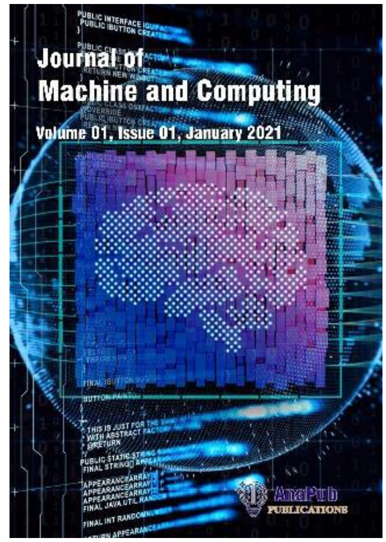
Journal Pre-proof



Beyond the Gradebook: Machine Learning and LMS Data for True Student Performance

Varsha Ganesh and Umarani S

DOI: 10.53759/7669/jmc202505042

Reference: JMC202505042

Journal: Journal of Machine and Computing.

Received 12 July 2024

Revised form 30 November 2024

Accepted 13 December 2024

Please cite this article as: Varsha Ganesh and Umarani S, “Beyond the Gradebook: Machine Learning and LMS Data for True Student Performance”, Journal of Machine and Computing. (2025). Doi: https://doi.org/10.53759/7669/jmc202505042

This PDF file contains an article that has undergone certain improvements after acceptance. These enhancements include the addition of a cover page, metadata, and formatting changes aimed at enhancing readability. However, it is important to note that this version is not considered the final authoritative version of the article.

Prior to its official publication, this version will undergo further stages of refinement, such as copyediting, typesetting, and comprehensive review. These processes are implemented to ensure the article's final form is of the highest quality. The purpose of sharing this version is to offer early visibility of the article's content to readers.

Please be aware that throughout the production process, it is possible that errors or discrepancies may be identified, which could impact the content. Additionally, all legal disclaimers applicable to the journal remain in effect.

© 2025 Published by AnaPub Publications.

BEYOND THE GRADEBOOK: MACHINE LEARNING AND LMS DATA FOR TRUE STUDENT PERFORMANCE

Varsha Ganesh[1] and S. Umarani[2], *

[1]Department of Computer Science, Faculty of Science and Humanities, SRM Institute of Science and Technology, Ramapuram, Chennai, India.

[2]Department of Computer Science and Applications, Faculty of Science and Humanities, SRM Institute of Science and Technology, Ramapuram, Chennai, India.

[1]varshasivashanker@gmail.com, [2]ravania@gmail.com

Correspondence should be addressed to S. Umarani: ravania@gmail.com

ABSTRACT

Analyzing student's behavior for performance prediction involves examining various data points and indicators to predict academic outcomes. This research uses the learner's activity tracker tool collected dataset, which contains three different sets of features including demographic, academic background, and behavior features. This approach combines these data to predict students' performance based on behavior, providing insights that can help in creating personalized learning experiences. A hybrid ML model is developed to improve the student's performance prediction. The hybrid ML model combines the ensemble feature selector with Optimized with support vector machine (SVM) classifier. The Linear Neural network (NN) model is used to form a three-layered feature extraction (FE) model. This approach uses ensemble feature selector to select the influential features from the Linear NN extracted features map. Finally, an optimized SVM classifier is developed to predict the students' performance. The optimized SVM model uses the GridSearchCV method to optimize the regularization parameter ('C'-value) and kernel options of the SVM model to improve the prediction performance. The performance evaluation analysis shows that the LNN-Ensemble-Optimized SVM based students' performance prediction approach achieves higher accuracy (98.12%), precision (98.51%), recall (99.23%), f-score (98.1[illegible]%) rate than comparison approaches on LMS data.

KEYWORDS:

Deep learning, ensemble feature selector, linear neural network, Machine learning, students' performance prediction, support vector machine

I. INTRODUCTION

Predicting students' performance using ML [1] is common and effective use case for data science. It helps educators, administrators, and policymakers to identify students at risk and tailor interventions. The performance analysis model use learning management system (LMS) [2] tools extracted data. The LMS data is a very relevant and impactful use case for ML. The platform collects vast amount of data related to student's activities, engagement, and interaction with course materials. LMS platforms (like Moodle, block board, or canvas) typically log various types of data that can help in predicting student performance and behavior. Some common type of data includes performance data, behavioral data, and democratic data (e.g., students attribute and instructor feedback: gender, age, academic history, teacher's evaluations report, etc.). This data can be leveraged to predict not only academic performance (e.g., grades) but also behavioral patterns (e.g., student's engagement, likelihood of dropout, or

Authors Pre-Proof

participation in activities). The goal is to predict academic outcome such as final grade, pass or fail prediction, and likelihood of completing the course successfully. The goal of behavior prediction is predicting the behavioral patterns predictions such as likelihood of students being inactive, time spent on the LMS [3], identification of the students at risk of dropping out, and participating in forums, assignment, or quizzes. The performance prediction model contains three important data processing stages such as data preprocessing, feature selection, and prediction model training. Some of the common challenges in the students' LMS data analysis are data quality issues' (like noisy or missing data), and model overfitting issues. These challenges can be resolved by adopting suitable preprocessing methods [4] [5], regularization techniques and cross validation. So, this study's goal is to overcome the addressed challenges in the LMS data-based prediction models. This study investigates a multi-step ML pipeline to improve the classification accuracy utilizing optimized feature selection and regularization parameters. This study uses the LMS data to predict students' performance, providing insights that can help in creating personalized learning experiences. A hybrid ML model is developed to improve the student's performance prediction. The hybrid ML model combines the ensemble feature selector with Optimized with support vector machine (SVM) classifier. The Linear Neural network (NN) model is used to form a three-layered NN model. This approach uses ensemble feature selector to select the influential features from the Linear NN extracted features map. Finally, an optimized SVM classifier is developed to predict the students' performance. The optimized SVM model uses the GridSearchCV method to optimize the regularization parameter ('C'-value) and kernel options of the SVM model to improve the prediction performance.

II. RELATED WORK

Dinh Thi Ha et al. (2020) [6] investigates the ML techniques to predict the final Grade Point Average of students based on personal characteristics and academic performance. The data is collected by combining data from a survey of graduate students of three different years and data from the student management information system of university.

Chan. Y et al. (2023) [7] Employed three different types of task-oriented educational data to investigate the performance of ML methods in different application scenarios. Specifically, seven parameter-optimized ML methods are implemented binary and multi-classification predication tasks. The experimental results demonstrate that Random Forest (RF) has achieved superior generality on all selected datasets.

M. Arashpour et al. (2023) [8] developed a hybrid techniques to reliably predict the student exam performance (fail-pass classes and final exam scores). The hybrid method uses support vector machine (SVM) and artificial neural network (ANN). The algorithm carries out the feature selection (FS) process of both ANN and SVM techniques in which the optimal combination of the input variable is determined. Finally, four hybrid models containing anonymized information on both discrete and continuous variable is developed using a comprehensive data set for learning analytics.

Asselman, Amal et al. (2021) [9] focuses on the exploitation of Ensemble Learning methods as an extremely effective ML to create many advanced solutions in several fields. This approach develops a new approach based on different models such as RF, AdaBoost, and XGBoost. The experimental results show that the scalable XGBoost has outperformed the other evaluated models.

