International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Predicting and Analysis of Students' Academic Performance using Hybrid Techniques

[1]M. AArul Rozario, [2]Dr. R. GunaSundari

Submitted: 25/01/2024 **Revised**: 03/03/2024 **Accepted**: 11/03/2024

Abstract: This paper presents a framework for predicting the academic performance of first-year bachelor's students in computer science courses using data mining techniques. With the exponential growth of data in educational databases, data mining offers a promising avenue for uncovering valuable insights and patterns. The framework employs classification methods including Decision Tree, Naive Bayes, and Multi-Layer Perception, implemented through the python tool, to construct prediction models for students' academic achievement. Experimental evaluations are conducted to determine the most effective model, with a focus on accuracy. Furthermore, the study emphasizes the significance of utilizing the extracted knowledge to profile students and assess their likelihood of success in the first semester. This research contributes to the field of educational data mining, offering insights that can potentially enhance student outcomes in computer science education**.**

Keywords: *Educational Data Mining, Decision Tree, Hybrid Algorithm, Support Vector Machine, Prediction, students' academic performance.*

I.Introduction

The ever-growing data deluge demands sophisticated analysis to unlock valuable insights. Data mining, or knowledge discovery from data (KDD), tackles this challenge by extracting hidden patterns and knowledge from massive datasets.[1] This powerful technique finds applications in education, economics, business, and medicine, to name a few. In the field of education especially, there's a growing interest in leveraging data mining techniques to improve learning experiences.

Educational Data Mining (EDM) dives into the rich pool of educational data, analyzing student performance, interactions, and course materials. By identifying patterns within this data, EDM helps pinpoint areas for improvement in teaching and learning. This study focuses on applying three classification techniques: Support Vector Machine (SVM), Decision Tree (DT), and a hybrid model combining SVM with Convolutional Neural Network (CNN) [2]. These algorithms will be utilized to develop models for predicting Students' Academic Performance (SAP).By analyzing the patterns extracted from these models, educators can gain deeper insights into student academic performance data. This knowledge can then be used to improve educational outcomes and optimize student learning. Data mining techniques in education essentially act as a tool to uncover actionable insights. These insights can then be used to drive improvements in teaching methodologies, ultimately leading to greater student success.

This research analyzes academic performance data for 240 second-graders from five college classes. The dataset is divided for model development and evaluation: 60 samples reserved for testing and 180 for training. This split ensures sufficient data for both training robust models and evaluating their generalizability to unseen data. Partitioning the data like this strengthens the study's methodology by promoting reliable and effective predictive models. By analyzing the patterns extracted from these models, educators can gain deeper insights into student academic performance data. This knowledge can then be used to improve educational outcomes and optimize student learning. Data mining techniques in education essentially act as a tool to uncover actionable insights. These insights can then be used to drive improvements in teaching methodologies, ultimately leading to greater student success.

The rest of the paper is organized as surveys: Section 2: Related Works: This unit delves into existing literature and research in the field of educational data mining. It delivers summary of previous educations, methodologies, and findings related to the forecast of academic presentation and other relevant topics. Section 3: Methodology: Here, the paper outlines the methodology proposed for predicting the academic achievement of first-year bachelor's students.Section 4: Experiments and Results: In this

[1]*Research Scholar, Department of Computer Science Karpagam Academy of Higher Education Coimbatore 641021*
[2]*Professor, Department of Computer Applicatios.Karpagam Academy of Higher Education Coimbatore 641021*

section, the testsshowed to evaluate the planned methodology are described. The results obtained from applying different classification techniques to the students' dataset are presented and analyzed. This section also discusses the performance metrics and the effectiveness of the prediction models developed. Section 5: Conclusion and Future Work: The paper determines with a swift of the key answers and remarks. Additionally, it outlines potential areas for future research and improvement in the proposed methodology. This section may also highlight any limitations encountered during the study and suggest avenues for further exploration and refinement.

II. Literature Review

Aggarwal et al. [3] conducted a significant study on predicting student outcomes using machine learning. Their research delved beyond just academic performance, incorporating non-academic factors like demographics and student activities. This holistic approach involved analyzing data from over 6,800 students at an Indian technical college. The data encompassed student demographics, course information, and their participation in various activities. One challenge they addressed was the potential for imbalanced data, where certain categories (e.g., low performers) might be underrepresented. To address this, they employed a technique called synthetic minority oversampling, which essentially creates artificial data points to balance out the dataset. Their findings were promising, with various machine learning algorithms achieving high performance (F1 scores between 90.3% and 93.8%). Interestingly, the study also highlighted the importance of considering socio-economic background alongside academic factors when predicting student success. This suggests that factors beyond academics can significantly influence a student's performance..

