



International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)

Machine Learning-Based Student Dropout Prediction and Rule-Based Counselling Support System: A Comparative Study

Ishan Singh

School of Information Technology, Indira University, Pune, India

Abstract— Student dropout is a persistent concern in higher education because learners may face academic, financial and personal difficulties that are not detected early. This paper describes the design and evaluation of a machine-learning-based student dropout prediction prototype integrated with a rule-based counselling support interface. The publicly available UCI Predict Students' Dropout and Academic Success dataset contains 4,424 records and 36 input features. After excluding 794 students with an unresolved Enrolled outcome and removing second-semester attributes, the binary classification dataset contained 3,630 records and 30 input features. Logistic Regression, Decision Tree and Random Forest classifiers were compared using a stratified 80:20 train-test split. Logistic Regression achieved 89.53% accuracy and 86.52% dropout-class F1-score, ahead of Random Forest (88.71%, 85.30%) and Decision Tree (83.61%, 79.73%). The selected model was deployed in a Streamlit application that estimates dropout probability from a short student form and generates transparent, rule-based academic and financial support suggestions. This is a proof-of-concept using historical benchmark data, not a validated institutional decision-making system. Limitations include selection on the test set, incomplete form inputs and the lack of prospective evaluation.

Keywords—educational data mining, dropout prediction, logistic regression, random forest, decision tree, early warning, counselling support, Streamlit.

I. INTRODUCTION

Higher-education institutions may benefit from identifying students who are likely to discontinue their studies before an adverse outcome occurs. Student withdrawal can be associated with a combination of academic progress, tuition obligations, prior qualifications and demographic context. A risk estimate can help direct attention to learners who might benefit from timely human-led support, although a statistical association must not be interpreted as a causal explanation for an individual student's decision.

This project addresses two practical needs: estimating dropout risk from available student records and translating selected warning indicators into understandable suggestions for academic or financial assistance.

Rather than presenting a classifier alone, the prototype integrates a prediction form, historical-data dashboard, performance results and counselling guidance in a local web interface. The research contribution is an implementation-level comparison of three common supervised learning methods and a transparent demonstration of their integration into a student-support workflow.

The research questions are: (RQ1) Which of the three evaluated algorithms provides the highest dropout-class F1-score under the chosen split? (RQ2) Can the selected classifier be integrated into a usable offline application? (RQ3) How can predefined support rules accompany a probability estimate without representing themselves as professional counselling?

II. RELATED WORK

V. Realinho and colleagues introduced a higher-education student-outcome dataset with enrolment, socioeconomic and academic variables for predicting dropout and academic success [1], [2]. The accompanying case study examined early prediction of student performance using educational records [3]. Such work illustrates the usefulness of structured administrative data for early-warning research while highlighting that the timing and availability of variables matter for realistic deployment.

The present work evaluates Logistic Regression as an interpretable linear baseline, Decision Tree as a set of learned branching rules and Random Forest as a tree ensemble. Decision-tree induction is a long-established classification approach [4], while Random Forest combines multiple randomized trees to improve stability [5]. All models are implemented using scikit-learn [6]. Unlike a purely retrospective comparison, the project also demonstrates an end-user interface; however, its counselling suggestions are explicitly authored rules rather than empirically tested interventions.

III. DATASET AND PROBLEM FORMULATION

The data were obtained from the UCI Machine Learning Repository, dataset 697, "Predict Students' Dropout and Academic Success" [1].

It consists of 4,424 student observations, 36 original predictors and one target with three outcomes. Records describe students across multiple undergraduate programs and include enrolment characteristics, prior educational history, socioeconomic factors and first- and second-semester performance.

Original target	Number of records	Share of dataset
Graduate	2,209	49.93%
Dropout	1,421	32.12%
Enrolled	794	17.95%
Total	4,424	100%

Because the application predicts a binary eventual outcome, Enrolled records were excluded: their final graduation or dropout status was not established. The retained cohort comprised 2,209 graduates (label 0) and 1,421 dropouts (label 1), totaling 3,630 observations. All features containing “2nd sem” in their names were removed to avoid requiring second-semester outcomes in an earlier-stage prediction scenario, leaving 30 predictors. First-semester indicators remain, so the resulting system should be described as an early academic-progress risk estimator rather than an enrolment-day predictor.

IV. PROPOSED METHODOLOGY

A. Data preparation and splitting

Input features were separated from the target label. The cohort was partitioned using `train_test_split` with `test_size = 0.20`, `random_state = 42` and stratification by the target, resulting in 2,904 training observations and 726 test observations. Stratification preserves approximately the same class balance in both subsets. A scikit-learn Pipeline applies median imputation and standardization; fitting these transformations on the training partition avoids directly learning preprocessing statistics from the test partition.