Moises Riestra-Gonzalez et al. (2021) [10] uses ML models for the early prediction of students' performance in assessment of LMS assignments. It predicts no detect at-risk, fail and excellent students in the early stages of the course. The ML framework contains Decision tree (DT), nave Bayes (NB),

Authors Pre-Proof

logistic regression (LR), multilayer perceptron (MLP) neural network (NN), and SVM models. The author also uses a clustering algorithm to detect six different cluster of repeated patterns in all the early stages of the course. Finally, the result show that the MLP obtains the best performance (80.1% accuracy).

M. M. Tamada et al. (2021) [11] compared 7 algorithm's performance on LMs data of technical courses, blended and distance learning, at high school to exploring the compromise between early and late detection of at-risk students. The results identify that the RF performs the better on this data while predicting a student's performance.

A. S. aljaloud et al. (2022) [12] determine how certain Key performance Indicators (KPIs) based on student interactions with Blackboard helped to forecast the learning outcomes of students. Designed four deep learning (DL) models for predicting student performance. Correlational analysis is performed to examine the extent to which these factors are linearly correlated with the performance indicators of students. Results indicated that a predictive model which combined convolutional neural network and long short-term memory (CNN-LSTM) is the optimal method among the four models tested. The CNN-LSTM method achieved 90.94% precision using only [illegible] features.

Perkash, A et al. (2024) [13] performs video learning analysis and data mining(DM) approaches to predict student academic achievement and identify the factors affecting their performance. The dataset containing records from SIS, Moodle, and eDify. This study advocates the use of balanced dataset and optimized feature set to obtain better performance for student academic performance (SAP) prediction. Several ML and DL models are applied to analyze their performance against the original dataset, balanced dataset, and balanced dataset with the optimized feature set. Experimental results demonstrate that the DT outperforms with an accuracy of 99.[illegible]% for a balanced dataset and optimized feature set.

Hasan et al. (2020) [14] predicts student's overall performance at the end of the semester using video learning analytics and DM techniques. Data from the student information system, LMS and mobile applications is analyzed using eight different classification algorithms. Furthermore, data transformation and preprocessing techniques is carried. Moreover, genetic search and principle component analysis (PCA) is carried out for reduce the features. Additionally, the CN2 Rule Inducer and multivariate projection can be used to assist faculty in interpreting the rules to gain insights into student interactions. The results showed that RF accurately predicted successful students at the end of the class with an accuracy of 8[illegible].3% with an equal width and information gain ratio.

Villegas-Ch et al. (2020) [15] proposes the integration of technologies, such as artificial intelligence (AI) and data analysis, with LMS in order to improve learning. This objective is outlined in a new normality that seeks robust educational models, where certain activities are carried out in an online mode, surrounded by technologies that allow students to have virtual assistants to guide them in their learning and the model classify the data with 94.1176%.

The enhancing the precision of s predictions by optimizing hyper parameters in ML techniques. In pursuit of optimal performance, a range of ML techniques is compared, and the most accurate one selected for hyperparameter optimization. The adopted method for this optimization is the Grid Search (GS) technique. It is found that hyperparameter optimization in the Gradient Boosting Regression Tree (GBRT) using the GS method bolsters the accuracy of predictions pertaining to SAP. The result using a five-fold cross-validation method.

The developed a flexible feature selection model for student performance prediction in four categories of student performance data. This prediction framework uses two concepts: improving the prediction

Authors Pre-Proof

performance with feature selection and skipping feature engineering. Initially, features are embedded continuously and applied directly on an ANN to perform prediction. The second approach uses all the embedded features reduction with the help of RF before performing the prediction. The evaluation results show that the FS-based model helps the prediction model to obtain a better accuracy of 93% for dropout prediction. This model also acquired 86% and 88% prediction accuracy for Students' pass grade and distinction grade date.

The developed a ML [16] [17] based ensemble model to predict students' performance. This model utilizes the ensemble of DT, K-Nearet neighbour (KNN), extra tree, and NB methods. It uses bagging-based boosting methods for performance prediction. The ensemble model accuracy is improved to 86.83% for the student's performance dataset. The result show that NB performs well compared to other models.

Hussain et al.,2021 [18] prepared an automatic students' marks and grade forecasting framework using ML models. A Genetic Algorithm (GA) selects features from the students' dataset. The GA-selected parts are classified by Regression and DT classifier. The Regression model obtained a reliable accuracy rate 96.64% among these two. Since the data volume has increased, it is getting a big issue. The ML-based model needs to perform more adequately. So, a deep learning-based regression model needs to be integrated.

Saba T et al., 2021 [19] developed an automatic exam monitoring system to assist the instructors in monitoring the students without being present in the exam centers. It builds a DL to form a 46-layered CNN model. The extracted features are used for selecting significant features using Atom Search Optimization (ASO) to improve the prediction performance of variants of SVM and KNN models, and among these KNN model obtained the best accuracy rate (93.88%) than another model.

Varsha ganesh et al. (2024) [20] uses two kinds of dataset contains behavioral and academic student's performance data. the author designed hybrid ensemble model for FE and FS to enhance the prediction rate. The ensemble models contain for methods to construct ensemble of feature set and the ResNet50 is integrated with SVM model to strengthen the training performance. The evaluation analysis demonstrates that the ensemble based SVM classifier obtained maximum of 98.03% for academic performance dataset and 99.06% for activity dataset.

The section reviewed the related recent researches on different ML and DL based SAP approaches. The review helps to identifies suitable methods for each stage of data analysis. The FE phase uses linear NN to extract the reduced feature set and ensemble feature selector is used in FS phase. moreover the SVM model prediction performance is enhanced using Grid search CV optimizers. These methods are identified as the best performing methods for each stage of the students performance prediction model.