Zeineddine's study [4] explored the potential of Automated Machine Learning (AutoML) to enhance student performance prediction. This approach leverages existing student data, in this case characteristics collected before a new curriculum, to build optimal prediction models. The results were promising, with AutoML achieving a significant reduction in false predictions while maintaining high accuracy (75.9%) and a Kappa value of 0.5 (indicating moderate agreement between predictions and actual outcomes). This suggests Auto ML is a valuable tool for researchers, particularly when working with pre-existing student data.The study goes beyond prediction to highlight the potential for proactive support. By leveraging pre-admission data, educators can identify students at risk before the new curriculum begins. This allows for early intervention and targeted support, such as consultation sessions, to improve their chances of success. Furthermore, Zeineddine addressed data imbalance, where some performance categories might be underrepresented, by employing SMOTE (Synthetic Minority Over-sampling Technique) preprocessing. This technique creates synthetic data points to balance the dataset and improve the accuracy of predictions. Overall, the study highlights the promise of AutoML and early intervention strategies for enhancing student success

Bueno-Fernández [5] championed the use of machine learning (ML) to predict student grades using past performance data. Their study, conducted in computer engineering departments of Ecuadorian universities, focused on gathering a large volume of high-quality data. This data, after processing, could be transformed into valuable educational tools. The emphasis on data quality highlights its importance in building reliable ML models for predicting student outcomes.

The focus on e-learning systems within Educational Data Mining (EDM) is a growing trend. Hussain's research [6] exemplifies this by applying machine learning to predict student difficulties on the DEEDS e-learning platform. By analyzing student engagement data, EDM can identify patterns that signal potential problems. This valuable information can then be used to inform and enhance instructional decisions, ultimately improving the learning experience for students using the platform.

Alhusban's research [7] utilized machine learning (ML) to tackle student retention in higher education. The study, conducted at Al Al-Bayt University, focused on analyzing a wide range of factors influencing retention. This included student demographics like gender, academic background (entrance marks, courses taken), and even personal details like marital status. To manage this vast amount of data, the researchers employed Hadoop, a powerful open-source platform designed for processing large datasets using ML techniques. Interestingly, their findings revealed a strong correlation between performance on the admissions exam and a student's chosen field of study. This suggests potential benefits of tailoring academic advising or course recommendations based on these factors to improve student fit and ultimately, retention rates.

Msaci's study [8] delved into factors influencing student performance on the Programme for International Student Assessment (PISA) 2005. Using machine learning, the research analyzed data from various countries like the US, UK, and Japan. The goal was to identify key factors impacting student achievement, encompassing both individual student characteristics and the school environment. By understanding these influences, educators and policymakers can develop more targeted strategies to improve student outcomes across different educational contexts.

Alowibdi's research [9] investigated the use of advanced learning analytics to predict student success in Pakistan.

The study compared various machine learning models, including probabilistic models (Bayes Network, Naive Bayes) and discriminative models (CART, SVM, C4.5 decision tree). To assess the effectiveness of these models in predicting student outcomes, the study employed evaluation metrics like precision, recall, and F-score. By comparing these models, Alowibdi aimed to identify the most accurate technique for predicting student success in the Pakistani educational context.

Al-shehri's study [10] compared the effectiveness of two machine learning algorithms for predicting student performance. They analyzed data from 395 students at two Portuguese schools, encompassing 33 characteristics like grades, demographics, and teacher reports. These characteristics were converted from categorical (e.g., high/low grades) to numerical values for analysis. The study focused on comparing Support Vector Machine (SVM) and K-Nearest Neighbors (KNN) algorithms, both supervised learning techniques commonly used in classification tasks. By evaluating these models, the research aimed to identify which algorithm best predicts student performance based on the specific data collected at the University of Minho.

In their work, Xu [11] introduced a novel machine learning (ML) technique characterized by two distinct features. Firstly, the proposed method involved a multi-tiered framework for forecasting, which accounted for pupils' dynamic performance behaviors. This framework incorporated multiple bases and ensemble classifiers in its layered structure. The multi-tiered approach allowed for a comprehensive analysis of various factors influencing student performance, enabling more accurate predictions. Secondly, the technique employed a data-driven strategy to determine lecture topics, enhancing the relevance and effectiveness of instructional content.

