Enhancing student classification using Transformer Neural Networks

Abderrafik LAAKEL HEMDANOU [a], Youssef ACHTOUN [a]

Sara MOUALI [a] and Mohammed LAMARTI SEFIAN [a]

[a] AMCS team, Normal Higher School, Abdelmalek Essaadi University, Morocco

abderrafik.laakelhemdanou@etu.uae.ac.ma
achtoun44@outlook.fr
mouali427@gmail.com
lamarti.mohammed.sefian@uae.ac.ma

Abstract. Classification represents a challenge for researchers and a very important task for educational systems in general to help make good decisions to improve student engagement in intelligent tutoring systems. In this study, we propose an approach to classifying students on the basis of various characteristics using the latest Transformer model, which has shown its effectiveness in handling the nature of student data. Our methodology concerns the feature of the most important factors influencing academic performance to facilitate student classification, followed by the encoding of various student information, such as academic results, socio-economic background and extra-curricular activities, into a complete input sequence. Then the KNN, SVC, DNN and Transformer (Encoder-Decoder) models are used to classify students on the full data and on sub-data from LightGBM, Elastic Net and PCA. To assess the effectiveness of the proposed method, multiple metrics were used to perform a comprehensive evaluation. The results indicate that the Transformer model exceeds the performance benchmarks set by other models, demonstrating its effectiveness.

Keywords: Transformer models, Student classification, Selection features.

M. Khaldi et al. (eds.), Proceedings of the E-Learning and Smart Engineering Systems (ELSES 2024), Atlantis Highlights in Social Sciences, Education and Humanities 36,
https://doi.org/10.2991/978-2-38476-408-2_47

1 Introduction

The basic concept of a Transformer Neural Network has its roots from the decades-old neural network research. This approach to student classification has overcome many drawbacks and challenges that the traditional algorithms are facing. By adopting a theoretical perspective that includes consideration for both student success and recent advancements in machine learning, we have based our objectives on this concept and set out to develop a more effective classification mechanism. With the power of deep learning and neural networks, it has the potential to better understand students. This not only benefits classification but also feedback and student support as well. However, it is quite new in educational data mining when compared to other fields, such as natural language processing or machine translation. It is still a research-level topic and lacks real development. Also, due to the problem complexity, it hasn't caught much attention from scholars in this field. That gives us the motivation to start this research, focusing on exploring and enhancing the idea for student classification by using a Transformer Neural Network. In this study, we will try to compare four popular models for classifying student performance with the transformer Encoder-decoder model [2]. The models chosen for comparison are : KNN, SVC, and DNN. We have tried to identify the most important factors influencing students' school performance using the three selection features methods: Elastic net, Light GBM, PCA. The process used in this study is shown in the following figure.

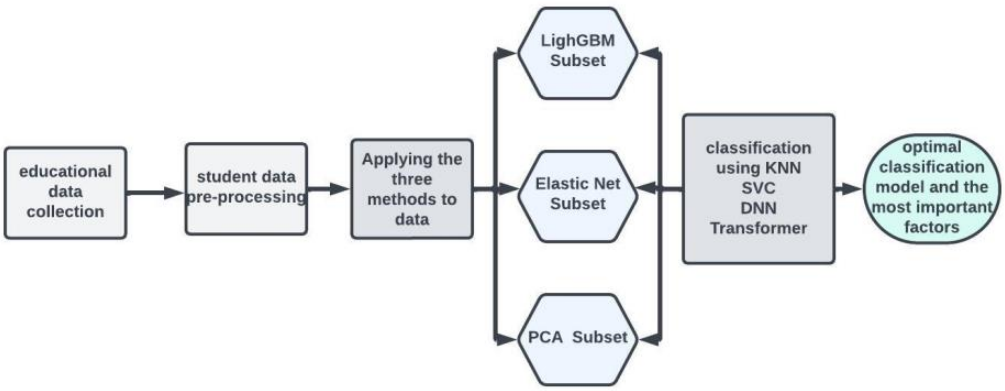

Figure 1: The used process.