III. LINEAR NEURAL NETWORK-ENSEMBLE FS BASED OPTIMIZED SVM APPROACH FOR STUDENTS' PERFORMANCE PREDICTION

Authors Pre-Proof

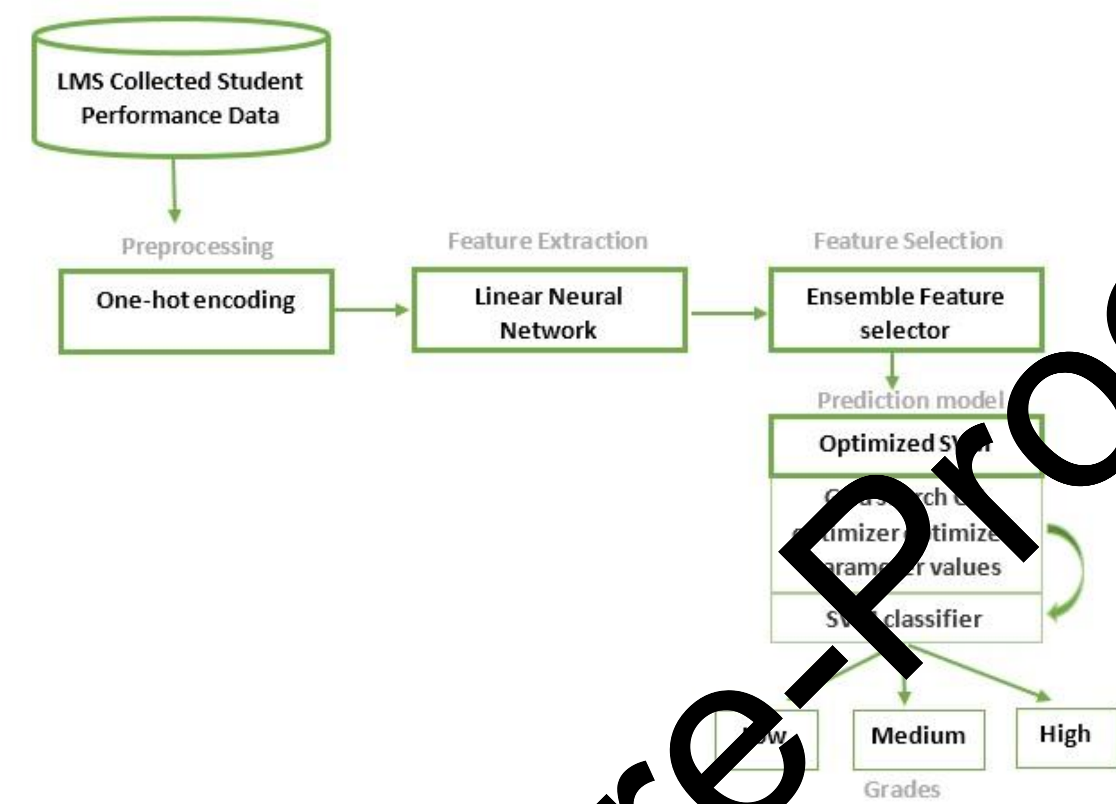


Figure 1: General work flow of the students' performance prediction approach

Figure 1 demonstrates the general workflow of the proposed student's performance prediction approaches. It contains four stage such as (1) preprocessing using one-hot-encoding, feature extracted using (2) Linear neural network, Feature selection using (3) Ensemble based feature selection, Prediction using (4) OptimizedSVM

A. Data source

This section discusses the dataset information used for the performance prediction. The model evaluation is carried out with freely available students' performance dataset. The dataset contains 480 students record and 16 features. The 16 features are categorized as three groups such as democratic (e.g., gender and nationality), academic (grade, section and educational state), and behavioral (e.g., opening resources, parents answer in survey, parent's satisfaction, and raise hands on class) features. The dataset is collected from different origins of students for two educational semesters, and school attendance based on their absence days. The overall dataset is spilt as 80% and 20% for training and testing.

B. Preprocessing

This section discusses the preprocessing methods used in this SAP prediction process. The dataset contains no missing values. So, data normalization technique is used to normalize both categorical and numerical features into normalized form using one hot encoding. The method converts a vector whose elements are only 0's and 1's. It helps to convert each word into unique vectors. The one hot encoding methods increases the dimensionality of the dataset; it may lead model overfitting. So, it is crucial to use dimensionality reduction methods to reduce the issue.

C. Linear NN

Authors Pre-Proof

The LNN for FE in predicting student performance essentially learns linear combinations of the input features to make predictions. It is simple to implement and can be a useful baseline when it is expected that the relationships between the features (e.g., study time, attendance, and performance) are approximately linear. It is essential to transform and combine the input features (such as study habits, attendance, etc.) into a predictive output (like exam scores or pass/ fail status). So, this study utilizes linear neural network (LNN) to transform the data into lower dimension using the feature extraction methods. When using a LNN for FE related to student performance, the process of FE in this case needs to be linear, meaning the model will only perform linear combinations of the input features to make predictions. So, the student's dataset is applied in LNN to perform linear combinations of this input features to make predictions. The target variable defined for the FE are Study Hours, Time spent studying, Attendance Rate, Grades, Family Background, student has access to online study resources, and Grade Category.

The architecture of LNN can be quite simple, it contains input layer, hidden layer and output layer. Each feature is an input to the model input layer. If there are n number of features (e.g., study hours, attendance rate, etc.), then the input layer will have n nodes. Then three hidden layers added, but there is no activation functions (e. g ., ReLU or sigmoid) between layer. Each hidden layer still performs a linear transformation of the inputs. The output layer is depending on the target variable. For classification, it outputs a single value that can be thresholded (e.g., grade category).

Feature Etraction Process: FE in LNN is essentially the process of learning weighted combinations of the input feature. For example: If the network learns that study hours have a higher weight than sleep hours, it is effectively recognizing that study hours contribute more significantly to predicting student performance.

The weights w_1, w_2,… w_n, represent the model's learned understanding of the importance of each feature in predicting student performance. Since the model is linear, it can only combine the features in a straight-forward way, without capturing higher-order or nonlinear relationships between them. The LNN is trained using gradient descent optimization algorithm to minimize the loss function (Binary Cross-Entropy for classification). The objective is to minimize the error between the model's prediction scores and the actual scores (predicted class labels and actual class labels). During training, the weights are updated so that the model's predictions become as accurate as possible. The FE process in this case is just the process of learning weights.

Authors Pre-Proof

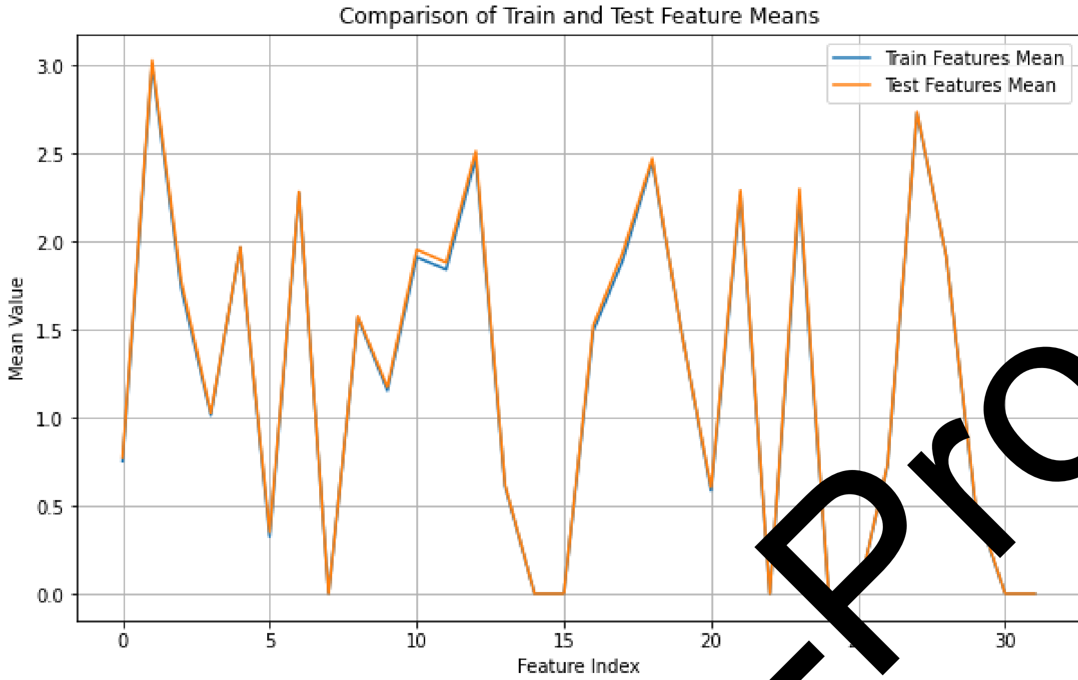


Figure 2: Linear neural network extracted features index's mean value

Figure 2 demonstrates the LNN models feature index and their mean values for training and testing phase. It shows that the both testing phase and training phase achieves better mean values. The LNN extracted features are applied to ensemble feature selector to enhance the dimensionality reduction performance and also to reduce the overfitting issues.

D. Ensemble FS

The feature selection is carried out with the help of ensemble FS method. The adopted ensemble method combines elastic net, recursive feature eliminator (RFE), and hybridization DT, and RF.

