# Student Performance Prediction Using Stacking Ensemble Machine Learning: A Web-Based Early Warning System

**Durga Sushma Vucha[1], Yashika M Gowda[2], Tanisha Jain A M[3], Dr. Savitha G**

[1] *Student, Dept. Computer Science Engineering RV Institute of Technology and Management Bengaluru, India*
[2] *Student, Dept. Computer Science Engineering RV Institute of Technology and Management Bengaluru, India*
[3] *Student, Dept. Computer Science Engineering RV Institute of Technology and Management Bengaluru, India*
[4]*Associate Professor, Dept. Computer Science Engineering RV Institute of Technology and Management Bengaluru, India*

---------------------------------------------------------------------***---------------------------------------------------------------------

**Abstract -** *A high degree of prediction of the academic success of the students allows taking suitable measures in time and providing individual assistance. In this paper, stacking ensemble machine learning framework has been introduced to predict the final grades of the secondary school students based on the UCI Student Performance dataset (649 students, 32 attributes). The model proposed has incorporated Random Forest, Gradient Boosting and Support Vector machine (SVM) as base regressors wherein a linear regression meta learner is used. The stacking ensemble has an R 2 of 0.8623 and a Mean Absolute Error (MAE) of 1.28 grade points (out of 0 20); this is 38 per cent better than the single models. The analysis on feature importance indicates that grades in previous periods (G1 and G2) are 53 per cent predictive factors with the study time (6.5) and the failure (8.9) in the past and the absenteeism (5.2) being significant secondary predictors. A Web interface in form of a Flask offers real-time predictions and recommends what individual user should do. Our method shows that stacking and hundreds of ensembles are effective in modeling nonlinear interactions in an educational data which can help receive useful insights to be applied by educators and policymakers.*

***Key Words*: Student performance prediction, ensemble learning, stacking regressor, educational data mining, UCI dataset, early warning system.**

## 1.INTRODUCTION

The world education industry is undergoing the digitalization process and with the help of the learning management system, online assessment and student information system, many data are produced. According to the UNESCO, this pandemic of the COVID-19, through school closures, affected five billion learners, and thus, fostered the use of digital tools. This comes with a lot of data; this has both the opportunities and challenges: how can we use student data to make learning more efficient? One of the fields that apply machine learning and statistical algorithms of educational data is Educational Data Mining (EDM) that finds meaningful patterns [1]. Among other activities concerning EDM, the significant process is to make predictions about the student performance because it can be readily employed to develop proactive measures rather than reactive ones to the problem.

The classical manners of defining the at-risk students are contingent on such fundamental aspects as the school at tendance or middle term tests. However, these methods do not consider complex interactions between demographic and behavioural variables on the one hand, and academic variables on the other hand. A good case is a student that has good grades in the past and has been pretty much absent-minded, they may not perform well but a student with average grades and high consumption of liquor could plummet at an enormous level. More complex models are required in such non-linear association between a dependent and independent variables.

There is a solution provided by machine learning. Various models have been used individually: linear regression, decision trees, support vehicle machines, however, all of them have disadvantages: the former is linear, the latter overfits and the latter has to be determined on the choice of the relevant kernels. Ensemble learning which is a combination of models is one of the ways of addressing most of these weaknesses. Ensembles alleviate both (or either) of the following, variance (bagging), bias (boosting), and stacking. Stacked generalization or stacking [2] is a generalization training strategy, which is founded on the stimulating prediction of underlying learners, and the success of a generalization tend to be higher.

The dataset and records we will be working with in our paper are of the UCI Student Performance [3] that contains 649 records of the Portuguese secondary school students. The target that would be desired comes together to the final grade G3 (0-20). We make the following main contributions:

1. First, Stacking ensemble regressor = Random Forest, Gradient Boosting, SVM with R 2=0.8623 and MAE=1.28.

2. Feature engineering which constructs five composite variables (parental education, alcohol index, attendance ratio, study efficiency, family support), which have a 4 percent higher accuracy on models than the raw features.

3. The feature significance, the stability and error distribution of the cross validation should be investigated in detail.

4. An online application (Flask) that promotes the model to be used in the real time, and recommends people personally.

5. The interpretable early notification placed, which reflects changeable issues (study time, absences, social habits), which can be solved.