1.1. Background

Due to the increasingly large volume of educational data, student classification has become an important research area in educational data mining. In particular, student classification aims to predict student academic performance or diagnose student learning problems based on student data [1]. So far, a wide variety of machine learning and data mining algorithms have been applied to student classification research. They include but are not limited to decision trees, support vector machines, and clustering. Among the many different algorithms that have been used for student classification, the regression algorithms have been particularly popular in the literature. They are easy to implement, quick to train and produce interpretable results. A wide variety of different regression algorithms, such as multinomial logistic regression and their ensemble versions, such as random forests. However, they require the students to be partitioned into a small number of discrete classes, which are often defined by human experts. Such an expert-defined classification approach is called supervised student classification. This article explores an innovative neural network algorithm for data mining called Transformer Neural Networks. This algorithm has only recently been proposed and to date has rarely been applied to educational data of this type. The proposed work aims to fill this gap and evaluate the potential of Transformer Neural Networks for student classification tasks and their comparative predictive effectiveness.

1.2. Objectives

It encourages to create an automatic educational system to provide necessary study materials and appropriate teaching strategies for different students. Different from the current about 70% accuracy student classification solutions, this research aims to increase the accuracy of student classification utilizing Transformer Neural Networks and thereby improving the classification performance. The objectives can be concluded as:

- To investigate the effectiveness of Transformer Neural Networks in student classification.
- To propose a new student classification approach using Transformer Neural Network.
- To compare the performance of the new approach with the existing methods.

Cross-validation will be used to assess the prediction capability of the model, which is a more reliable method than single validation. In single validation, data is randomly partitioned into two sets and used for training and for validation. But K-fold

cross-validation, in which the original sample is partitioned into K subsamples, is repeatedly used to cross-validate the model. This process will be performed many times, and the results will be averaged over runs. For this research, K will be set to 5, which is considered as a balance approach, in selecting the K value. It will minimize bias and speed up the computation.

2. Methodology

2.1 Selection of features

We used a statistical method called principal component analysis (PCA), which converts a data set containing correlated variables into a new set of linearly uncorrelated variables called "principal components," in order to reduce the dimensionality of the data and identify the factors that have a strong correlation with the students' academic performance, thereby facilitating the classification process. Before using PCA [4], data must be standardized to make sure they are on the appropriate scale. These principal components are ranked in descending order of significance in terms of variance, meaning that the first principal component captures the greatest variance, the second captures the second greatest variance, and so on. The PCA equation can be summed up as follows:

$$Y = XP\ .$$

Such that Y represents the matrix of data projected on the principal components (the new features after PCA), X is the matrix of original data that have been standardized (centered and reduced) and P is the matrix containing the eigenvectors (principal axes) selected from the eigenvalue decomposition of the covariance matrix of the standardized data.

We used the Elastic Net method, a regularization technique used in linear or logistic regression models to prevent overfitting and manage multicollinearity problems [8]. It combines the properties of two types of regularization: L1 (Lasso) and L2 (Ridge), by adding to the cost function a penalty which is a linear combination of the L1 and L2 norms of the model coefficients. Elastic Net regularization is useful when used as a feature selection method, which can lead to a reduction in dimensionality.

$$\hat{\beta} = argmin_{\beta} \left\{ \frac{1}{2n} \|y - X\beta\|_2^2 + \gamma \left(\frac{1-\alpha}{2} \|\beta\|_2^2 + \alpha \|\beta\|_1 \right) \right\}.$$

Such that $\hat{\beta}$ are the estimated coefficients of the model, y is the response variable vector, X is the matrix of explanatory variables, $\beta$is the vector of coefficients to be

estimated, n is the number of observations, γ is the regularization parameter which controls the strength of the penalty, α is a parameter that mixes the L1 and L2 penalties, $\|\beta\|_1$is the L1 norm which corresponds to the sum of the absolute values of the coefficients (Lasso penalty), $\|\beta\|_2^2$ is the L2 squared norm, which corresponds to the sum of the squared coefficients (Ridge penalty). and we used LightGBM because one of its features is its ability to evaluate the importance of variables (or features) in the predictive model [9] . Variable importance is calculated on the basis of the number of times a feature is used to split the data in the trees, as well as the improvement in overall model performance brought about by these splits. In other words, the more times a feature is used to make key decisions in the trees, and the more these decisions improve prediction, the more important the feature is considered to be. This enables LightGBM users to understand which features contribute most to prediction and can be used to simplify the model by eliminating less important ones, which can also help reduce dimensionality and improve model efficiency.