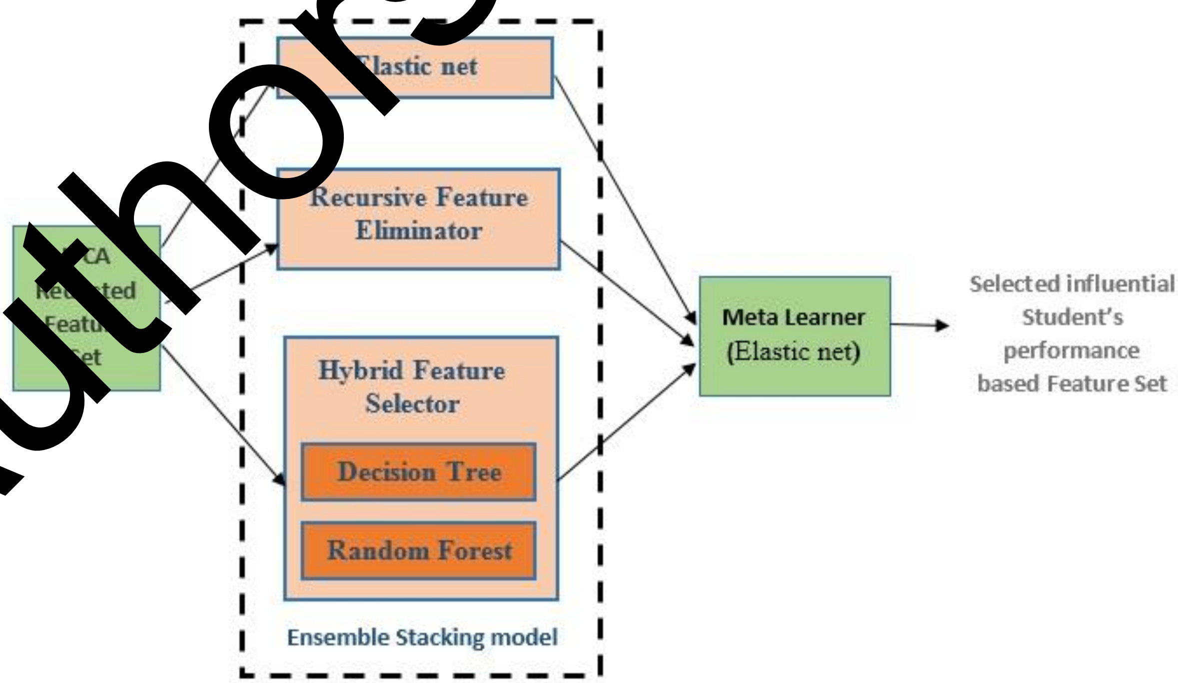


Figure 3 Ensemble Feature selector

Authors Pre-Proof

The Figure 3 illustrates the architecture of the ensemble feature selector-based feature selector method used in this section. The role of each method (ElasticNet, RFE, DT, and RF) used in the ensemble methods is discussed in the subsequent subsections.

a. Elastic net

It is a combination of ridge and lasso regression methods. These two methods are popular for regularizing variants of linear regression. Lesso used the penalty L1, and Ridge used the penalty L2 method. The specialty of the elastic net is using both L1 and L2 penalty regularization.

b. RFE

RFE is a wrapper-type feature selection method. It is in contrast to filter-based feature selection that scores each feature and selects those features with the most significant score. It searches for a subset of features by starting with all feature in the training dataset and successfully removing feature until the desired number remains. It has been used to fit the ML algorithm. Rank features by importance. It gives an external estimator that assigns weights to features. The estimator is trained on the initial set of features, and the features' importance is obtained through any specific attribute. Discard the less critical feature and refit the model. These steps are repeated until the preferred number of features to select is eventually reached.

c. Hybrid method

The hybrid method combines the decision tree (DT) Random Forest (RF) methods. The single DT method is unsuitable for high dimensional data, so the RF method is combined with the DT to improve the performance of the feature selection model.

i) DT

Graphical representation for all possible solutions to a problem based on given conditions. DT is a tree-structured method where internal nodes indicate the dataset's features, branches show the decision rules, and the leaf node indicates the predicted outcome. The decision nodes contain multiple units and make it makes any decisions. It does not have any additional node. It asks questions to split the tree into subtrees based on the answer. The man issue in the DT algorithm is the best attribute selection for root and sub-nodes. It uses two popular to perform the best attribute.

ii) RF

RF is simply a collection of DT whose results are aggregated into one final result. RF is a strong modelling technique and much more potent than a single DT. It aggregates many DTs to limit overfitting and also errors due to bias. It can restrict overfitting without significantly increasing error due to bias. It reduces variance by training of different samples of the data. Another method is using a random subset of features. Each tree can utilize a specified number of random features. More trees in the RF include many of all features. The presence of many features helps in limiting the error due to bias and error due to variance. If features are not selected randomly, then base trees in the forest correlate highly. Since few features are partially predictive, many base trees can choose the same features. Many of these contain the same features; it cannot be combined error due to variance. The ensemble method uses LNN method extracted features of students performance data to evaluate the performance of hybrid ensemble method.

Authors Pre-Proof

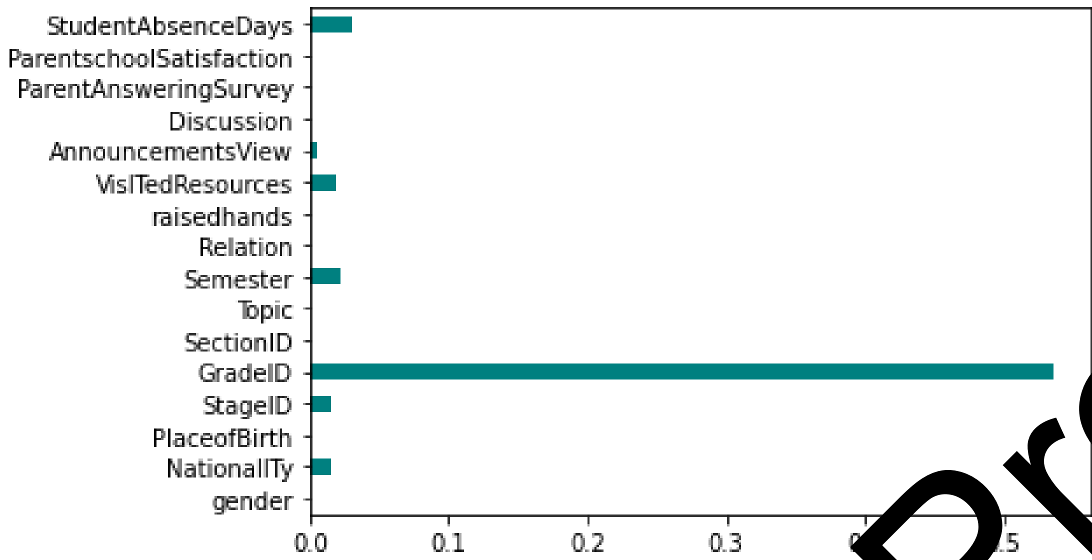


Figure 4: Ensemble FS selected features

The figure 4 represents the selected features among the original dataset features. The original dataset contains 16 student's information related features and the ensemble FS selects seven features among sixteen. The seven selected feature and their feature important score is represented in green color bar.

