supporting a fact and appears to be a delusion.

As a result, the system eliminates the majority of the features and selects the most informative four behavioral features. Even more, this approach will be a strong base for a future stable system that can run on big data.

8. Conclusion

The student success prediction process is useful in monitoring the student's current progress throughout the overall period of study. In this paper, a novel student success prediction model is proposed to estimate the students' grades. The proposed model applies a variance-based feature selection procedure. The system achieves 91.95% and 93.17% acceptable prediction rates with and without the feature selection process, respectively. The feature selection overcomes the practical problems of extracting all the attributes of a student sample. Instead, it gives the possibility and comfortability of extracting only the selected features for a successful student success prediction. It is observed from the results that the selected features are those that are the behavioral features. The detected behavioral features can be strong indicators of the student's success level in the near future. Consequently, the student success prediction model proposed in the paper can be adapted to any teaching model, and it can provide significant assistance to the management together with the guidance services supplied. Future works may include the enriching process of the features from the students' backgrounds to extract more behavioral features. Also, besides variance metric, other information content metrics like Fisher's score or entropy can be employed to enhance the information content analysis for better predictions.

8.1. Limitations

The prediction of student success is a complicated task in a real scenario. In this study, artificial neural networks are the basis, which requires observations and training data to be supplied. Possible limitations of this kind of prediction approach can be in extracting the features. There can be missing information about a student, like the behavioral features, or some features may not be accurately extracted. Although, in any case, there will be some healthy features extracted about every student, the system will learn the information that is provided in the training set. That is, limitations included in the training set are inherited from the overall system. On the other hand, in the proposed approach, the number of selected features is very low, and they are behavioral features. Therefore, these features can be extracted by teacher's observations, and the possible limitations of the system are minimized.

References

[1] E. ALYAHYAN and D. DÜTEGÖR, *Predicting academic success in higher education: literature review and best practices*, International Journal of Educational Technology in Higher Education **17**(1), pp. 1–21, 2020.

[2] D. IFENTHALER, J. Y. YAU, *Utilising learning analytics to support study success in higher education: a systematic review*, Educational Technology Research and Development **68**(4), pp. 1961–1990, 2020.

[3] H. S. ALENEZI, M. H. FAISAL, *Utilizing crowdsourcing and machine learning in education: Literature review*, Education and Information Technologies **25**(4), pp. 2971–2986, 2020.

[4] D. HOOSHYAR, M. PEDASTE, Y. YANG, *Mining educational data to predict students' performance through procrastination behavior*, Entropy **22**(1), paper 12, 2020.

[5] S. PHAUK and T. OKAZAKI, *Integration of educational data mining models to a web-based support system for predicting high school student performance*, International Journal of Computer and Information Engineering **15**(2), pp. 131–144, 2021.

[6] C. POZNA, R.-E. PRECUP, *Aspects concerning the observation process modelling in the framework of cognition processes*, Acta Polytechnica Hungarica **9**(1), pp. 203–223, 2012.

[7] E.-L. HEDREA, E. M. PETRIU, *Evolving fuzzy models of shape memory alloy wire actuators*, Romanian Journal of Information Science and Technology **24**(4), pp. 353–365, 2021.

[8] A. HELLAS, P. IHANTOLA, A. PETERSEN, V. V. AJANOVSKI, M. GUTICA, T. HYNNINEN, S. N. LIAO, *Predicting academic performance: a systematic literature review*, Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education, Larnaca, Cyprus, pp. 175–199, 2018.

[9] M. C. CRISTESCU, *Machine learning techniques for improving the performance metrics of functional verification*, Romanian Journal of Information Science and Technology **24**(1), pp. 99–116, 2021.

[10] I.-D. BORLEA, R.-E. PRECUP, A.-B. BORLEA, *Improvement of k-means cluster quality by post processing resulted clusters*, Procedia Computer Science **199**, pp. 63–70, 2022.