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + \varphi f_t(x_i) \ .$$

Such that $\hat{y}_i^{(t)}$ is the prediction for the observation at turn, $\hat{y}_i^{(t-1)}$ is the cumulative prediction for the observation up to the previous turn, φ is the learning rate, which controls the contribution of each new tree added to the model, $f_t(x_i)$ is the tree prediction for observation .

2.2 Classification models

We will describe each model utilized in the research in this section:
KNN, also known as k-Nearest Neighbors, is a supervised learning technique used for regression and classification. Because it maintains instances of the training data instead of explicitly building an internal model, it is referred to as a lazy learning or non-parametric algorithm [7]. In the test phase, classification or prediction is carried out by examining the k closest cases, or neighbors. It works by using the k nearest neighbors of an unknown data point to determine its label (class or value). The first step in the KNN classification process is to compute the distances between each unknown data point and every other data point in the training set. The next step is to choose the k closest data points based on the estimated distances—the Euclidean distance being the most widely utilized metric. It is possible to identify the majority class among these k neighbors. Lastly, it is projected that the missing data point is a member of the majority class. Formally, this step is represented by the following equation:

$$\hat{y} = argmax \sum_{i=1}^{k} (y_i == c) \ .$$

Such that $\hat{y}$ is the label prediction for the unknown data point, c represents the possible classes, y_i is the label of each neighbor among the k nearest neighbors. SVC a advanced classification method derived from support vector machines (SVM) is the Support Vector Classifier. The goal of this approach is to create the best possible hyperplane to divide data classes with the highest margin. Support vectors, or the data points that are closest to the hyperplane, are crucial for establishing the margin and providing direction for building the hyperplane. SVC handles situations where data are not fully separable by using a cost function that penalizes mistakes while accepting some deviations. SVC uses kernel functions to project data into a higher-dimensional space where linear separation is possible when data cannot be separated linearly [7]. These kernel functions, which include sigmoid, polynomial, RBF, and linear, enhance the model's adaptability to a broad range of data types. SVC is an effective tool for classification since it uses quadratic optimization techniques to address the associated optimization problem. The fundamental formula for the Support Vector Classifier (SVC), sometimes referred to as the Support Vector Machine (SVM), is based on the pursuit of the ideal hyper plane, with the hyper plane equation being written as follows:

$$W.X + b = 0\,.$$

Such that W is the normalized weight vector perpendicular to the separating hyperplane, X is the observation feature vector, b is the bias term. DNN A deep neural network is an artificial neural network architecture characterized by its deepness, through the high number of hidden layers it contains. These intermediate layers enable the DNN to capture and model complex, non-linear relationships between data. Each layer is made up of neurons that perform calculations by weighting inputs, adding bias and passing the result through a non-linear activation function. These activation functions, such as ReLU or Sigmoid, are essential to enable the network to learn complex patterns. The learning process in a DNN involves iterative adjustment of weights and biases via backpropagation and gradient descent, minimizing a cost function that evaluates the network's prediction error. DNNs are extremely powerful for applications such as image and speech recognition [3], but they require substantial amounts of data and computational resources to be effective. Their ability to learn hierarchical representations of data makes DNNs a central pillar in advances in deep learning and artificial intelligence [10]. The essential equation of a Deep Neural Network (DNN) for a single neuron in a hidden layer can be represented as follows:

$$a_i^{(l)} = \sigma\left(\sum_j \left(w_{ij}^{(l)}.a_j^{(l-1)}\right) + b_i^{(l)}\right).$$

Such that $a_i^{(l)}$is the activation of the neuron i in the layer l, $w_{ij}^{(l)}$ is the weight associated with the connection between the neuron I in the layer l and the neuron j in the layer l-1, $a_j^{(l-1)}$ is the activation of the neuron j in the previous layer l-1, $b_i^{(l)}$ is the bias associated with the neuron i in the layer l, σ is the activation function, for our model the activation function used is Relu for the first and second layer, sigmoid for the third layer , and Adam optimizer, the sigmoid function is defined by $\sigma(X) = \frac{1}{1+e^{-X}}$ and Relu defined by $R(X) = \max(0, X)$.