E. Optimized SVM for grade prediction

The ensemble model selected features are used as the input to train the classification model. This student performance prediction model uses the SVM classifier with optimized parameter to enhance the prediction performance of the student performance analysis model. SVM is a popular ML model for classification and regression problems. It assigns the newly entered sample to one of the trained categories. So, it is called a non-probabilistic binary linear classifier. The classifier efficiently performs the classification task by applying the proper kernel tricks. SVM classifier separates data points with different class labels using a hyperplane with the maximum amount of margin. The hyperplane acts as a decision boundary. Sample data points are called support vectors. This data defines the hyperplane by estimating as a perpendicular distance from the line to data points or SV. The SVM try to improve the separation gap to get the maximum margin. Sometimes, the sample data points are so discrete that it is not conceivable to distinguish using the hyperplane. In such a situation, kernel tricks transform the input space to a higher dimension space. It uses a mapping function to transform the input space. The linear separation method is applied to the data points to separate them. This student performance analysis model uses the linear kernel to map the students' data to higher dimensional data. The regularization parameter C in SVM plays a crucial role in controlling the trade- off between maximizing the margin and minimizing classification error on the training data. In other words, C helps to balance the model's complexity and its ability to generalize. A larger C allows the model to tolerate more classification errors on the training data. It forces the SVM to find a decision boundary that correctly classifies as many training points as possible, even if it means a smaller margin (i.e., it allows fewer margin violations or misclassifications). As a result, a high C leads to less regularization and a model that is more sensitive to the training data (risking overfitting). This might result in a highly complex model that fits the training data will but performs poorly on unseen data. A smaller C allows the model to tolerate more classification errors on the training data. It focuses on maximizing the margin, even if it means misclassifying some data points. A low C leads to more regularization and a model that is more likely to generalize well to unseen date (but might under fit the training data if C is too small). When C is large, the model heavily penalizes misclassifications, pushing the boundary to ensure all data points are classified correctly. When C is small, the penalty for misclassification is

Authors Pre-Proof

lighter, allowing the model to focus more on the general structure of the data and not on fitting data perfectly. The role of C is straightforward in linear SVMs, where it controls the trade-off between margin maximization and error minimization. Small C Encourages a wider margin, allows more misclassification, and helps prevent overfitting (more regularization). Choosing the optimal value for C depends on the nature of the data. For smaller datasets, a high C might perform well, while for larger, noisier datasets, a smaller C may help the model generalize better. So, this study use grid search CV optimization technique to choose suitable and optimal values for the hyper parameters (Kernel and 'C') of the SVM classifier.

Grid search CV for optimize hyper parameter SVM

To use GridSearchCV for hyperparameter tuning of an SVM model for student performance prediction, we can proceed with a dataset that contains student-related feature (e.g., study time, absences, etc.) and the target variable representing performance (e.g., pass/fail, grade prediction). To perform a grid search for hyperparameter optimization in a SVM GridSearchCV function is utilized from Sklearn.model_selection. This technique involves specifying a set of hyperparameters to search over, training the SVM on different combinations of parameters, and evaluating the model's performance using cross-validation.

The param_grid specifies the values used to test for c, kernel, gamma, and degree (for polynomial kernels). C is a regularization parameter. A higher c will penalize misclassification more. The kernel type ('linear', 'poly', 'rbf', or 'sigmoid') determines the decision boundary. Degree is only used for the polynomial kernel ('poly'). It specifies the degree of the polynomial function. Grid Search with Cross-Validation function perform GridSearchCV will search through all combinations of the parameters defined in param_grid, using 5-fold cross-validation (cv = 5). The best combination of parameters is selected based on the cross-validation performance. After finding the best parameters, the model predicts the test set and evaluate it using classification metrics like precision, recall, and F1-score. The subsequent section discusses the performance analysis of the proposed approach on students' performance related LMS data.

15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30

IV. RESULT AND ANALYSIS

This section analyzes the performance analysis of the proposed student performance prediction using LLN-Ensemble-Optimized SVM is evaluated by comparing the various performance wise best ML methods such as MLP [19], CNN-LSTM, RF, ASO-KNN, GA-Regression, and Ensemble-SVM. These comparison methods are considered for analysis based on their better performance on student datasets in recent times. Different evaluation metrics such as accuracy, precision, recall, f-score, are utilized to evaluate the performance and efficiency of the LLN- Ensemble-Optimized SVM method on improving the prediction performance of the prediction model.

Table 1: Parameters of Linear NN-Ensemble-OptimizedSVM for the student performance prediction model

S.No	Parameters	Values
LLN		
1	Epoch	10
2	Batch size	32
3	Learning rate	0.001
GridsearchCV for SVM		
1	C	[1.e-03 1.e-02 1.e-01 1.e+00 1.e+01 1.e+02 1.e+03]

Authors Pre-Proof

2	Kernel	'linear', 'rbf'
3	Fitting 5 folds- totaling -	for each of 14 candidates 70 fits
4	optimized Parameters	'C': 1000.0, 'kernel': 'rbf'
SVM		
1	C	1.0
2	break_ties	False
3	cache_size	200
4	class_weight	None
5	coef0	0.0
6	decision_function_shape	Ovr
7	Degree	3
8	Gamma	Scale
9	Kernel	Rbf
10	max_iter	-1
11	Probability	True
12	random_state	None
13	Shrinking	True
14	Verbose	False

Table 1 contains the parameter values used in the LNN-Ensemble-Optimized SVM approach in each stage.

Table 2: Accuracy rate comparison for three models on the LMS data with and without ensemble model

Accuracy rate			
	Linear NN-Ensemble-OptimizedSVM	**Linear NN-Ensemble-SVM**	**Ensemble-SVM**
With feature selection	98.12	97.78	97.14
Without feature selection	95.15	91.26	77.22

Authors Pre-Proof

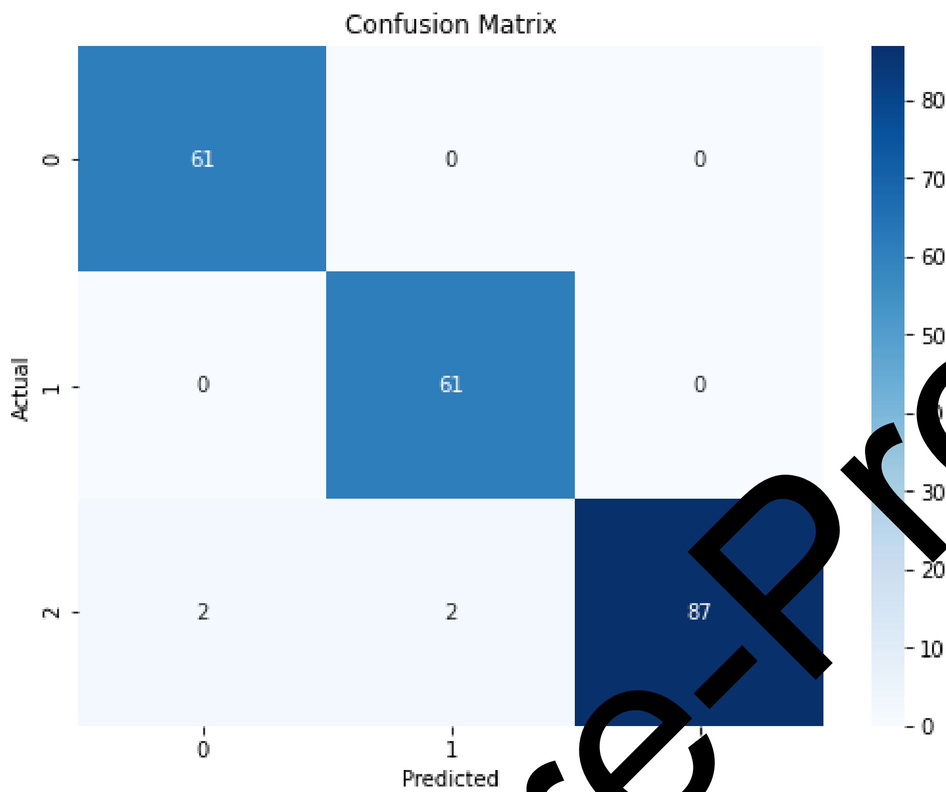


Figure 5: Confusion matrix for the students' performance prediction model

Figure 5 illustrates the Confusion matrix of ELLN-Ensemble-OprimizedSVM model (students' performance prediction model) for LMS data. It is used to evaluate the performance of the model using different accuracy metrics.