[11] I. A. ZAMFIRACHE, R.-E. PRECUP, R.-C. ROMAN, E. M. PETRIU, *Policy iteration reinforcement learning-based control using a grey wolf optimizer algorithm*, Information Sciences **585**, pp. 162–175, 2022.

[12] C. LEI, K. F. LI, *Academic performance predictors*, Proceedings of IEEE 29th International Conference on Advanced Information Networking and Applications Workshops, Gwangju, Korea (South), pp. 577–581, 2015.

[13] N. TALIB, S. S. SANSGIRY, *Determinants of academic performance of university students*, Pakistan Journal of Psychological Research **27**(2), pp. 265–278, 2012.

[14] X. XU, J. WANG, H. PENG, R. WU, *Prediction of academic performance associated with internet usage behaviors using machine learning algorithms*, Computers in Human Behavior **98**, pp. 166–173, 2019.

[15] B. GBKA, *Psychological determinants of university students' academic performance: an empirical study*, Journal of Further and Higher Education **38**(6), pp. 813–837, 2014.

[16] S. SOTHAN, *The determinants of academic performance: evidence from a Cambodian university*, Studies in Higher Education **44**(11), pp. 2096–2111, 2019.

[17] Z. J. KOVACIC, *Early prediction of student success: Mining student enrollment data*, Proceedings InSITE 2010: Informing Science + IT Education Conference, Cassino, Italy, pp. 647–665, 2010.

[18] E. A. AMRIEH, T. HAMTINI, I. ALJARAH, *Mining educational data to predict student's academic performance using ensemble methods*, International Journal of Database Theory and Application **9**(8), pp. 119–136, 2016.

[19] D. KABAKCHIEVA, *Predicting student performance by using data mining methods for classification*, Cybernetics and Information Technologies **13**(1), pp. 61–72, 2013.

[20] C. JALOTA, R. AGRAWAL, *Analysis of educational data mining using classification*, Proceedings of 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing, Faridabad, India, pp. 243–247, 2019.

[21] I. MILOS, S. PETAR, V. MLADEN, A. WEJDAN, *Students' success prediction using Weka tool*, INFOTEH-JAHORINA 15, pp. 684–688, 2016.

[22] P. KAVIPRIYA, *A review on predicting students' academic performance earlier*, using data mining techniques, International Journal of Advanced Research in Computer Science and Software Engineering **6**(12), pp. 101–105, 2016.

[23] N. ANKITA, R. ANJALI, *Analysis of student performance using data mining technique*, International Journal of Innovative Research in Computer and Communication Engineering **5**(1), paper 12, 2017.

[24] J. L. RASTROLLO-GUERRERO, J. A. GÓMEZ-PULIDO, A. DURÁN-DOMÍNGUEZ, *Analyzing and predicting students' performance by means of machine learning: A review*, Applied Sciences **10**(3), paper 1042, 2020.

[25] S. J. H. Yang, O. H. T. Lu, A. Y. Q. HUANG, J. C. H. HUANG, H. OGATA, A. J. Q. LIN, *Predicting students' academic performance using multiple linear regression and principal component analysis*, Journal of Information Processing **26**, pp. 170–176, 2018.

[26] L. M. A. ZOHAIR, *Prediction of student's performance by modelling small dataset size*, International Journal of Educational Technology in Higher Education **16**(1), paper 27, 2019.

[27] N. ALANGARI, R. ALTURKI, *Predicting students final GPA using 15 classification algorithms*, Romanian Journal of Information Science and Technology **23**(3), pp. 238–249, 2020.

[28] A. AHADI, R. LISTER, H. HAAPALA, A. VIHAVAINEN, *Exploring machine learning methods to automatically identify students in need of assistance*, Proceedings of 11$^{\text{th}}$ Annual International Conference on International Computing Education Research, Omaha, NE, USA, pp. 121–130, 2015.