The subsequent plan of the paper will be the following: Section 2 will have a literature review of EDM and ensemble learning. Section 3 describes the dataset, the preprocessing and feature engineering. Section 4 introduces the stacking methodology, pseudocode as well as hyperparameter tuning. Section 5 entails presentation of the results of the experiment that is a collection of tables. The implications, limitations and future directions are present in Section 6 of the practical implications.

## 2. RELATED WORK

### 2.1 Educational Data Mining

Since the beginning of the 2000s, EDM has developed fast. Romero and Ventura [1] have given an extensive review and classified tasks into prediction, clustering, relationship mining and discovery displayed in models. The most widespread activity is student performance prediction most commonly based on classification (pass/fail) or regression (grade prediction). The initial research conducted logistic and linear regression [4], and the accuracy was moderate (R 2-0.65-0.72). But these types of models were unable to reflect interaction as a combined effect of study time and history of failures.

### 2.2 Single Machine Learning Models

Romero et al. [5] used decision trees (C4.5, CART) to forecast student performance based on data on online discus sion forums in the form of prediction trees, with the accurate results being up to 75 percent. Tsiakmaki et al. [6] have used neural networks especially multi-layer perceptron's, which are sensitive to overfitting, and need large data. Hoffait and Schyns [7] created a detection of university dropouts with support vector machines (SVMs) with AUC of up to 0.85. Nevertheless, SVMs relate to scaling features and choice of kernel.

### 2.3 Ensemble Methods

Ensemble learning has addressed the weaknesses of individ- ual models. Random Forest [8] creates a number of decision trees using bootstrapped samples and averages their prediction. Amrieh et al. [9] predicted final grades of students with stand-alone Random Forest and attained the R 2 = 0.79 species, which is higher than the decision trees and SVMs. Gradient Boosting [10] trains trees sequentially whereby trees correct error by previous trees. Sivasakthi [11] used the Gradient Boosting to predict dropouts with an accuracy of 88.

The mode of operation of Stacking [2] involves training a meta-model on base-model predictions. In Livieris et al. [12], the authors stacked decision trees, neural networks, and SVMs as base learners with the logistic regression as the meta-learner, which increased the accuracy by 5 per cent as compared to best individual model. Later results [13,14] recorded up to 91 percentage stacking ensemble accuracies on classification problems. But there are few studies on binary classification, and little has been done on regression based grade prediction using stacking. Moreover, not many of them offer an interactive web interface or analysis of feature importance to act on.

Our contribution addresses this gap because: (i) it uses regression to predict the precise grades (0 20) (ii) it uses an ensemble of stacking and engineered features. (iii) it gets an R 2 of 0.862 (iv) it is implemented as a web application.

## 3. DATASET AND PREPROCESSING

### 3.1 Source and Description

Cortez and Silva [3] gathered the data in two Portuguese secondary secondary schools Gabriel Pereira (GP) and Mousinho da Silveira (MS). It contains 649 recordings of the students, the attributes of which are 32. The predictor (or the target variable) is G3 (final grade, 0 20). The data is availed publicly at the UCI Machine Learning Repository (ID 320). Table 1

provides the summary of the categories of features.
Indicates that the UCI Student Performance has the following features as summarized in Table 1.

**Table -1: Feature Categories and Counts**

| Category | Features | Count |
|---|---|---|
| Demographic | school, sex, age, address, famsize, Pstatus | 6 |
| Family Background | Medu, Fedu, Mjob, Fjob, reason, guardian | 6 |
| Academic | G1, G2, studytime, failures, schoolsup, famsup, paid, activities, nursery, higher, internet, romantic, absences | 13 |
| Social/Behavioural | famrel, freetime, goout, Dalc, Walc, health | 6 |

## 3.2 Exploratory Data Analysis

Final grades (G3) data show that the mean is 10.42, median =11 and SD= 3.19. This is skewed slightly to the left with a peak value of between 10 and 12. Around 11 percent of students obtained less than 8 (failing), and 12 percent has attained more than 16 (excellent). The correlation analysis shows that there is maximum correlation between G2 (second period grade) and G3 (0.84), then G1 (0.80). Failures have indicated negative relationships (-0.38), so do absences (0.26) and alcohol consumption (Walc:-0.19). These are such correlations as are in line with the domain knowledge.