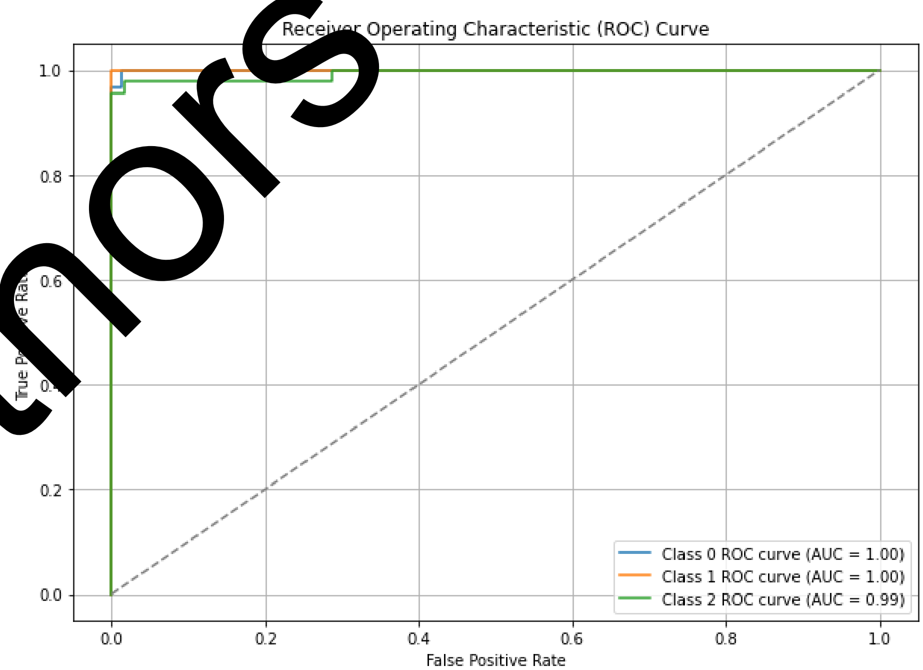


Figure 6: ROC obtained

Figure 6 shows the ROC curve obtained by the proposed approaches. It clearly demonstrates the comparison of false positive and true positive rate.

Authors Pre-Proof

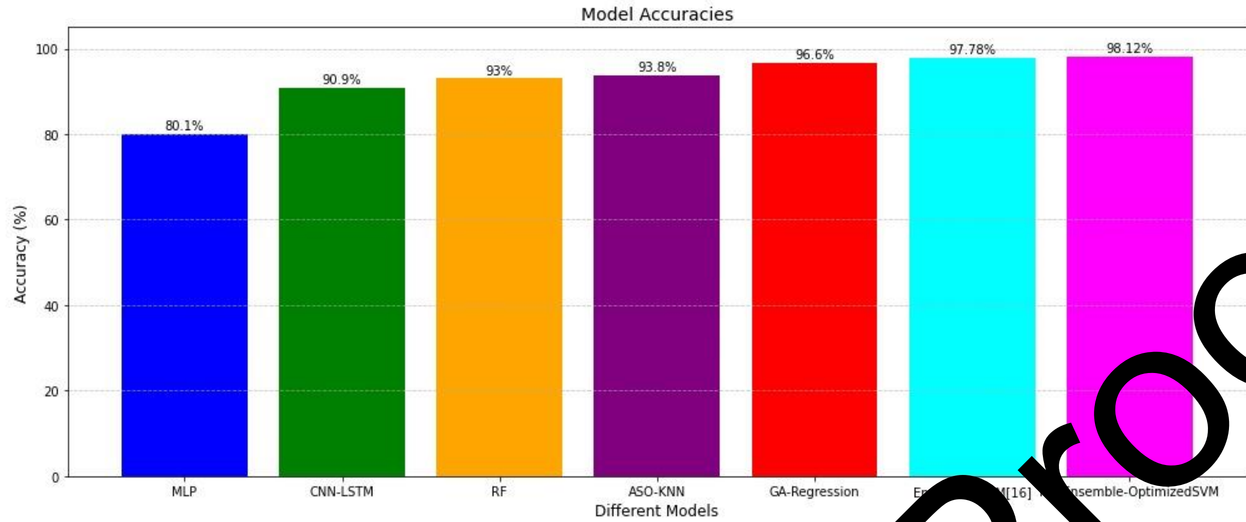


Figure 7: Accuracy rate comparison of different student's performance prediction models.

Figure 7 illustrates the accuracy rate comparison of different ML, hybrid, and DL models-based student's performance prediction approaches. The LinearNN-Ensemble-OptimizedSVM approach obtained maximum of 98.12% as the accuracy rate. The results show that the proposed approach achieves better accuracy rate than comparison model.

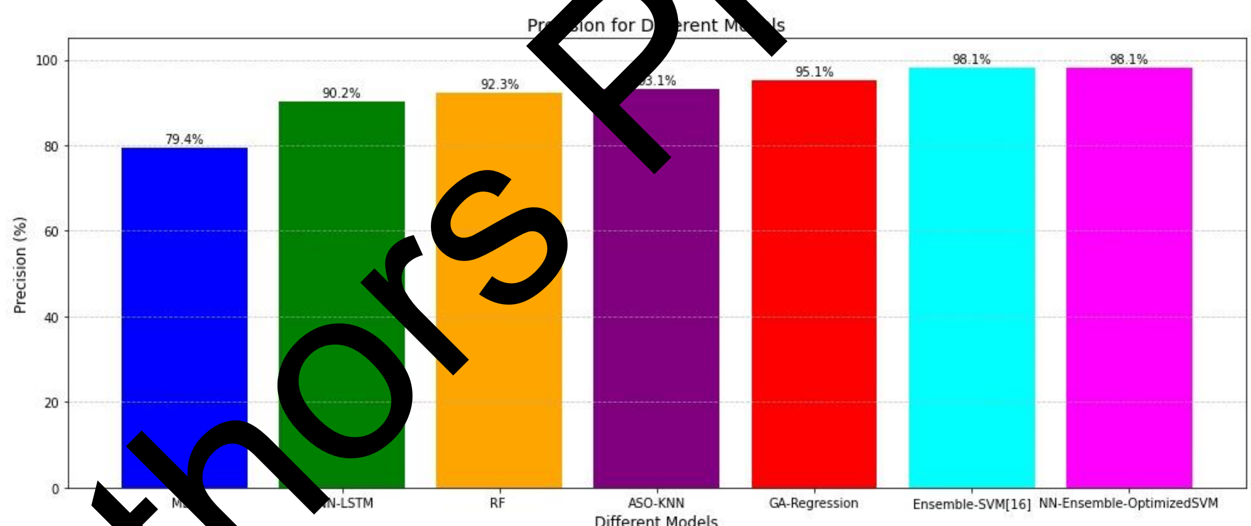


Figure 8: Precision rate comparison of different student's performance prediction models.

Figure 8 illustrates the precision rate comparison of different ML, hybrid, and DL models based student's performance prediction approaches. The LinearNN-Ensemble-OptimizedSVM approach obtained maximum precision rate (98.51%) than comparison model.

