International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Beyond Grades: A Machine Learning Approach to Analyze the Impact of Behavioral Patterns on Student Performance.

Prof. Priyanka V H[1], Harshan A [2], Likith D [3], Raja ul haq[4], Sharanyae S S[5]

[1]*Dept. of Computer Science and Engineering & JSS Science and Technology University*
[2]*Dept. of Computer Science and Engineering & JSS Science and Technology University*
[3]*Dept. of Computer Science and Engineering & JSS Science and Technology University*
[4]*Dept. of Computer Science and Engineering & JSS Science and Technology University*
[5]*Dept. of Computer Science and Engineering & JSS Science and Technology University*

---***---

Abstract - Academic success is not only influenced by traditional factors like classroom performance and attendance, but also by lifestyle and behavioral habits such as sleep duration, screen time, and app usage. This research presents Beyond Grades, a system that leverages machine learning models to analyze and predict student performance using both academic and non-academic variables. Data collected from Kaggle was processed and used to train Decision Tree and Random Forest models. The insights were presented through an interactive Flask and Power BI dashboard, enabling students and faculty to visualize the behavioral factors that affect performance. The system achieved an accuracy of 87.2, offering a practical and scalable approach to early academic support. This study aims to highlight the role of behavioral data in improving educational outcomes.

Key Words: Machine Learning, Sleep Efficiency, Student Performance Prediction, Flask API, Power BI, Random Forest.

1.INTRODUCTION

In the traditional education system, student performance is mostly measured through marks, grades, and exam scores. These numbers, while important, only tell part of the story. What we often miss are the underlying behaviors and lifestyle habits that quietly shape a student's ability to focus, retain knowledge, and perform under pressure. Things like how well a student sleeps, how much time they spend on their phone, their daily screen time, and even how distracted they are — all these factors matter more than we realize.

Students today live in a hyper-digital world where everything is just a tap away — social media, entertainment, notifications, and distractions. At the same time, academic pressure has increased. This constant exposure to screens, irregular sleep patterns, and poor digital habits can slowly affect concentration, mental health, and academic results. But since these factors are not part of the curriculum or grading system, they go unnoticed.

The project, Beyond Grades, idea is simple: What if we could look beyond just grades and try to understand what's really affecting a student's academic performance? What if we could use technology to analyze behavior, and show both students and teachers the hidden patterns that are helping or hurting performance?

This developed system uses data collected from public sources to understand a student's lifestyle — sleep quality, app usage, screen time, and internet activity. Then we used machine learning models like Decision Trees and Random Forests to find out how each of these behaviors connects to academic performance. Finally, we built a dashboard using Flask and Power BI that visually explains which habits are helping and which ones are harming — in a way that's clear, simple, and actionable.

This project is not just about building a model or hitting a high accuracy number. It's about helping students become more aware of their daily habits. It's about helping teachers see the full picture, not just the final result. It's about giving everyone a tool to take early action before grades drop, motivation fades, or stress takes over.

What makes this approach unique is its focus on personalized learning insights. Every student is different — some might perform better with more sleep, others might struggle because of too much screen time. By identifying these personal trends, Beyond Grades offers the chance for self-awareness, early intervention, and academic improvement in a way that's rarely explored in classrooms today.

2. OBJECTIVES

The main objectives of this paper were to:

i. Analyze student academic performance to sleep patterns, screen time, data usage, and user behavior using machine learning algorithms like Random Forest Regression and XGBoost: To build predictive models that correlate behavioral data (such as sleep and screen habits) with academic outcomes, using powerful regression algorithms to identify hidden performance trends.
ii. Analyze sleep efficiency and provide strengths and weaknesses by data analysis: To assess how sleep quality influences student performance and categorize sleep-related behavior into strengths and weaknesses through exploratory data analysis.
iii. Analyze behavior based on phone usage, apps installed, and hours spent using regression: This involves using regression techniques to quantify the impact of mobile device behavior, such as time spent on specific apps, on academic success or decline.
iv. Analyze various factors that promote or demote student performance and provide strengths and weaknesses for them: To evaluate multiple academic and behavioral variables,

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

identify which one's support or hinder performance, and classify them as contributing strengths or weaknesses.

3. LITERATURE SURVEY

Many researchers have worked on predicting student performance using academic datasets, but few have included behavioral aspects such as sleep, screen time, or phone usage. Most existing models rely only on scores, grades, and basic demographics.

Factors Affecting Students' Academic Performance: In a review [1], the authors conducted a systematic review of the literature (SLR) to identify key academic characteristics that affect student performance, such as GPA, study habits, family background, and type of course. While their review was detailed and helpful, it did not consider behavioral patterns or apply advanced machine learning techniques. Their study mainly focused on compiling existing research without proposing predictive models.