III. Methodology

The methodology for predicting and analyzing students' academic performance using data mining techniques involves a systematic approach to collecting, preprocessing, and analyzing various data sources related to students' academic records and demographics. Key steps include data cleaning, feature selection, model selection, training, and evaluation. By employing appropriate data mining algorithms such as Decision Trees, Naive Bayes, or Neural Networks, educators and researchers can uncover patterns, correlations, and predictive insights that inform strategies for improving educational outcomes. This methodology aims to leverage the power of data mining to enhance our understanding of the factors influencing students' academic success and to provide valuable insights for personalized interventions and support.

a)Support Vector Machine (SVM)

Support Vector Machines (SVMs) are powerful supervised learning algorithms that excel at classifying data. They achieve this by finding the optimal hyperplane, a decision boundary that maximizes the separation between different data classes. This focus on margin maximization allows SVMs to be robust to noise and outliers in the data, making them a popular choice for various classification tasks.

Classification and Regression: SVMs can be used for both classification tasks (separating data into categories) and regression tasks (predicting continuous values).

Hyperplane: The concept of a hyperplane as a decision boundary in different dimensions.

Maximizing Margin: The focus on maximizing the margin to improve generalization and reduce sensitivity to noise**.**

Support Vectors: These are the MVPs of the SVM world! Imagine them as the training data points closest to the decision boundary. They play a crucial role in defining the hyperplane's position and orientation, essentially shaping the entire classification model.

Kernel Trick: Real-world data often isn't perfectly separable with a straight line. The kernel trick is a clever way for SVMs to handle this. It essentially transforms the data into a higher-dimensional space where a clear separation becomes possible using a linear hyperplane. Think of it like unfolding a wrinkled map to reveal a clear path – the kernel trick does the same for complex data!

C Parameter: This parameter acts like a balancing act coordinator in SVMs. It controls the trade-off between maximizing the margin (keeping the decision boundary away from the data points) and minimizing classification errors (correctly classifying the training data). A smaller C value prioritizes a wider margin but might miss some classifications, while a larger C value focuses on accuracy but could lead to a tighter margin and higher sensitivity to noise.

Types of SVM Kernels:

Linear Kernel: This is the simplest case, working best when your data is already linearly separable in its original form. Imagine a perfectly straight line dividing apples from oranges – that's the linear kernel in action!

Polynomial Kernel: This kernel is like a magnifying glass for more complex data. It creates a higher-dimensional space by multiplying your existing features, allowing for curved decision boundaries. Think of it as bending a flat map to create valleys and hills, separating your data points more effectively.

Radial Basis Function (RBF) Kernel: This is a highly versatile and popular choice, especially for non-linear data.

It uses a bell-shaped function to transform the data, creating smooth decision boundaries. Imagine a flexible sheet draped over your data, able to conform to various shapes and effectively separate the classes.b) Decision Trees (DT)

b) Decision Trees

Decision Trees are like choose-your-own-adventure stories for machine learning! These versatile algorithms tackle both classification (sorting things into categories) and regression (predicting continuous values). Imagine a flowchart, where each decision you make leads you down a different path. Decision Trees work similarly, recursively splitting the data based on features until they arrive at a final prediction.

Tree Structure: Think of a family tree, but instead of relatives, it has nodes and branches. Internal nodes represent decisions based on features, branches represent the outcomes, and leaf nodes hold the final predictions.

Asking Questions: At each internal node, the tree asks a question about a specific feature. For example, "Is the weather sunny?" Depending on the answer, the data gets split into branches, like "Yes" leading to the "beach" branch and "No" leading to the "park" branch.

Reaching the Finish Line: Leaf nodes are the end of the line, where the tree makes its final prediction. This could be a category label ("beach day!") or a numerical value (predicted temperature).

Making Smart Splits: The tree doesn't split data randomly. It uses a "splitting criteria" like a good judge, aiming to create the most distinct groups at each step. Common criteria include "Gini impurity" for classification (keeping things similar within each group) and "mean squared error" for regression (minimizing the difference between predictions and actual values).

Growing Wisely: Decision Trees can grow quite large, capturing complex relationships in the data. But just like a story with too many twists, a very big tree can overfit and become unreliable. Techniques like "pruning" help control the size and prevent overfitting.