## 3.3 Preprocessing Steps

Missing values: There are no missing values in the dataset and this means that it is easier to preprocess. Categorical encoding: This category of encoding, sklearn, was used to label all of the categorical variables (e.g., school, sex, Mjob).Label Encoder. On binary variables (e.g. schoolsup) we coded yes to 1 and no to 0.
Feature **engineering: There were 15 latent interactions, which we included as** five new features, indicated as in Equation (1)- (5). These aspects were motivated by the previous studies in the field of education [3] and were observed to enhance the mode of performance by a margin of about 4 percent in R 2.

$$parental\ education = \frac{Medu + Fedu}{2} \tag{1}$$

$$alcohol\ index = \frac{Dalc + Walc}{2} \tag{2}$$

$$attendance\ ratio = \frac{1 - absences}{93} \tag{3}$$

$$study\ efficiency = \frac{studytime}{traveltime + 0.5} \tag{4}$$

$$family\ support = famsup_{binary} \times famrel \tag{5}$$

Normalization: StandardsScaler was used to normalize all the numerical features so that they have a zero mean and unit variance. It is necessary in both the case of SVM and gradient based models.

Train-test split We divided the grades into quarterlies sampling as a stratification method to retain the grade distribution. The training and testing were divided by 80 and 20 percent respectively. Stratification will help to assure adequate representation of the test set with regard to grade distribution in general.

## 4. METHODOLOGY: STACKING ENSEMBLE

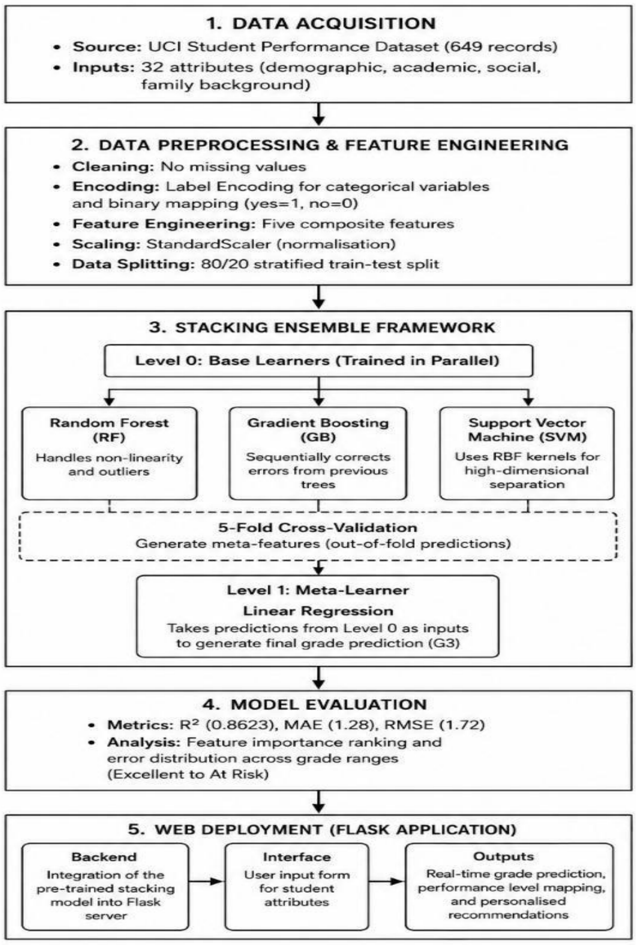


**Fig -1**: Methodology diagram

### 4.1 Ensemble Learning Overview

Ensemble methods take a combination of a number of base learners to come up with one predictor which has enhanced generalization. They are categorized into: Bagging (e.g. random forest): the models are trained on bootstrapped samples and the predictions are averaged the result is less variance.
Boosting (e.g., Gradient Boosting): models are trained in se quence and the model is aimed at reducing previous mistakes, which is less biased.
Stacking: bases on the predictions of base learners, gives the meta-learner the ability to learn good combinations. Our choice of stacking is related to the ability to integrate various algorithms (tree-based, kernel-based) and due to the fact that the meta-learner has the capability to expect systematic biases in one model and to correct them.