Transformer model is a revolutionary neural network architecture that relies on the attention mechanism to process sequences of data, and is particularly effective in natural language processing tasks such as text classification. Unlike recurrent or convolutional models, the Transformer uses positional encodings to maintain knowledge of the order of elements in a sequence [5]. Its architecture consists mainly of encoding blocks that transform input into a context-rich representation. Each encoding block includes a multi-headed attention mechanism, which allows the model to focus on different parts of the sequence, and a feedforward neural network that refines the attention output. For classification, the resulting representation is usually aggregated by a pooling layer, and then passed through one or more dense layers to determine the probabilities of each class. The Transformer model is known for its ability to handle long sequences and for its efficient parallelization, which has led to reduced training times and improved performance on various NLP tasks. It has served as the basis for advanced models such as BERT and GPT, which have set new standards in the field. For the transformer, we will summarize these equations in two essential equations, the encoder equation and the decoder equation. The encoder equation is defined as follows:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V .$$

Where Q, K, and V represent the query, key and value matrices respectively, and $d_k$is the dimension of each key vector.

to introduce the essential decoder equation we'll use the encoder's intermediate representations $H = \{h_1, h_2, \ldots . h_n\}$ and the partial target sequence up to the current time step $Y_{<t} = \{y_1, y_2, \ldots . y_{t-1}\}$, the attention equation in the decoder is similar to that of the encoder but with a modification to prevent the decoder from looking at future steps:

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V\ .$$

Where $Q = Y_{<t}W^Q$, $K = HW^K$ and $V = HW^v$, such that W^Q , W^K and W^v are learned weight matrices.

3. Result and discussion

The secondary school student outcomes from two Portuguese schools comprise the data used in this study[13]. Utilizing school reports and questionnaires, data variables such as student grades, demographic, social, and school related aspects were gathered. Regarding performance in the two different topics of mathematics (mat) and Portuguese language (por), two data sets are supplied. Features are listed, and the download page [13] provides an explanation of each feature. When analyzing data for student classification, it is essential to be able to identify and understand the most important factors influencing the results. Two effective methods for exploring and interpreting large data sets are Principal Component Analysis (PCA), LightGBM and PCA.

The table below shows the results of these three methods.

Method	Selected characteristics
PCA	G2, Absences, Age, Famrel, Health, Reason-other, Studytime, Gout, Fedu, Freetime, Activities.
LightGBM	G2, Absences, G1, health, Reason, Age, Famrel ,Fjob, Activities, Mjob, Freetime.
Elasctic Net	G2, G1, Absences, famrel, failures, romantic, schoolsup, Activities, reason, age, Fjob.

Table 1: Characteristics most selected by the methods.

The analysis of the features selected by PCA, LightGBM, and Elastic Net, as shown in Table 1, reveals a compelling insight into the factors that are deemed influential in our study. Notably, 'G2' (second-grade scores) and 'Absences' emerge as common

significant predictors across all three methods, highlighting the critical role of continuous academic performance and attendance in the model's outcomes. The presence of personal and family-related features such as 'Age', 'Famrel', 'Health', and 'Reason' in the PCA method, and 'Health', 'Reason', 'Famrel', and 'Fjob' in LightGBM, underscores the multifaceted influences on a student's life. Elastic Net's selection, which includes a mix of academic, personal, and support-related features, further corroborates the complexity of the factors at play. The convergence of certain features across different methods reinforces their importance, while the unique attributes identified by each algorithm reflect their individual strengths and assumptions. This collective evidence points towards a nuanced interplay between academic, personal, and environmental factors in shaping the educational outcomes of students. A model's performance in binary classification is assessed using F1-score, accuracy, precision, and recall. Precision is concerned with the percentage of true positives among positive predictions, whereas accuracy evaluates the percentage of accurate predictions. Recall measures the capacity to recognize every true positive. A fair assessment is provided by the F1-score, which is the harmonic mean of recall and precision. This is especially true when there are uneven classes. The characteristics of the issue and the relative significance of Type I and Type II faults determine which features to provide. It is recommended to employ these indicators in tandem for a comprehensive assessment. The following definition applies to these assessment metrics:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN} \qquad \mathrm{F1-Score} = 2 \times \frac{\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision}+\mathrm{Reacll}}$$

With TP is True Positives, TN is True Negatives, FP is False Positives, and

FN is False Negatives.