Some original attributes represent coded categories, but the prototype treats the retained numeric codes within the same general preprocessing pipeline. This is a simplifying design choice: one-hot encoding or category-aware processing could improve statistical appropriateness for certain features and should be evaluated in future experiments.

B. Model training

Three supervised classifiers were trained using the same feature representation: Logistic Regression, Decision Tree and Random Forest (100 estimators). Logistic Regression models the log-odds of dropout as a linear function of the transformed features and returns class probabilities.

Decision Tree recursively partitions feature space into regions, while Random Forest aggregates many trees fitted with randomized variation. Each classifier was evaluated on the same held-out observations.

C. Evaluation criteria

Accuracy is the fraction of all correctly classified observations. For the positive dropout class, precision is $TP/(TP+FP)$, recall is $TP/(TP+FN)$, and $F1 = 2 \times \text{precision} \times \text{recall} / (\text{precision} + \text{recall})$, where TP, FP and FN denote true positives, false positives and false negatives. Dropout-class F1 was emphasized because a model should both identify students who drop out and avoid an excessive number of false alarms. No single metric is sufficient to establish real-world suitability; calibration, subgroup behavior and operational costs also matter.

V. EXPERIMENTAL RESULTS

Model	Acc	Prec.	Recall	F1
Log. Reg.	0.8953	0.8714	0.8592	0.8652
Decision Tree	0.8361	0.7723	0.8239	0.7973
Random Forest	0.881	0.8686	0.8380	0.8530

Logistic Regression yielded the highest observed accuracy (89.53%) and dropout-class F1-score (86.52%). Random Forest was close, with 88.71% accuracy and 85.30% F1, while Decision Tree obtained 83.61% accuracy and 79.73% F1. The result demonstrates that a comparatively simple linear model can be competitive on this dataset and split; it does not establish that Logistic Regression is universally superior for student dropout prediction.

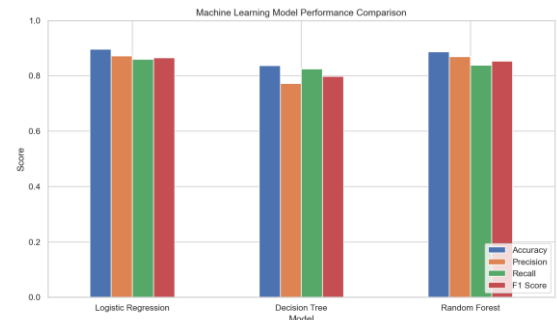


Fig. 1. Model performance comparison (accuracy, precision, recall and F1-score).

The best-performing model was selected according to F1 on the same held-out test partition used to report final metrics.



Consequently, the reported best-model performance may be optimistically biased by the model-selection process. A stronger protocol would use cross-validation or a separate validation set for model selection and an untouched final test set. The results are single-split observations; no confidence intervals, repeated runs or external validation are reported.

VI. SYSTEM ARCHITECTURE AND IMPLEMENTATION

The prototype consists of an offline model-development component and a Streamlit presentation component. A Jupyter Notebook acquires the public dataset, prepares the binary classification cohort, fits the candidate pipelines, computes comparison metrics and serializes the selected model using joblib. Supporting files store the feature-column order, default input values, evaluation table and figures. The Streamlit application loads these local artifacts without retraining the model for each prediction.

The web interface provides four navigation sections: Dashboard, Dropout Prediction, Counselling and Model Performance. The Dashboard summarizes the original three-class dataset and shows a small table preview. The prediction page accepts age at enrolment, scholarship status, tuition-payment status, outstanding debt, first-semester units enrolled, units passed and average first-semester grade. A validation rule prevents the number of passed subjects from exceeding the number enrolled. The input is assembled in the feature order expected by the trained pipeline.

Only seven features are directly entered by the user, while the model expects 30. The remaining features are populated from stored training-data defaults. The system therefore illustrates a feasible interaction pattern but cannot be considered a fully individualized estimate. The selected classifier returns a predicted class and the estimated probability for dropout. The interface displays low ($<40\%$), moderate ($40\% - <70\%$) and high ($\geq 70\%$) probability bands. These bands are illustrative interface choices, not empirically calibrated intervention thresholds.

For offline operation, the historical dataset is stored as student_data.csv, and the dashboard reads it from the local project directory. Model artifacts and evaluation figures are also stored locally. Once Python dependencies are installed, the application can run through a localhost Streamlit server without requiring an internet connection.