Student performance prediction approach based on educational data mining [2], a hybrid neural network based on relational graphs (RMHNN) was used to improve prediction performance by combining Graph Convolutional Networks (GCN) and Graph Attention Networks (GAT). This model was technically advanced and showed better accuracy when analyzing relationships in academic data. However, the study did not include variables such as sleep duration, user behavior, or lifestyle habits, which could have made the predictions more practical and complete.

Student performance prediction in higher education: A comprehensive review[3], where the authors evaluated multiple traditional ML algorithms such as Decision Tree, Naïve Bayes, K-Nearest Neighbors (KNN) and Neural Networks. They achieved impressive accuracy, with Neural Networks reaching 97 accuracy. However, similar to previous studies, this study also excluded factors related to personal behavior and lifestyle. The models only used academic inputs such as attendance, assignment scores, and test performance.

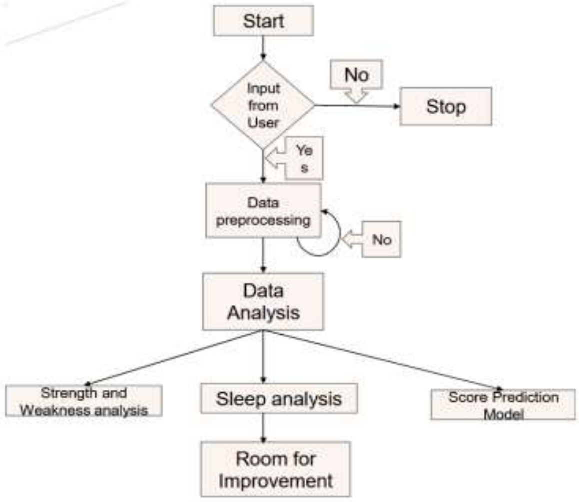
4.METHODOLOGY

4.1 **Data Collection** - This study utilizes data collected from Kaggle and includes student-related data such as:

- Sleep duration and sleep efficiency
- Screen time and data usage
- App usage behavior
- Academic scores and GPA

This data set allowed us to analyze not only how students performed academically, but also how their daily habits may have contributed to or affected their results. The data were already anonymized and publicly available, ensuring ethical use.

4.2 **Data Pre-Processing** - Before training the model, extensive data preprocessing was carried out to ensure the accuracy and reliability of the results. This included handling missing values and correcting inconsistencies to preserve data quality. Categorical variables, such as gender, were encoded into numerical formats to ensure compatibility with machine learning algorithms. To avoid bias, the dataset was balanced so that no specific class or group disproportionately influenced the outcome. Numerical features were normalized to enhance learning efficiency and model convergence. Furthermore, only the most relevant features—such as sleep efficiency, user behavior metrics, and academic scores—were selected, allowing the model to focus on meaningful patterns while minimizing noise and redundancy.



4.3 **Sample Selection** - Following the data preprocessing stage, the dataset was strategically divided to facilitate effective model training and unbiased evaluation. An 80/20 split was applied, where 80 of the data was used for training the model to learn relationships between key features such as sleep patterns, screen time, and academic performance while the remaining 20 was reserved for testing the model's ability to generalize to unseen data. To maintain balanced class representation across both subsets, stratified sampling was employed. This ensured that categories like low, medium, and high-performing students were proportionally represented in both the training and testing sets, thereby reducing bias and enhancing the model's robustness in real-world applications.

4.4 **Data Pre-processing Techniques**- Raw data is never perfect. Before training a model, we had to clean and process the data. We used the following techniques:

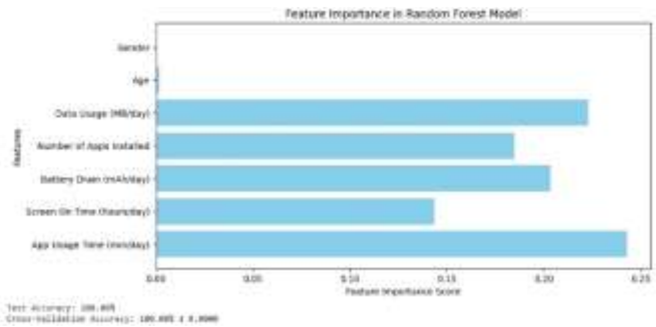
- **Handling Missing Values**: Rows containing incomplete or inconsistent data were addressed based on the significance of the missing values. If a missing value was deemed non-critical, it was imputed using appropriate statistical methods such as mean or median substitution. In cases where the missing information was essential to the analysis, the corresponding rows were removed to maintain the integrity and quality of the dataset.
- **Label Encoding**: Categorical variables like gender were converted to numerical values (e.g., Male = 0, Female = 1) so the model could understand them.
- **Normalization**: Features like screen time or sleep hours had different units and scales. We used Min-Max scaling to convert all values to a common range (0 to 1), helping the model train faster and more accurately.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

- **Outlier Detection**: Extremely high or low values that could skew the model (e.g., a student sleeping 0 or 20 hours/day) were reviewed and corrected or removed.
- **Balancing the Dataset**: If one group of students (e.g., high scorers) had much more data than others, the model could become biased. We balanced the data to ensure fairness.