[29] A. ALSHANQITI, A. NAMOUN, *Predicting student performance and its influential factors using hybrid regression and multi-label classification*, IEEE Access **8**, pp. 203827–203844, 2020.

[30] J. XIAO, L. WANG, J. ZHAO, A. FU, *Research on adaptive learning prediction based on XAPI*, International Journal of Information and Education Technology **10**(9), pp. 679–684, 2020.

[31] T. L. FINE, *Feedforward Neural Network Methodology*, Springer Science & Business Media, Cham, 2016.

[32] M. M. ABU TAIR, A. M. EL-HALEES, *Mining educational data to improve students' performance: A case study*, International Journal of Information **2**(2), pp. 56–65, 2012.

[33] Backpropagation. Accessed: August 2, 2022. [Online]. Available: https://brilliant.org/wiki/backpropagation/$\#:\sim$.

[34] X. YU, M. O. EFE, O. KAYNAK, *A general backpropagation algorithm for feedforward neural networks learning*, IEEE Transactions on Neural Networks **13**(1), pp. 251–254, 2002.

[35] E. A. AMREIH, T. HAMTINI, I. ALJARAH, *Student's academic performance dataset (xAPI-Edu-Data)*, Accessed: May 5, 2021. [Online]. Available: https://www.kaggle.com/aljarah/xAPI-Edu-Data .

[36] K. YURTKAN, H. SOYEL, H. DEMIREL, *Feature Selection for Enhanced 3D Facial Expression Recognition Based on Varying Feature Point Distances*, in Information Sciences and Systems, Springer, Cham, pp. 209–217, 2013.

[37] S. SRIRAM, R. VINAYAKUMAR, M. ALAZAB, K. P. SOMAN, *Network flow based IoT botnet attack detection using deep learning*, In IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops, Toronto, ON, Canada, pp. 189–194, 2020.

[38] R. VINAYAKUMAR, M. ALAZAB, S. SRINIVASAN, Q. V. PHAM, S. K. PADANNAYIL, K. SIMRAN, *A visualized botnet detection system based deep learning for the internet of things networks of smart cities*, IEEE Transactions on Industry Applications **56**(4), pp. 4436–4456, 2020.

[39] S. SELVIN, R. E. VINAYAKUMAR, A. GOPALAKRISHNAN, V. K. MENON, K. P SOMAN, *Stock price prediction using LSTM, RNN and CNN-sliding window model*, Proceedings of 2017 International Conference on Advances in Computing, Communications and Informatics, Udupi, India, pp. 1643–1647, 2017.

[40] V. ATHIRA, P. GEETHA, R. VINAYAKUMAR, K. P. SOMAN, *Deepairnet: Applying recurrent networks for air quality prediction*, Procedia Computer Science **132**, 1394–1403, 2018.

[41] R. VINAYAKUMAR, K. P. SOMAN, P. POORNACHANDRAN, *Applying deep learning approaches for network traffic prediction*, Proceedings of 2017 International Conference on Advances in Computing, Communications and Informatics, Udupi, India, pp. 2353–2358, 2017.

[42] S. ASWIN, P. GEETHA, R. VINAYAKUMAR, *Deep learning models for the prediction of rainfall*, Proceedings of 2018 International Conference on Communication and Signal Processing, Chennai, India, pp. 0657–0661, 2018.

[43] Y. BENGIO, Y. GRANDVALET, *No unbiased estimator of the variance of k-fold cross-validation*, Journal of Machine Learning Research **5**, pp. 1089–1105, 2004.

[44] E. A. AMRIEH, T. HAMTINI, I. ALJARAH, *Preprocessing and analyzing educational data set using X-API for improving student's performance*, Proceedings of 2015 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies, Amman, Jordan, pp. 1–5, 2015.

[45] B.A. DATTA, N. V. DHAKANE, *EPSP: Early prediction of student performance using classification method of data mining*, Resincap Journal of Science and Engineering **4**(4), 2020.