Localized Actionable Explainability for MOOC Dropout Intervention Using Clickstream Learning Analytics

Mohamed Amine OUASSIL[1], Mohammed JEBBEARI[2], Mouaad ERRAMI[3],
Rabia Rachidi[4], Soufiane HAMIDA[5], Bouchaib CHERRADI[6], Abdelhadi RAIHANI[7]
EEIS Laboratory, ENSET of Mohammedia, Hassan II University of Casablanca, Mohammedia 28830, Morocco[1, 2, 3, 6, 7]
LaROSERI Laboratory-Faculty of Science, Chouaib Doukkali University, El Jadida, Morocco[4]
2IACS Laboratory- ENSET of Mohammedia, Hassan II University of Casablanca, Mohammedia, Morocco[5]
STIE Team-CRMEF Casablanca-Settat, provincial section of El Jadida, El Jadida 24000, Morocco[2, 6]
GENIUS Laboratory, SupMTI of Rabat, Rabat, Morocco[5]

***Abstract*—Massive Open Online Courses (MOOCs) have expanded access to learning, but learner dropout remains a persistent challenge in online education. Most dropout prediction studies transform learner traces into behavioral features and train machine learning or deep learning models to identify at-risk learners. Although this predictive layer is useful, it remains limited for educational decision-making because a risk score does not explain why a learner is at risk or what type of support should be proposed. This study introduces Localized Actionable Explainability (L-AXAI), a learning analytics framework that combines dropout prediction, local explanation, and personalized pedagogical nudging. The framework first transforms clickstream logs into weekly behavioral indicators, then trains predictive classifiers, and finally applies SHAP-based local explanation to identify the strongest risk driver for each high-risk learner. These local explanations are mapped to targeted pedagogical actions using a transparent rule-based translation matrix. Experiments on the KDD Cup MOOC dropout benchmark produced closely comparable results for the two strongest boosting models. Under the fixed stratified test split, Gradient Boosting achieved 85.79% accuracy, 86.57% ROC-AUC, and 94.41% PR-AUC, while XGBoost achieved 85.77% accuracy and the same ROC-AUC. Because repeated-seed evaluation and statistical significance testing were not conducted, the small difference between these point estimates should not be interpreted as definitive model superiority. The explainability analysis identifies model-associated patterns involving weekly activity intensity, active days, and early problem engagement. The intervention layer generates candidate nudge categories, mainly navigation, re-engagement, and session-completion suggestions. The framework therefore returns an estimated dropout risk, a local risk driver, and a candidate pedagogical action. The nudges are intended as low-intensity decision-support suggestions; their appropriateness, perceived pressure, and effectiveness require validation with instructors and learners.**

Keywords*—MOOC dropout prediction; learning analytics; explainable artificial intelligence; SHAP; localized explanation; actionable intervention

I. Introduction

Online learning platforms have become one of the main sources of data for education. Every learner activity such as opening a course page, viewing a video, trying a problem, or posting in a discussion forum leaves a trace that can be analyzed to better understand the engagement and learning progression [1], [2], [3]. These traces are used by learning analytics and educational data mining to support teachers, designers, and institutions in tracking learning processes and improving educational decisions [4], [5], [6], [7], [8]. MOOCs are a prime example of data-rich learning environments, with large-scale enrollment and continuous clickstream records. But the very openness that makes MOOCs appealing also makes it hard to persist. Learners enter MOOCs with different goals, different time constraints, and different levels of commitment. Thus, completion rates cannot fully account for disengagement, as learners may follow several trajectories rather than a single common path [9], [10]. Consequently, predicting dropout has become a central task in learning analytics. Previous work has shown that activity indicators, weekly engagement signals, and temporal representations can be used to identify learners who are likely to stop participating [11], [12], [13], [14]. Later work also advised that predictive performance should be carefully evaluated because training and testing on the same course may overestimate the real accuracy of a dropout model [15], [16]. Predictive models are useful, but prediction is not sufficient for teaching. A model may identify a learner as high-risk, but this information is weak if the instructor does not know which behavior created the risk. A learner who doesn't try problems needs a different type of support than a learner who watches videos but never uses discussions. Likewise, a learner with a long inactivity gap needs a different nudge than a learner who browses through many pages without completing learning tasks [17], [18], [19], [20]. Explainable Artificial Intelligence (XAI) helps to open this black-box problem by showing which features affect a prediction. LIME and SHAP are popular methods for explaining machine learning models, but counterfactual explanations offer an action-oriented perspective by asking what changes are needed to obtain a different outcome [21], [22], [23], [24], [25], [26]. However, many educational applications of XAI are confined to global feature importance. A global plot can show that problem events are important in general, but it cannot automatically tell what to do for a specific learner [27], [28], [29], [30], [31]. Thus, the

real challenge is not only to predict dropout, but also to associate the prediction with localized and educationally meaningful action. This study introduces Localized Actionable Explainability (L-AXAI), a framework that identifies at-risk learners, explains each individual prediction, and converts the top local risk driver into a pedagogical nudge. The framework is evaluated on a public MOOC clickstream benchmark for dropout prediction.

The main contributions of this work are fivefold. First, it proposes a localized actionable explainability framework for MOOC dropout intervention. Second, it extracts directly interpretable weekly behavioral features from clickstream events, including access, problem-solving, video engagement, discussion participation, navigation, wiki usage, and inactivity. Third, it formalizes the predictive and explainability layers through supervised classification, tree-based boosting, SHAP values, and an intervention mapping function. Fourth, it introduces a transparent rule-based translation matrix that converts local SHAP risk drivers into pedagogical nudges. Finally, it evaluates the framework not only in terms of predictive performance, but also through global and local explanation analysis and the distribution of generated intervention types.