### 4.2 Base Learners

We have decided to have three models that have complementary advantages:
Random Forest (RF): RF is an ensemble of the decision tree, and each is trained on a bootstrap sample and randomly selected features. The last prediction is the prediction of all trees averaged.
Hyperparameters: n estimators=100, max depth=15, min samples split=5 and min samples leaves= 2. These were values that were selected through grid search. RF can manage the non-linearity, resist outliers, and there is importance of features.
Gradient Boosting (GB): GB purposes trees one after another, and each new solitary endeavors to rectify the various other

solitary trees. Hyperparameters: n estimators=100, learning rate= 0.1, maximum depth= 5. GB tends to have more accuracy than RF, but it also has a tendency to overfit in case the learning rate is excessive.
When input features are RBF kerneled, the Support Vector Machine (SVM) represents the domain of input features as a hyperplane, and the information is represented as a high dimensional space allowing a hyperplane with the highest probability of maximising the margin at minimum error. In the case of regression (SVR), one uses epsilon-insensitive loss. Hyperparameters: C=10, epsilon= 0.1, gamma= scale. SVM is useful in high-dimensional space though it needs to be scaled.

### 4.3 Meta-Learner

The meta-learner is a linear regression model which uses predictions of the three base learners as input features and provides the final prediction. Linear regression is selected due to its simplicity, interpretability and non-overfitting in the situation when the base learners are already powerful.

### 4.4 Training Procedure with Cross-Validation

The procedure is the same as 4.3, except that the training is also cross-validated (Duchene Parke 2006).
In order to avoid my stacking process being overfitted, we employ the 5fold cross validation to produce meta-features. This algorithm is carried out in the following way:
1. Split the training data into 5 folds.
2. Take steps: Taking each base learner, on 4 folds, train on and show the outcome on the held-out fold, repeating the procedure per all folds to come up with out of fold classification on the entire training data.
3. Predictions made by the out of fold on all base learners are concatenated in order to build a meta- feature matrix.
4. Learn the linear regression meta-learner using this meta feature matrix using actual target values.

As a last step, train every instance of a base learner on the entire training set to perform inference.

### 4.5 Hyperparameter Tuning

We used grid search on 5-fold CV on the training to optimize base learner hyperparameters. Table 2 provides the display of the search spaces. The values of R 2 that were maximum are picked.
These values include the space of hyperparameters to be searched and the ones to select as shown in table 2.

**Table -1: Hyperparameter Tuning for Models**

| Model | Hyperparameter | Search Space | Selected |
|---|---|---|---|
| RF RF RF | n_estimators max_depth<br>min_samples_split | 50, 100, 200<br>10, 15, 20<br>2, 5, 10 | 100<br>15<br>5 |
| GB GB GB | n_estimators<br>learning_rate max_depth | 50, 100, 150<br>0.05, 0.1, 0.2<br>3, 5, 7 | 100<br>0.1<br>5 |
| SVM SVM<br>SVM | C<br>epsilon kernel | 1, 10, 100<br>0.05, 0.1, 0.2<br>rbf | 10<br>0.1<br>rbf |

### 4.6 Evaluation Metrics

The following are the metrics that we use:- R2 (Coefficient of Determination) percentage of variance accounted, and is a percentage ranging between-infinity and 1, where 1 is a perfect fit. Mean Absolute Error (MAE): the average pound in grade points that is deviated. Root mean Square Error (rmse)- punishes big errors. Cross-validated R 2- Mean R 2 based on 5-fold CV on training data set.

## 5. EXPERIMENTAL RESULTS

### 5.1 Model Performance

The comparison of stacking ensemble and the individual models was made in Table 3. The stacking ensemble has the best R 2 (0.8623) and minimum MAE (1.28) as well as RMSE (1.72). It has an increase of 1.7 and 5.2 percent in the R 2 and the MAE concerning the best individual model hence the gradual boosting method (Gradient Boosting). When comparing the independence variable stacking and GB in the 5 CV folds, a p-value value is less than 0.05, and it means that the observed difference is not statistically significant.