C) Support Vector Machine(SVM)+ Convolutional Neural Network (CNN)

In the context of predicting and analyzing students' academic performance, a hybrid approach combining Convolutional Neural Network (CNN) and Support Vector Machine (SVM) could be beneficial. Here's how such an architecture might be structured:

Firstly, the CNN component would be responsible for processing and extracting relevant features from various sources of data related to students' academic performance. This could include textual data from essays or assignments, numerical data such as test scores or grades, and even multimedia data like images or videos of classroom activities. The CNN would utilize convolutional and subsampling operations to learn hierarchical representations of these diverse types of data. For textual data, it might employ techniques like word embeddings or recurrent neural networks to capture semantic information. For numerical data, it could use standard convolutional layers or pooling operations to identify patterns. And for multimedia data, it might employ convolutional layers specifically designed for image or video processing.

Once the CNN has extracted meaningful features from the input data, these features would be passed to the SVM component. The SVM would then take on the task of classifying or predicting students' academic performance based on these learned features. It would leverage its ability to handle high-dimensional data and perform accurate classification tasks to make predictions about students' future grades, likelihood of passing or failing, or other relevant metrics.

The integration of CNN and SVM in this architecture allows for a comprehensive analysis of students' academic performance by leveraging both the feature extraction capabilities of CNN and the classification power of SVM. By combining these two methodologies, the hybrid model aims to achieve more accurate and robust predictions, ultimately providing valuable insights for educators, administrators, and policymakers to support students' learning and success.

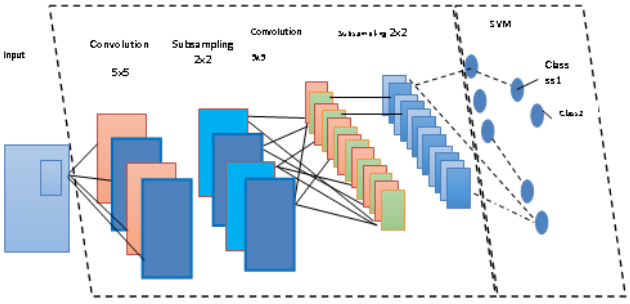

Fig.1 Architecture of proposed Hybrid CNN-SVM Classifier

Input: D = [X, Y]; X (array of input with m features), Y (array of class labels)

Y = array(C)// Class label

Output: Performance of the system

function train_svm(X, Y, number_of_runs)

Step 1: initialize: w, learning_rate = random_values()

Step 2: for run in 1 to number_of_runs

Step 3: error = 0

Step 4: CNN_features = extract_features(X) // Implement a CNN feature extraction function

Step 5: for i in range(1, len(CNN_features))

Step 6: if (Y[i] * (dot_product(CNN_features[i], w))) < 1 then

Step 7: update: w = w + learning_rate * ((CNN_features[i] * Y[i]) * (-2 * (1 / number_of_runs) * w))

Step 8: else

Step 9: update: w = w + learning_rate * (-2 * (1 / number_of_runs) * w)

Step 10: end if

Step 11: end for

Step 12: end for

Step 13: return w

Step 14: X_train, Y_train = load_training_data() // Load training data

Step 15: number_of_runs = 1000

Step 16: trained_weights = train_svm(X_train, Y_train, number_of_runs)

Step 17: X_test, Y_test = load_testing_data() // Load test data

Step 18: accuracy = evaluate_svm(trained_weights, X_test, Y_test)

Step 19: end

Hybrid Model –Algorithm

IV.Experiments And Results

a) Environment

The implementation was carried out in Python, a scientific programming language, on a system with an i7 processor and 16GB RAM running Windows 10. comparisons.

b) Evaluation Measures

To assess the effectiveness of our classification models, we employed four key metrics: Accuracy, Precision, Recall, and F-Measure. These metrics provide a comprehensive picture of the model's performance, going beyond just the overall correct predictions (Accuracy). We'll delve deeper into each metric to understand how well the models identified true positives and negatives.