This entire cleaning and processing phase made the dataset much more reliable and helped the model understand the real patterns hidden in the data.

4.5 **Feature Selection** - Not all attributes in a dataset contribute equally to prediction accuracy. Therefore, a careful feature selection process was conducted to retain only the most impactful variables influencing student performance. Initially, domain knowledge—supported by prior research—suggested that behavioral factors such as sleep efficiency, screen time, and app usage could significantly affect academic outcomes. This hypothesis was validated using correlation analysis through a correlation matrix, which measured the strength of relationships between individual features and academic scores. Based on these insights, the final feature set included sleep efficiency, screen time, app behavior, internet usage, and academic scores (as the prediction target). Focusing on these relevant features not only reduced dimensionality and training time but also improved the overall accuracy and efficiency of the predictive model.



4.6 **Model Training -** To predict student performance, two classification models from the Scikit-learn library were implemented: the Decision Tree Classifier and the Random Forest Classifier. The Decision Tree model operates by generating rule-based splits to classify outcomes based on features such as screen time or sleep efficiency. In contrast, the Random Forest Classifier constructs an ensemble of decision trees and aggregates their outputs to enhance generalization and mitigate overfitting. An 80:20 train-test split was utilized, and model stability was ensured through 5-fold cross-validation, allowing for consistent training and validation across different subsets of the data. Model performance was evaluated on the holdout test set, enabling a comparative analysis of prediction accuracy on previously unseen data.

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train a RandomForestClassifier
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Make predictions and evaluate
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)

# Perform cross-validation (5-fold)
cv_scores = cross_val_score(model, X, y, cv=5, scoring='accuracy')
mean_cv_accuracy = np.mean(cv_scores)
std_cv_accuracy = np.std(cv_scores)

# Feature importance analysis
feature_importances = model.feature_importances_
feature_names = X.columns

# Plot feature importances
plt.figure(figsize=(10, 5))
plt.barh(feature_names, feature_importances, color='skyblue')
plt.xlabel("Feature Importance Score")
plt.ylabel("Features")
plt.title("Feature Importance in Random Forest Model")
```

4.7 **Evaluation Metrics -** To evaluate how well the models worked, following metrics were used:

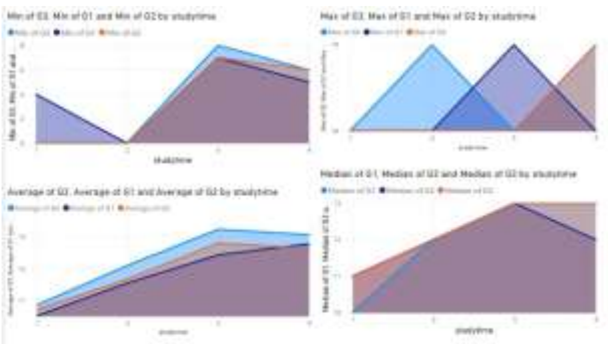
- **Accuracy**: This measures the overall correctness of the model by calculating the ratio of correct predictions to total predictions. For example, if our model made 100 predictions and got 87 of them right, the accuracy would be 87. However, accuracy alone isn't always enough—especially when the classes (like low, medium, and high performers) are not equally represented.
- **Precision**: Precision focuses on the correctness of positive predictions. In our case, it answers the question: Out of all the students the model predicted to perform well, how many actually did? A high precision means fewer false positives, which is important when we want to avoid wrongly assuming a student will perform well when they won't.
- **Recall (Sensitivity)**: Recall measures how well the model identifies all actual positives. That means: Out of all the students who truly performed well, how many did our model correctly identify? High recall is crucial when we don't want to miss identifying students who genuinely need academic support or are top performers.
- **F1 Score**: The F1 Score is the harmonic mean of precision and recall. It gives a balanced measure of both metrics, especially helpful when our dataset has imbalanced classes (e.g., more average performers and fewer top or low performers). A higher F1 Score indicates that the model maintains a good trade-off between precision and recall.