The table below shows the different results obtained from each model on the data and on the subsets of PCA, Light GBM, and Elastic net.

Model		Data	Student		
	Accuracy	Precision	F1-Score	Reacll	Roc-Auc
KNN	0.62	0.66	0.74	0.84	0.55
SVC	0.68	0.69	0.79	**0.94**	0.66
DNN	0.69	0.70	**0.80**	0.92	0.69
Trans-former	**0.71**	**0.72**	**0.80**	0.90	**0.72**

		Subset	LightGBM		
KNN	0.78	0.79	0.84	0.90	0.85
SVC	0.86	0.87	0.89	**0.92**	0.89
DNN	0.88	0.93	0.91	0.88	0.94
Trans-former	**0.93**	**1.0**	**0.94**	0.90	**0.96**
		Subset	Elastic Net		
KNN	0.82	0.81	0.87	**0.94**	0.88
SVC	0.88	0.90	0.91	0.92	0.96
DNN	0.89	0.94	0.92	0.90	0.95
Trans-former	**0.92**	**1.0**	**0.94**	0.93	**0.97**
		Subset	PCA		
KNN	0.74	0.75	0.82	0.92	0.81
SVC	0.87	0.88	0.90	0.92	0.94
DNN	0.89	0.89	0.92	**0.96**	0.96
Trans-former	**0.91**	**0.95**	**0.93**	0.92	**0.98**

Table 2: Results of different models.

The ROC (Receiver Operating Characteristic) curve is used to evaluate the discrimination capability of a binary classifier. It graphically illustrates the model's performance at different decision thresholds, by plotting the true-positive rate (also called sensitivity or recall) against the false-positive rate (1 - specificity). The x-axis represents the false-positive rate, i.e. the proportion of misclassified negative samples among all negative samples [12]. In parallel, the y-axis indicates the true positive rate. The area under the ROC curve (AUC) provides an aggregate measure of classifier performance. It evaluates the probability that the model assigns a higher probability of classification to a positive sample than to a negative sample, chosen at random. An AUC of 1 indicates perfect discrimination, while an AUC of 0.5 corresponds to classification equivalent to chance. The AUC is therefore a synthetic indicator for comparing and selecting binary classification models.