Accuracy: Accuracy is a metric used in classification tasks to measure the overall effectiveness of a model. It represents the proportion of instances the model correctly classified compared to the total number of instances. In simpler terms, it reflects the percentage of predictions that were accurate.

$$Accuracy = \frac{Total\ Number\ of\ Predictions}{Number\ of\ Correct\ Predictions}$$

Precision: Precision measures the ratio of true positives to all positive predictions. It essentially tells us what proportion of instances labeled as positive by the model are actually correct. In other words, it reflects the accuracy of the model's positive identifications.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

Recall: Recall, also known as sensitivity, focuses on the model's ability to identify all relevant cases. It's calculated as the proportion of true positive instances (correctly classified positive cases) divided by the total number of actual positive instances in the data. In simpler terms, Recall measures how well the model avoids missing true positive cases.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

F-Measure (F1 Score):F-measure strikes a balance between precision and recall. It's a single metric calculated as the harmonic mean of both, offering a more comprehensive view of a model's performance compared to relying solely on precision or recall. The harmonic mean emphasizes instances where both precision and recall are low.

$$F - Measure = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

c) Results

In the analysis, the performance of three classification techniques on the dataset was evaluated. The obtained results revealed notable differences in the effectiveness of these techniques. The proposed SVM+CNN model emerged as the top performer, achieving an accuracy of 97.85%. This implies that 169 out of 173 instances were correctly classified. Moreover, SVM+CNN exhibited the highest precision at 95.47%, indicating its capability to accurately identify relevant instances among the retrieved ones. In terms of recall, SVM+CNN excelled again, with a score of 97.69%, suggesting its ability to capture a high proportion of positive instances. The F-Measure, which associations precision and recall, yielded a strong result of 96.58% for the Hybrid technique. These findings underline the robustness of the Hybrid model in classification tasks. Comparative analysis with Decision Trees (DT) and Support Vector Machine (SVM) algorithms further reinforced the superiority of SVM+CNN. DT trailed with an accuracy of 92.59%, while SVM exhibited the lowest accuracy at 86.82%. Figures depicting confusion matrices offered detailed visual representations, confirming the accuracy and reliability of the models utilized in this study. Overall, the comprehensive evaluation underscores the

effectiveness of the Hybrid technique in accurately classifying instances within the dataset.

Table 1. Estimate Measures of (SVM+CNN, SVM and DT)

Evaluation Measures	SVM+CNN	SVM	DT
Accuracy	97.85	86.82	92.59
Recall	97.69	86.8	92.6
Precision	95.47	85.4	90.5
F-Mesure	96.58	85.9	91.5

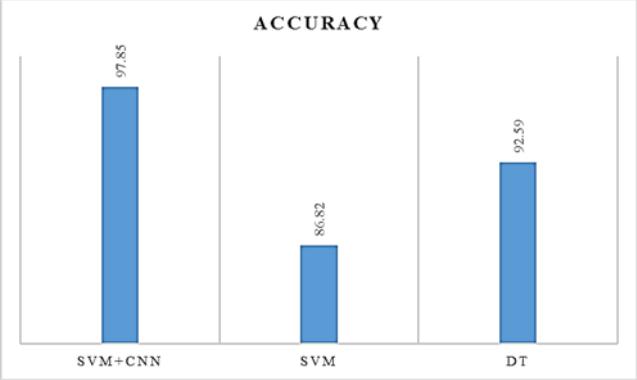

Fig. 2. Accuracy of the three algorithms (SVM+CNN, SVM and DT)

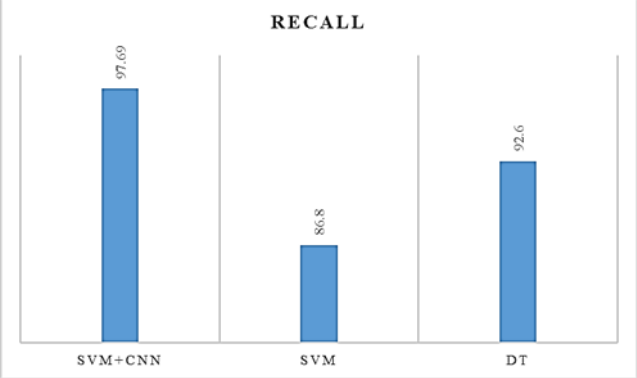

Fig. 3. Recall of the three algorithms (SVM+CNN, SVM and DT)