VII. RULE-BASED COUNSELLING MODULE

The support module uses deterministic conditions on the user-entered fields and estimated probability. When fewer than 60% of enrolled first-semester units have been passed, the system recommends tutoring and academic guidance.

An average grade below 10/20 triggers study-strategy counselling. Unpaid tuition or outstanding debt triggers financial-assistance suggestions, and absence of a scholarship triggers scholarship guidance. An estimated dropout probability of at least 50% adds a recommendation for individual counselling. If no conditions apply, the interface recommends routine progress monitoring.

Observed condition	Suggested support
Passed/enrolled units < 0.60	Tutoring and academic guidance
First-semester grade $< 10/20$	Academic counselling
Tuition unpaid or outstanding debt	Financial assistance
No scholarship	Scholarship guidance
Estimated dropout probability ≥ 0.50	Individual counselling

These suggestions are not generated by a language model and have not been tested for efficacy. They should be interpreted as prompts for a qualified adviser, not autonomous decisions. In particular, a predicted probability must not be used to deny enrolment, financial aid or other opportunities. The application is a decision-support demonstration rather than a substitute for human judgement.

VIII. DISCUSSION, ETHICS AND LIMITATIONS

The observed performance suggests that student administrative and first-semester indicators can contain useful signal for distinguishing historical dropout and graduation outcomes. Integrating the classifier with a lightweight interface also makes the output easier to inspect than a notebook-only experiment. Nonetheless, several constraints limit generalization. First, the dataset originates from a particular higher-education context and may not reflect different institutions, cohorts or geographic settings. Second, the removal of Enrolled observations changes the prediction population and excludes students with unresolved outcomes. Third, categorical codes were not handled with a dedicated encoding strategy.

Fourth, only a small subset of features is supplied interactively; the rest use defaults. Fifth, the risk bands and counselling thresholds were chosen for demonstration rather than derived from stakeholder requirements or intervention studies. Sixth, there is no prospective deployment, fairness audit, probability calibration study, usability assessment or comparison with an institution-specific baseline. Finally, choosing the winning algorithm on the same test set used for reporting introduces model-selection bias.



Ethical use requires protecting student data, limiting access to predictions, checking performance across relevant groups and ensuring that interventions are supportive rather than punitive. This prototype relies on a public research dataset and does not report collecting identifiable student records. Any future institutional deployment would require local governance, privacy review, consent or other lawful basis as applicable, and evaluation of whether the recommendations improve outcomes.

IX. CONCLUSION AND FUTURE WORK

This study implemented and compared three machine-learning classifiers for binary student dropout prediction and connected the selected model to a rule-based counselling support interface. On a single stratified hold-out split of the UCI dataset, Logistic Regression achieved the best reported dropout-class F1-score (0.8652) and accuracy (0.8953). The Streamlit application demonstrates local dataset exploration, individual risk estimation, performance visualization and transparent support suggestions, including offline execution after installation.

Future work should separate model selection from final evaluation, test multiple cross-validation folds, evaluate category-aware feature processing and calibrated probabilities, collect all required prediction inputs, assess subgroup performance, and validate risk thresholds with education professionals. Prospective studies would also be needed to determine whether counselling recommendations lead to improved student outcomes.

X. DATA AVAILABILITY AND DECLARATIONS

Dataset: UCI Machine Learning Repository, dataset 697, DOI: 10.24432/C5MC89 [1]. The work is an academic software prototype; no claim is made that the model or counselling suggestions have been validated in a live institution. Conflicts of interest: none declared by the author. Funding: no external funding reported for this project.

REFERENCES

- [1] V. Realinho, M. Vieira Martins, J. Machado, and L. Baptista, "Predict Students' Dropout and Academic Success," UCI Machine Learning Repository, 2021, doi: 10.24432/C5MC89.
- [2] V. Realinho, J. Machado, L. Baptista, and M. V. Martins, "Predicting Student Dropout and Academic Success," *Data*, vol. 7, no. 11, art. 146, 2022, doi: 10.3390/data7110146.
- [3] M. V. Martins, D. Tolledo, J. Machado, L. M. T. Baptista, and V. Realinho, "Early prediction of student's performance in higher education: a case study," in *Trends and Applications in Information Systems and Technologies*, 2021, doi: 10.1007/978-3-030-72657-7_16.
- [4] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, pp. 81–106, 1986, doi: 10.1007/BF00116251.
- [5] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
- [6] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [7] Streamlit, "Streamlit documentation," <https://docs.streamlit.io/> (accessed Oct. 20, 2026).