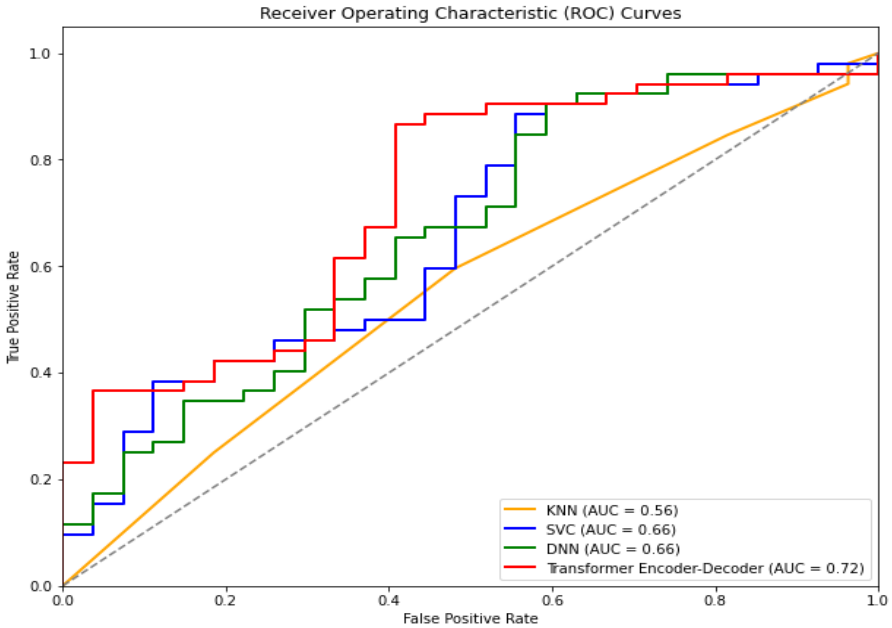

Figure 2 The ROC curve on student data

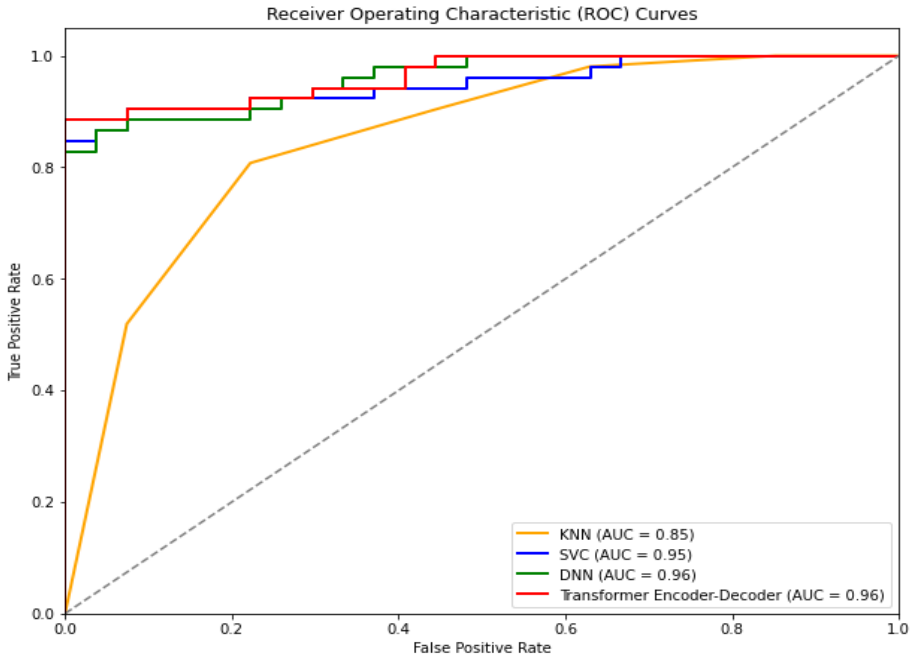

Figure 3 The ROC curve on subset LightGBM

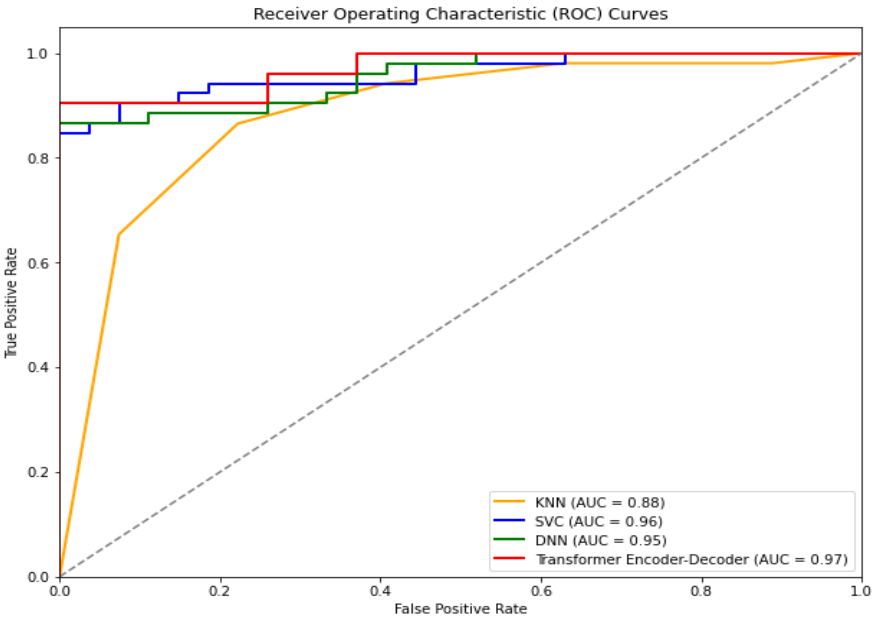

Figure 4 The ROC (Receiver Operating Characteristic) curve on subset Elastic Net

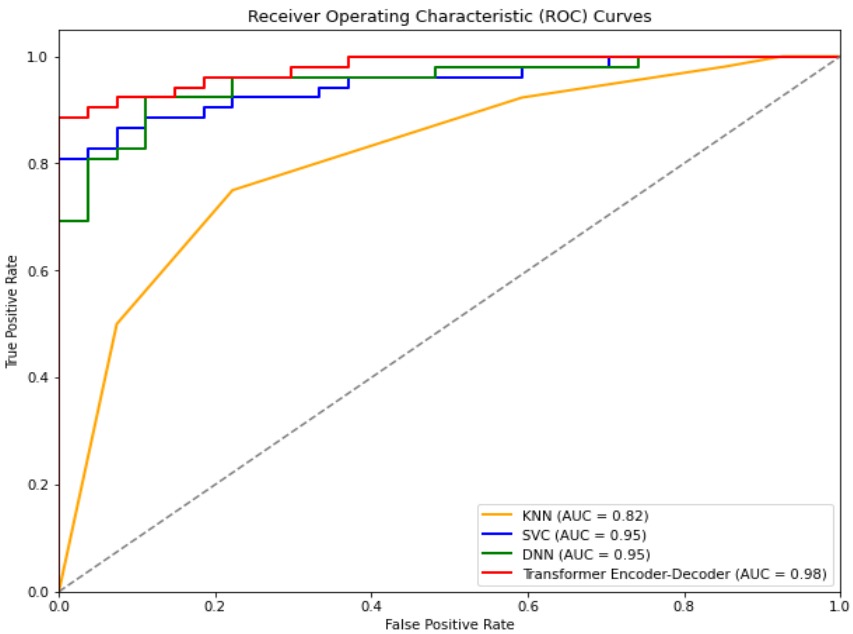

Figure 5 The ROC curve on subset PCA

The performance of four different machine learning models KNN, SVC, DNN [6], and Transformer was evaluated across various data subsets: Student, Light GBM, Elastic Net, and PCA. The models were assessed based on five key metrics Accuracy, Precision, F1-Score, Recall, and Roc-Auc.

For the Student data set, the Transformer model outperformed the others with the highest accuracy (0.71), precision (0.72), and Roc-Auc (0.72). It also achieved competitive F1-Score (0.80) and Recall (0.90) values. The DNN model followed closely behind, with a slightly lower performance across all metrics compared to the Transformer. In the Light GBM subset, the Transformer again led with the highest scores in accuracy (0.93), precision (1.0), and Roc-Auc (0.96), indicating a near-perfect ability to distinguish between classes. The DNN model showed superior performance in terms of precision (0.93) and Roc-Auc (0.94), suggesting a strong discriminative capacity. The Elastic Net subset results were similar, with the Transformer model achieving the highest scores in accuracy (0.92), precision (1.0), and Roc-Auc (0.97). The DNN model maintained high scores, particularly in precision (0.94) and Roc-Auc (0.95). and for the PCA subset, the Transformer model demonstrated the highest accuracy (0.91), precision (0.95), and Roc-Auc (0.98), while the DNN model showed the highest F1-Score (0.92) and Recall (0.96).

The results indicate that the Transformer model consistently outperforms the other models across all data subsets and metrics [10, 11]. Its superior performance can be attributed to its ability to capture complex patterns and dependencies in the data, which is particularly beneficial for classification tasks. The DNN model also performed well, particularly in the Light GBM and Elastic Net subsets, where it achieved high precision and Roc-Auc scores. This suggests that deep learning models are capable of capturing intricate features in the data, leading to strong predictive performance. The SVC model showed improvement in performance as the data complexity increased, performing best in the PCA subset. This improvement can be due to the model's ability to handle high-dimensional data effectively after dimensionality reduction techniques like PCA. The KNN model, while generally the least performant still showed respectable scores, especially in the Elastic Net and PCA subsets. Its performance in the Light GBM subset was notably strong, with an accuracy of 0.78 and Roc-Auc of 0.85, which could be due to the subset's characteristics that favor the KNN's instance-based learning approach [11].