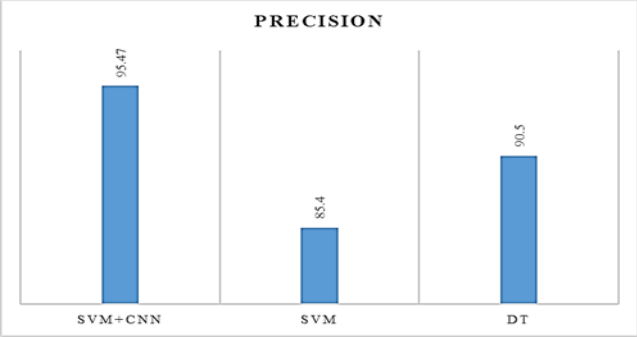

Fig. 4. Precision of the three algorithms (SVM+CNN, SVM and DT)

V. Conclusion

Due to the exponential growth of data, data mining techniques have become indispensable for uncovering valuable insights. In particular, classification methods play a crucial role in identifying contributing factors to students' performance. This study conducted a comparative analysis of three classification algorithms: SVM+CNN (Support Vector Machine with Convolutional Neural Network), SVM, and DT (Decision Tree) using Python. The experimental results demonstrated that SVM+CNN outperformed SVM and DT in terms of classification accuracy. The developed model holds promise for guiding educators in taking proactive measures to support underperforming and average students, thereby facilitating their academic improvement.However, it's significant to accept the limits of this study, notably the small size of the dataset due to incomplete and missing values. Future research endeavors could address this limitation by expanding the dataset with additional data from various years or by incorporating more limitations to enhance the predictive power of the model. Additionally, exploring alternative data mining techniques such as genetic algorithms, Deep Convolutional Neural Networks (DCNN), K-Nearest Neighbor (KNN), and others could further enrich the analysis and potentially uncover deeper insights into students' performance patterns. This iterative approach towards refining the model and exploring diverse methodologies holds the potential to advance our understanding of educational dynamics and improve intervention strategies for students' academic success.

References

[1] Han, Jiawei, Jian Pei, and Hanghang Tong. Data mining: concepts and techniques. Morgan kaufmann, 2022.

[2] Yağcı, Mustafa. "Educational data mining: prediction of students' academic performance using machine learning algorithms." Smart Learning Environments 9, no. 1 (2022): 11.

[3] Aggarwal, Deepti, Sonu Mittal, and Vikram Bali. "Significance of non-academic parameters for predicting student performance using ensemble learning techniques." International Journal of System Dynamics Applications (IJSDA) 10, no. 3 (2021): 38-49.

[4] Zeineddine, Hassan, Udo Braendle, and Assaad Farah. "Enhancing prediction of student success: Automated machine learning approach." Computers & Electrical Engineering 89 (2021): 106903.

[5] Buenaño-Fernández, Diego, David Gil, and Sergio Luján-Mora. "Application of machine learning in predicting performance for computer

engineering students: A case study." Sustainability 11, no. 10 (2019): 2833.

[6] Hussain, Mushtaq, Wenhao Zhu, Wu Zhang, Syed Muhammad Raza Abidi, and Sadaqat Ali. "Using machine learning to predict student difficulties from learning session data." Artificial Intelligence Review 52 (2019): 381-407.

[7] Alhusban, Safaa, Mohammed Shatnawi, MuneerBaniYasin, and Ismail Hmeidi. "Measuring and enhancing the performance of undergraduate student using machine learning tools." In 2020 11th International Conference on Information and Communication Systems (ICICS), pp. 261-265. IEEE, 2020.

[8] Masci, Chiara, Geraint Johnes, and TommasoAgasisti. "Student and school performance across countries: A machine learning approach." European Journal of Operational Research 269, no. 3 (2018): 1072-1085.

[9] Daud, Ali, NaifRadiAljohani, RabeehAyazAbbasi, Miltiadis D. Lytras, Farhat Abbas, and Jalal S. Alowibdi. "Predicting student performance using advanced learning analytics." In Proceedings of the 26th international conference on world wide web companion, pp. 415-421. 2017.

[10] Al-Shehri, Huda, Amani Al-Qarni, Leena Al-Saati, ArwaBatoaq, Haifa Badukhen, Saleh Alrashed, Jamal Alhiyafi, and Sunday O. Olatunji. "Student performance prediction using support vector machine and k-nearest neighbor." In 2017 IEEE 30th canadian conference on electrical and computer engineering (CCECE), pp. 1-4. IEEE, 2017.

[11] Xu, Jie, KyeongHo Moon, and Mihaela Van DerSchaar. "A machine learning approach for tracking and predicting student performance in degree programs." IEEE Journal of Selected Topics in Signal Processing 11, no. 5 (2017): 742-753.