Authors Pre-Proof

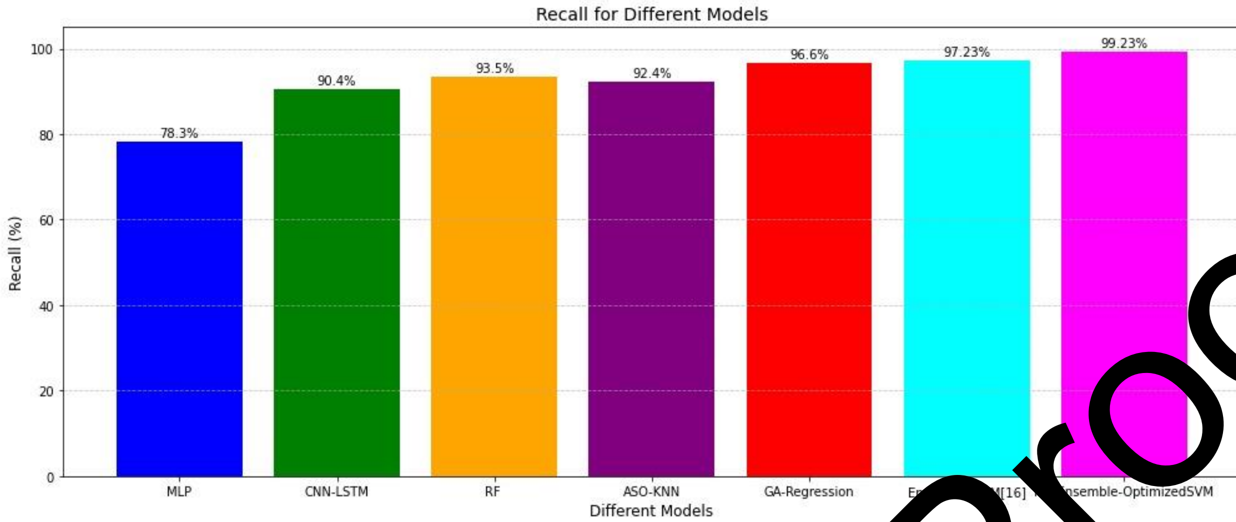


Figure 9: Recall rate comparison of different student's performance prediction models.

Figure 9 illustrates the recall rate comparison of different ML, hybrid, and DL models based student's performance prediction approaches. The LinearNN-Ensemble-OptimizedSVM approach obtained maximum of 99.23% as the recall rate. The results show that the proposed approach achieves better recall rate than comparison model.

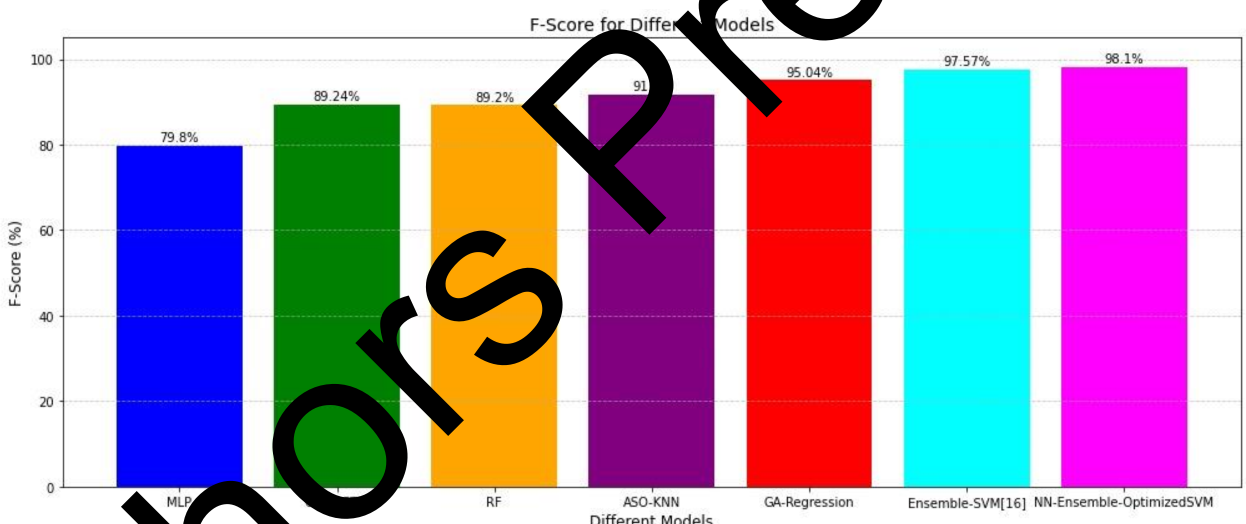


Figure 10: F-score rate comparison of different student's performance prediction models.

Figure 10 illustrates the f-score rate comparison of different ML, hybrid, and DL models-based student's performance prediction approaches. The LinearNN-Ensemble-OptimizedSVM approach obtained maximum f-score rate (98.1%) than comparison model.

Authors Pre-Proof

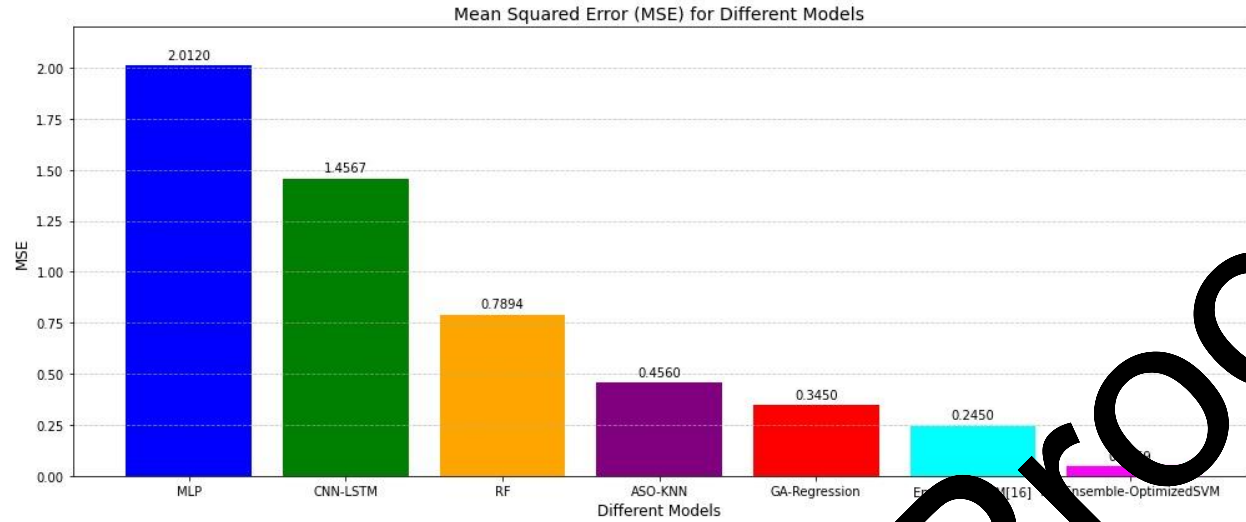


Figure 11: MSE rate comparison of different student's performance prediction models.

Figure 11 illustrates the MSE rate comparison of different ML, hybrid, and DL models-based student's performance prediction approaches. The LinearNN-Ensemble-OptimizedSVM approach obtained minimum of 0.269 as the least MSE rate. The results show that the proposed approach achieves better MSE rate than comparison model.

The time taken to for the LNN-Ensemble-Optimized SVM approach is 67.92(s) without feature selection and 14.84(s) After feature selection, moreover, the approaches utilize 4.23(ms) CPU Without Feature selection and 2.11 (ms) with feature selection.