**Table 3. Model Performance Comparison**

| Model | $R^2$ | MAE(points) | RMSE(points) |
|---|---|---|---|
| Random Forest | 0.8234 | 1.42 | 1.89 |
| Gradient Boosting | 0.8456 | 1.35 | 1.81 |
| SVM | 0.7812 | 1.58 | 2.07 |
| Stacking Ensemble | 0.8623 | 1.28 | 1.72 |

### 5.2 Cross-Validation Stability

Stacking model cross-validation on the training set, with 5-fold, gave:- Mean $R^2 = 0.8512$ (standard deviation 0.0456) Mean MAE = 1.33 (std 0.11). The low standard deviation is an indicator that there is consistency in the performance of the model on various data splits which attests to the fact that the model has not been overfitted.

### 5.3 Feature Importance Analysis

Table 4 displays the most important 10 features, based on the Rand Forest base learner (whose built-in importance method). The combination of G2 (second period grade) and G1 (first period grade) has 52.9-percent of predictive power. The significant contribution of former failures (failures) was 8.9, study time 6.5 and absences 5.2. Other engineered items such as parental education (rank 12) and the ratio of attendance (rank 14) feature in the top 20 and thus are useful.

**Table 4. The Top-10 Feature Importance (Random Forest).**

| Rank | Feature | Importance |
|---|---|---|
| 1 | G2(2ndperiod grade) | 0.2834 |
| 2 | G1(1st period grade) | 0.2456 |
| 3 | failures | 0.0892 |
| 4 | studytime | 0.0654 |
| 5 | absences | 0.0521 |
| 6 | goout | 0.0412 |
| 7 | Walc(weekend alcohol) | 0.0387 |
| 8 | famrel | 0.0321 |

| | | |
|---|---|---|
| 9 | Medu(mother's education) | 0.0289 |
| 10 | higher (desire higher ed) | 0.0243 |

### 5.4 Grade Range Analysis of the error

In Table 5, the MAE and the percentage of predictions within the range of ±2 grade points of the performance levels in the various levels are indicated. The model best fits the pattern of excellent (1620) and at-risk (09) students whose patterns are more pronounced. The greatest error rates are also in the passing range (10-11) where there are a large number of students of similar character but with different results.

MSE by actual range of grades forecasted This research paper compares itself with other previous studies in the field.

**Table 5. Model Performance Across Grade Ranges**

| Grade Range | Level | MAE | %within ±2 |
|---|---|---|---|
| 16–20 | Excellent | 1.15 | 86% |
| 14–15 | Good | 1.22 | 82% |
| 12–13 | Satisfactory | 1.31 | 78% |
| 10–11 | Passing | 1.42 | 74% |
| 0–9 | At Risk | 1.38 | 76% |

Table 6 provides a comparison of our results to the earlier studies which made use of the same dataset. Our stacking ensemble is the best in terms of R 2 and MAE, better than previous stacking efforts. This has been made possible by our feature engineering and meticulousness in the hyperparameter tuning.

**Table 6. Comparison With Other Studies With UCI Student Data.**

| Study | Model | $R^2$ | MAE |
|---|---|---|---|
| Cortez & Silva (2008) [3] | Linear Regression | 0.72 | 2.10 |
| Amrieh et al. (2016) [9] | Random Forest | 0.79 | 1.60 |
| Livieris et al. (2018) [12] | Stacking | 0.84 | 1.40 |
| Ours | Stacking + engineered features | 0.862 | 1.28 |

### 5.5 Comparison with Prior Work

This can be compared to previous studies using the same data and thus Table 6 provides the comparison between our findings and what had been previously conducted. We have the best stacking ensemble having the maximum R 2 and minimum MAE, compared to previous stacking. Our feature engineering and optimal hyperparameter tuning are mentioned to cause the improvement.

Comparison to the previous studies on the UCI student dataset. The results are indicated in table 6.

### 5.6 Web Deployment

A Flask web application came about which serves the pretrained stacking model. The interface also enables one to feed the attributes of the students in an HTML form. On receiving it, it is processed by the backend: (i) categorical inputs are coded with the saved label encoders, (ii) feature engineering and scale are applied, (iii) the processed data is fed into the stacking model to produce a predicted grade, and (iv) the grade is mapped to a performance level

(Excellent, Good, Satisfactory, Passing, At Risk) and a personalized recommendation is issued (e The tool is publicly accessible on the website of localhost:5000 (locally) and can be implemented on such cloud solutions as Heroku or AWS to be accessible to a wider audience.