In general, the Transformer model's top performance across nearly all metrics and subsets suggests that it is a robust choice for classification tasks.

4. Conclusion

In conclusion, this study successfully identified key factors influencing student outcomes. Academic performance (G2 scores) and attendance (Absences) emerged as consistently important predictors across different feature selection methods. Furthermore, the inclusion of features like family background and student support highlights the multifaceted influences beyond just academic factors. The Transformer model consistently outperformed others across various data subsets and metrics, demonstrating its strength in capturing complex data patterns. Deep learning models like DNNs also showed promising results, particularly in identifying intricate features. Additionally, the study revealed that model selection can be optimized based on data characteristics. For instance, the SVC performed better with high-dimensional data processed by PCA, while KNN was effective in specific subsets due to this approach. In general, this study offers valuable insights for student outcome prediction. The Transformer model's dominance suggests its potential for such classification tasks. In future research, we will try to collect and use a large set of data and a wide variety of variable selection methods, in order to build a more general model and discover the key factors in classifying student performance that will help educators and researchers strengthen the factors of school success and overcome the causes of school failure.

References

1. Mohammed Afzal Ahamed, Prayuk Chaisanit, and TR Mahesh. Performance of student prediction. International Journal of Computer Science and Information Technology, 8(6), 2019.

2. Nicholas Robert Beckham, Limas Jaya Akeh, Giodio Nathanael Pratama Mitaart, and Jurike V Moniaga.Determining factors that affect student performance using various machine learning methods. Procedia Computer Science, 216:597‑603, 2023.

3. Mohammed El Fouki, Noura Aknin, and Kamal El Kadiri. Multidimensional approach based on deep learning to improve the prediction performance of dnn models. International Journal of Emerging Technologies in Learning, 14(2), 2019.

4. Ian T Jolliffe. Principal component analysis for special types of data. Springer, 2002.

5. Tieyuan Liu, Meng Zhang, Chuangying Zhu, and Liang Chang. Transformer-based convolutional forgetting knowledge tracking. Scientific Reports, 13(1):19112, 2023.

6. Aya Nabil, Mohammed Seyam, and Ahmed Abou-Elfetouh. Prediction of students' academic performance based on courses' grades using deep neural networks. IEEE Access, 9:140731‑140746, 2021.

7. Kingsley Okoye, Julius T Nganji, Jose Escamilla, and Samira Hosseini. Machine learning model (rg-dmml) and ensemble algorithm for prediction of students 'retention and graduation in education. Computers and Education: Artificial Intelligence, 6:100205, 2024.

8. Yoo, Jin Eun ,TIMSS 2011 student and teacher predictors for mathematics achievement explored and identified via elastic net, Frontiers in psychology, 9:279404,2018.

9. Amal, A, Khaldi, M.Aammou. Enhancing the prediction of student performance based on the machine learning XGBoost algorithm. In Interactive Learning Environments; Taylor & Francis: Abingdon, UK, 2021; pp. 1–20, 2019.

10.Kukkar A, Mohana R, Sharma A and Nayyar A. A novel methodology using RNN + LSTM + ML for predicting student's academic performance. Education and Information Technologies. 10.1007/s10639-023-12394-0, 2024.

11. Rahul and Katarya R. A Systematic Review on Predicting the Performance of Students in Higher Education in Offline Mode Using Machine Learning Techniques. Wireless Personal Communications. 10.1007/s11277-023-10838-x. 133:3. (1643-1674). Online publication date: 1-Dec-2023.

12. P. P and Veeramanju K. (). A Systematic Review on Machine Learning Algorithms for Customer Satisfaction Classification in Various Fields. International Journal of Management, Technology, and Social Sciences. 10.47992/IJMTS.2581.6012.0305. (326-339) , 2023.

13.Cortez,Paulo. (2014). Student Performance. UCI Machine Learning Repository. https://doi.org/10.24432/C5TG7T.