The overall competence analysis discussed in this section shows that the LNN-Ensemble-Optimized SVM based students' performance prediction approach better than comparison approaches on LMS data.

V. CONCLUSION

Thus the section discusses the conclusion of the research findings. Predicting student performance using ML based approaches and LMS data is a powerful application that can help educators identify at-risk students, optimize learning resources, and improve overall student outcomes. By identifying patterns in how students engage with the material, what factors contribute to success or failure, and how they interact within the course, educators can better support learners. However, it's important to approach such efforts with care to ensure fairness, transparency, and respect for student privacy. The main contribution of the study is improving the overall performance and reliability of student's performance prediction approach. So, this study developed a ML based approach to enhance the students' performance prediction models reliability and performance by adopting appropriate preprocessing and overfitting approaches. The ML model is constructed with Linear NN-ensemble feature selector and Gridsearch CV optimized SVM model (LNN-Ensemble-Optimized SVM). The overall competence analysis discussed in this previous section shows that the LNN-Ensemble-Optimized SVM based students' performance prediction approach achieves higher accuracy (98.12%), precision (98.51%), recall (99.23%), f-score (98.1%) rate than comparison approaches on LMS data. So, the study concluded that the LNN-Ensemble-Optimized SVM approach is suitable for enhance the student performance prediction. Moreover, the proposed students' performance (grade level) prediction system used to personalized recommendation system.

REFERENCE

Authors Pre-Proof

1. Y. Shi, F. Sun, H. Zuo and F. Peng, "Analysis of Learning Behavior Characteristics and Prediction of Learning Effect for Improving College Students' Information Literacy Based on Machine Learning," in IEEE Access, vol. 11, pp. 50447-50461, 2023, doi: 10.1109/ACCESS.2023.3278370.
2. M. Khan, S. Naz, Y. Khan, M. Zafar, M. Khan and G. Pau, "Utilizing Machine Learning Models to Predict Student Performance From LMS Activity Logs," in IEEE Access, vol. 11, pp. 86953-86962, 2023, doi: 10.1109/ACCESS.2023.3305276.
3. Zhu X, Ye Y, Zhao L, Shen C. MOOC Behavior Analysis and Academic Performance Prediction Based on Entropy. Sensors. 2021; 21(19):6629. https://doi.org/10.3390/s21196629
4. N. Raveendhran and N. Krishnan, "A novel hybrid SMOTE oversampling approach for balancing class distribution on social media text," Bulletin of Electrical Engineering and Informatics, vol. 14, no. 1, pp. 638–646, Feb. 2025, doi: 10.11591/eei.v14i1.83[illegible].
5. N. R, N. K, S. R, S. Banu S, S. P, and B. P, "Graph-Based Rumor Detection on Social Media Using Posts and Reactions," International Journal of Computing and Digital Systems, vol. 15, no. 1, pp. 173–182, Jul. 2024, doi: 10.12785/ijcds/160114.
6. Dinh Thi Ha, Cu Nguyen Giap, Pham Thi To Loan, Nguyen Thi Lien Huong, An Empirical Study for Student Academic Performance Prediction Using Machine Learning Techniques, International Journal of Computer Science and Information Security (IJCSIS), Vol. 18, No. 3, March 2020, https://www.researchgate.net/publication/3[illegible]351415
7. Chen, Y., Zhai, L. A comparative study on student performance prediction using machine learning. Educ Inf Technol 28, 12039–120[illegible] (2023). https://doi.org/10.1007/s10639-023-11672-1
8. M. Arashpour, E. M. Golafshani, R. Parthiban, J. Lamborn, A. Kashani, H. Li, and P. Farzanehfar, Predicting individual learning performance using machine learning hybridized with the teaching-learning-based optimization, Comput. Appl. Eng. Educ. 2023; 31: 83–99. https://doi.org/10.1002/cae.22572
9. Asselman, Amal, Mohamed Khaldi, and Souhaib Aammou. 2021. "Enhancing the Prediction of Student Performance Based on the Machine Learning XGBoost Algorithm." Interactive Learning Environments 31 (6): [illegible]60–79. doi:10.1080/10494820.2021.1928235.
10. Moises Riestra-González, Maria del Puerto Paule-Ruíz, Francisco Ortin, Massive LMS log data analysis for the early prediction of course-agnostic student performance, Computers & Education, Volume 163, 2021, 104108, ISSN 0360-1315, https://doi.org/10.1016/j.compedu.2020.104108.
11. M. M. Tamada, R. Giusti and J. F. de Magalhães Netto, "Predicting Student Performance Based on Logs in Moodle LMS," 2021 IEEE Frontiers in Education Conference (FIE), Lincoln, NE, USA, 2021, pp. 1-8, doi: 10.1109/FIE49875.2021.9637274.
12. A. S. Aljaloud et al., "A Deep Learning Model to Predict Student Learning Outcomes in LMS Using CNN and LSTM," in IEEE Access, vol. 10, pp. 85255-85265, 2022, doi: 10.1109/ACCESS.2022.3196784.
13. Perkash, A., Shaheen, Q., Saleem, R. et al. Feature optimization and machine learning for predicting students' academic performance in higher education institutions. Educ Inf Technol 29, 21169–21193 (2024). https://doi.org/10.1007/s10639-024-12698-9
14. Hasan, Raza, Sellappan Palaniappan, Salman Mahmood, Ali Abbas, Kamal Uddin Sarker, and Mian Usman Sattar. 2020. "Predicting Student Performance in Higher Educational Institutions Using Video Learning Analytics and Data Mining Techniques" Applied Sciences 10, no. 11: 3894. https://doi.org/10.3390/app10113894

Authors Pre-Proof

15. Villegas-Ch, William, Milton Román-Cañizares, and Xavier Palacios-Pacheco. 2020. "Improvement of an Online Education Model with the Integration of Machine Learning and Data Analysis in an LMS" Applied Sciences 10, no. 15: 5371. https://doi.org/10.3390/app10155371
16. Krishnan, R., & Durairaj, S. (2024). Reliability and performance of resource efficiency in dynamic optimization scheduling using multi-agent microservice cloud-fog on IoT applications. *Computing*, 1-42. Ali Al-Zawqari, Dries Peumans, Gerd Vandersteen, A flexible feature selection approach for predicting students' academic performance in online courses, Computers and Education: Artificial Intelligence, Volume 3, 2022, 100103, ISSN 2666-920X, https://doi.org/10.1016/j.caeai.2022.100103.
17. Durairaj, S., & Sridhar, R. (2024). Coherent virtual machine provisioning based on balanced optimization using entropy-based conjectured scheduling in cloud environment. *Engineering Applications of Artificial Intelligence*, *132*, 108423.
18. Hussain, S., Khan, M.Q. Student-Performulator: Predicting Students' Academic Performance at Secondary and Intermediate Level Using Machine Learning. Ann. Data. Sci. (2021). https://doi.org/10.1007/s40745-021-00341-0
19. Saba, T., Rehman, A., Jamail, N. S. M., Marie-Sainte, S. L., Raza, M., & Sharif, M. (2021). Categorizing the Students' Activities for Automated Exam Proctoring Using Proposed Deep L2-GraftNet CNN Network and ASO Based Feature Selection Approach. IEEE Access, 9, 47639–47656. doi:10.1109/access.2021.3068223
20. Varsha Ganesh, S Umarani, Ensemble Feature Selection for Student Performance and Activity-Based Behaviour Analysis, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 15, No. 7, 2024, DOI: 10.14569/IJACSA.2024.01507122

Authors Pre-Proof