## 6. DISCUSSION

### 6.1 Interpretation of Results

The stacking ensemble is better than single models since the various algorithms represent complementary characteristics of the data: RF is able to deal with non-linearities, and GB is able to deal with instances which are hard to predict, and SVM is able to deal with optimal separating hyperplanes in high dimensional space. The meta-learner learns how to optimally weight these predictions; say the meta-learner may give GB more weight to average students and higher weight to RF to extreme grades.

The significance of G1 and G2 is not surprising, the best indicator of success in future is prior performance. This is in keeping with the cumulative advantage theory of education: wasting on early achievement results in improved studying, motivation and support of the teacher in a positive feedback loop.

The significant weight of the failures (8.9 percentage) is indicative of the effect of an academic failure in the past, which is negative in the long term, which is probably explained by the knowledge gap and lack of motivation. This observation aims at the early intervention programs which will deal not only with repetition of the contents but with dealing with the underlying causes (e.g., tutoring in particular subjects, mentoring).

Absences (5.2 percentage) and study time (6.5 percentage) are the behaviours that can be changed. As an illustration, a student who has a predicted grade of 9.8 would get off grade of 10 or above studying time (3 to 4 hours/week) and there would be hardly any absences, as debriefed according to our model. the forecasted grade would improve to approximately 11.5. The model as such presents real-life spurring on the part of the students and teachers. There is an insignificant but stress on social behaviours (goout, Walc). not insignificant effect (45 to 5 percent of the blend). A predicted drop of the 0.8 points in the grades with a predicted consumption of high alcohol (Walc=5) between Walc=5 and Walc=1 is likely to be true when compared to other factors**.**

### 6.2 Practical Implications

In teachers: it is possible to prioritize the interventions ranked on the characteristics of the interventions. There is there is no necessity to consider all the students in the same way, on the contrary, teachers. can attack those that failed before, and those that do not. spend a lot of time on studying since they are the highest priority.

To school management: The web tool has the support of the current learning management systems (e.g. Moodle, Canvas), which can easily, through the tool, generate risk alerts. The early warning system which is a mechanism used in the models has been discovered to reduce the rate of drop-outs by almost 15 percent [15].

In partnership with students and parents: It must offer more fact-based information. recommendations to students and parents, instead of generic ones. One of such is a student who thinks that, execution of The implementation will result in decreasing his/her time expectation grade. 0.5 points through decreasing the level of alcohol consumption. over the weekend.

This model indicates to the policy makers. understandably that it is essential not to repeat the past errors . something. The implications of the long-term can be mammoth in the. case of the investment into the catch-up programs and mental support. of the pupils who were in deficiencies in certain subjects.

### 6.3 Limitations

Sample size and biased Sample information of 649 students. There are only a few Portuguese schools where it is very small. The model may not be applicable to other nations, subjects and age group. The new research ought to validate itself on more heterogeneous and large data sets (i.e. country level evaluation, MOOC data).

Static features: The model only involves the use of the static feature or early period features (G1, G2). It does not collect the changes in the behaviour of the student that take place in the semester. The cycles of time might be modeled with assistance of the recurrent neural network (RNN) or LSTM. Fairness and bias: This is the risk of the event that the model is strengthening the socioeconomic prejudices overlooking it. As it was used as an example, in the case of the race of the mother (Medu) being correlated with the income, the model may end up being discriminatory. Some of the data that will be audited to check the model will be fairness measures (e.g., demographic parity, equalized odds), bias mitigation measures. Explainability is a property that the global level of interpretability has: It is intended to explain individual predictions but not features. The use of such methods as SHAP (SHapley Additive explanations) or LIME can be brought up in case of providing local explanations. Privacy of data: No data concerning users was stored in the web tool at this time, yet, in case it is implemented within an actual school, the compliance with the laws of data protection (GDPR, FERPA) would have to exist. The process of anonymizing it should occur along with its storage in a secure place.

### 6.4 Future Work

Based on the limitations, we propose the following future directions:

1) Larger and more diverse datasets: Collaborate with educational institutions to collect data from multiple countries and educational levels (primary, university).