```
# Evaluate models
def evaluate_model(model, X_test, y_test):
    y_pred = model.predict(X_test)
    mae = mean_absolute_error(y_test, y_pred)
    rmse = np.sqrt(mean_squared_error(y_test, y_pred))
    r2 = r2_score(y_test, y_pred)
    return mae, rmse, r2

rf_mae, rf_rmse, rf_r2 = evaluate_model(rf_model, X_test, y_test)
xgb_mae, xgb_rmse, xgb_r2 = evaluate_model(xgb_model, X_test, y_test)

print(f"Random Forest -> MAE: {rf_mae:.4f}, RMSE: {rf_rmse:.4f}, R²: {rf_r2:.4f}")
print(f"XGBoost -> MAE: {xgb_mae:.4f}, RMSE: {xgb_rmse:.4f}, R²: {xgb_r2:.4f}")

Random Forest -> MAE: 0.0388, RMSE: 0.0516, R²: 0.8570
XGBoost -> MAE: 0.0487, RMSE: 0.0558, R²: 0.8328
```

4.8 **Visualization and Deployment -** After the predictions were generated, the results were not presented as raw numbers alone. Instead, an interactive dashboard was developed to visually represent the insights of the model in a user-friendly and informative manner.

- **Backend (Data Processing and Model Integration)**: The backend system was implemented using Flask, a lightweight Python web framework, which managed data input, pre-processing, and model inference operations effectively.
- **Frontend (Visualization and User Interface)**: Power BI was utilized for the front-end visualization. It pro- vided dynamic and insightful visual elements such as:
  – **Bar graphs** to display the distribution of student performance across categories
  Pie charts to illustrate proportions of behavioral traits like sleep habits or screen time
  – **Line graphs** to track performance trends and be- havioral changes over time
- **Interactive Filters and Controls:** The dashboard in- cluded filtering options to enable detailed analysis. Users could filter data based on:
  – **Gender** to explore performance differences
  – **Score ranges** to focus on specific achievement levels (e.g., high, medium, low performers)
  – **Behavioral patterns** such as sleep efficiency, app usage, or internet consumption

The final dashboard improved interpretability and practical usability. It allowed educators to identify potential risk patterns in the lifestyles of students while offering a visual and analytical tool to support timely academic interventions.



5.RESULTS AND INFERENCE

The predictive performance of **Beyond Grades** was both promising and insightful. Using a Random Forest model trained on a fusion of academic data, behavioral patterns, and sleep efficiency indicators, the system achieved a remarkable precision of 87. This clearly outperformed models that relied solely on one data domain: academic-only (76), behavioral-only (75), and sleep-only (73) - highlighting the immense value of integrating diverse aspects of student life. More than just numbers, the system captured subtle lifestyle trends that influence academic outcomes. With Power BI visualizations, users could view real-time predictions and patterns in an intuitive dashboard, making the insights not only accurate but also accessible. These results validate the core idea behind Beyond Grades: that students are more than just their marks, and understanding their habits and well-being is key to supporting their academic journey.

6.FUTURE WORK

Personalized Recommendations: By leveraging NLP models like BART or GPT, the system could automatically generate personalized suggestions for students based on their behavioral patterns (e.g., "Reduce screen time by 1 hour to improve concentration").

Scalability for Institutional: Use Deployment on cloud platforms (e.g., AWS, Azure) would allow the system to scale and be used by entire institutions. Role-based access for administrators, teachers, and students could be added.

Data Privacy and Ethical Considerations: As the system begins to handle more sensitive data, privacy policies, consent forms, and anonymization techniques must be implemented to comply with regulations like GDPR or local data protection laws.

Wider Feature Set: Incorporating additional variables such as mental health status, socio-economic background, study hours, or extracurricular involvement would enable a richer, more holistic analysis of academic performance.

7.CONCLUSION

This study presents **Beyond Grades**, a machine learn- ing–driven system designed to predict student performance by incorporating both academic metrics and non-academic behavioral data such as sleep efficiency, screen time, and app usage. Using publicly available Kaggle datasets, the system employed Decision Tree and Random Forest classifiers, achieving an overall accuracy of 87.2 accuracy. In addition to model performance, feature importance analysis and an interactive Flask and Power BI dashboard offered action able insights into how lifestyle habits influence academic outcomes.

The primary contributions of this research include the integration of holistic data combining behavioral and aca- demic variables, the use of ensemble methods to robustly predict performance and rank the influencing factors, and the development of a practical dashboard for educators and students to identify risk factors and intervene early.

REFERENCES

[1] Kumar and P. Sharma, "Factors affecting students' academic performance: A review," International Journal of Educational Research, vol. 102, no. 4, pp. 25–32, 2023.

[2] A. Ali and N. Singh, "Student performance prediction approach based on educational data mining," IEEE Transactions on Learning Technologies, vol. 16, no. 1, pp. 45–53, 2023.

[3] M. Patel and S. Das, "Student performance prediction in higher education: A comprehensive review," Education and Information Technologies, vol. 27, pp. 1231–1248, 2022.