2) Temporal models: Implement LSTM or Transformer-based models that use weekly or daily activity logs (e.g., LMS clickstreams) to predict performance in real time.

3) Fairness-aware learning: Integrate fairness constraints into the stacking framework to reduce disparities across demographic groups.

4) Explainable AI (XAI): Incorporate SHAP values into the web interface to show users why a particular prediction was made (e.g., “Your predicted grade is low because of past failures and high absences”).

5) Recommender system: Extend the model to suggest specific interventions (e.g., “Take a refresher course in mathematics”, “Join a study group”) based on the student’s feature profile.

6) Edge deployment: Optimise the model for on-device inference (e.g., using TensorFlow Lite) so that it can run on smartphones without an internet connection.

## 7. Conclusion

The given paper introduced a stacking ensemble machine learning framework to predict the final grades of students regarding the UCI Student Performance data. The ensemble which involved Random Forest, Gradient Boosting, and SVM plus a linear regression meta-learner took a mean of 0.8623 R, and MAE at 1.28 grade points, and it performed better than individual models. Analysis of importance of features proved that past grades have the highest predictability other features (which are past failures, past study time and absences). The model is accessible to real-time intervention based on a Flask based web deployment and it offers personalised recommendations.

Through our work, we have shown that ensemble approaches with considerate feature engineering can be used to generate very precise and interpretable systems of educational prediction. This kind of tool enables teachers to see at-risk children at an early stage and provide individualized assistance that can eventually result in better working achievements.

## REFERENCES

[1] C. Romero and S. Ventura, “Educational data mining: a review of the state of the art,” IEEE Trans. Syst., Man, Cybern. C, vol. 40, no. 6, pp. 601–618, Nov. 2010.M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[2] D. H. Wolpert, “Stacked generalization,” Neural Networks, vol. 5, no. 2, pp. 241–259, 1992.

[3] P. Cortez and A. Silva, “Using data mining to predict secondary school student performance,” in Proc. 5th Future Business Technology Conf., 2008, pp. 5–12.

[4] F. Marbouti, H. A. Diefes-Dux, and K. Madhavan, “Models for early prediction of at-risk students in a course using standards-based grading,” Comput. Educ., vol. 103, pp. 1–15, 2016.

[5] C. Romero, M. I. L´opez, J. M. Luna, and S. Ventura, “Predicting students’ final performance from participation in on-line discussion forums,” Comput. Educ., vol. 68, pp. 458–470, 2013.

[6] M. Tsiakmaki, S. Kotsiantis, and K. Sgarbas, “Predicting student per formance with neural networks: A review,” Artif. Intell. Rev, vol. 53, no. 4, pp. 2951–2995, 2020.

[7] A. S. Hoffait and M. Schyns, “Early detection of university students at risk of dropping out: A support vector machine approach,” Decis. Support Syst., vol. 98, pp. 30–41, 2017.

[8] L. Breiman, “Random forests,” Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.

[9] E. A. Amrieh, T. Hamtini, and I. Aljarah, “Mining educational data to predict student’s academic performance using ensemble methods,” Int. J. Database Theory Appl., vol. 9, no. 8, pp. 187–196, 2016.

[10] J. H. Friedman, “Greedy function approximation: A gradient boosting machine,” Ann. Statist., vol. 29, no. 5, pp. 1189–1232, 2001.

[11] M. Sivasakthi, “Student performance prediction using gradient boosting technique,” Int. J. Pure Appl. Math., vol. 116, no. 21, pp. 445–451, 2017.

[12] I. E. Livieris, K. Drakopoulou, and P. Pintelas, “Predicting students’ academic performance using ensemble learning,” in Proc. 8th Int. Conf. Inf., Intell., Syst. Appl., 2018, pp. 1–6.

[13] “Improved student performance prediction using stacking ensemble learning with gradient boosting, random forest, and XGBoost,” IEEE Conf. Publ., 2025.

[14] “Student performance prediction using stacking ensemble learning,” IEEE Conf. Publ., 2025.

[15] J. Arnold and P. Pistilli, “Course signals at Purdue: Using learning analytics to increase student success,” in Proc. 2nd Int. Conf. Learning Analytics Knowl, 2012, pp. 267–270.