The rest of this study is organized as follows: Section II reviews the related literature. The mathematical formulation and the proposed L-AXAI framework architecture are described in Section III. Results are given in Section IV. Discussion is presented in Section V. Finally, Section VI concludes this study.

II. Literature Review

Over the past decade, MOOC dropout prediction has progressed from simple activity-count models to temporal, representation-based, and explainable learning analytics frameworks [11], [12], [13], [14], [32]. Existing studies mainly focus on improving early detection, generalization across courses, or predictive performance [15], [16], [33], [34]. More recent work has started to consider interpretability, but many studies still stop at feature importance, SHAP plots, or visual explanations without converting the explanation into a direct learner-level pedagogical action [17], [19], [20], [27], [35]. This section reviews activity-based, temporal, deep representation, explainable, and action-oriented dropout prediction studies, highlighting key contributions and limitations.

Early dropout prediction studies were mainly based on learner activity features extracted from platform logs. Halawa et al. [11] used activity indicators to red-flag students who were likely to leave a course, showing that learner interaction traces can support early warning. Similarly, Kloft et al. [12] predicted dropout over weeks using machine learning models and showed that weekly behavior can provide useful signals for early detection. Taylor et al. [13] developed a scalable dropout prediction pipeline and demonstrated that temporal and non-temporal features can produce strong AUC values. However, these approaches rely heavily on engineered counts and feature importance analysis, which limits their ability to explain the specific barrier faced by each learner.

Several studies then focused on validation, transferability, and robustness. Boyer and Veeramachaneni [36] explored transfer learning for predictive models across MOOC contexts, while Whitehill et al. [15] showed that same-course evaluation can give overly optimistic estimates of dropout prediction accuracy. Gardner and Brooks [16] also argued that dropout model evaluation needs more rigorous procedures because feature extraction methods and algorithms interact strongly. These works are important because they show that high accuracy alone is not enough. Nevertheless, their main concern is predictive validity rather than pedagogical actionability.

Temporal and representation learning methods were proposed to capture changes in learner behavior more deeply. Fei and Yeung [14] introduced temporal models for dropout prediction, moving beyond static activity totals. Jeon et al. [37] proposed interpretable multi-layer representation learning to reduce manual feature engineering, and Jeon and Park [38] modeled clicks and videos jointly to improve weekly dropout prediction. These studies improve the modeling of sequential engagement and learning resources. However, their outputs still mainly support classification, and the connection between learned representations and instructor-level intervention remains limited.

Other studies investigated behavior patterns and learner trajectories. Kizilcec et al. [9] showed that MOOC disengagement is not one single behavior but a set of learner subpopulations with different engagement trajectories. Feng et al. [18] analyzed dropout behaviors and identified meaningful categories of learners in large-scale MOOCs. Alamri et al. [19] used visually driven, multi-granularity explanations to compare learning paths of completers and non-completers, showing that non-completers may jump forward or follow irregular paths. These studies are closer to pedagogy because they describe behavior, but they do not automatically generate a concrete nudge for each high-risk learner.

Explainability has been introduced more directly in dropout prediction pipelines. Nagrecha et al. [17] showed that MOOC dropout prediction pipelines can be made interpretable with performance close to stronger black-box methods. Er [20] used explainable machine learning to understand dropout in two MOOC runs and discussed feature importance from a pedagogical point of view. Swamy et al. [27] compared several explanation methods in student performance prediction, including LIME, SHAP, DiCE, and contrastive explanations, and showed that explanation methods can disagree. These studies show the relevance of XAI in education, but explanation is still often presented as a separate analysis after prediction rather than as an intervention engine.

Recent work also moved toward SHAP-based and intervention-oriented systems. Liu [39] used a dilated convolutional behavior model and SHAP analysis to identify important learning behavior variables. Nti and Ramanayake [35] proposed an explainable dropout prediction framework that also considers tailored interventions in online personalized education. Ajayi [40] combined stacked ensemble learning with explainability for online dropout risk, and Nagy et al. [41] connected interpretable dropout prediction with personalized intervention. These studies are important because they

explicitly connect interpretability with student support. However, most of them do not define a transparent event-to-nudge translation matrix based on local clickstream deficits.

Counterfactual and group-level explanation methods provide another direction for actionable learning analytics. Wachter et al. [24] formalized counterfactual explanations as a way to explain automated decisions without opening the black box, while Mothilal et al. [25] proposed diverse counterfactual explanations that are useful when several possible changes may lead to a desired outcome. Buñay-Guisñan et al. [42] applied group counterfactual explanations to support students at risk of dropout, aiming to recover larger groups with lower intervention cost. These works are highly relevant for future learning analytics because they bring actionability into explanation. Still, direct mapping from local MOOC event deficits to practical pedagogical nudges remains underdeveloped.

The literature shows a clear evolution in the field of MOOC dropout prediction, from activity-count models to more complex temporal, representation-based, and more explainable models. However, many studies still focus only on prediction accuracy or visual explanation. One remaining gap is how to translate local learner-level explanations into transparent and practical pedagogical nudges that are personalized and can be directly used by instructors.

III. Materials and Methods

A. Proposed Method

The study is based on quantitative learning analytics, utilizing supervised machine learning and post-hoc explainability. The goal is not only to predict at-risk learners, but also to explain why each at-risk prediction is produced and to transform this explanation into a concrete pedagogical action. The proposed L-AXAI workflow presented in Fig. 1 begins from raw MOOC clickstream logs, which record learner interaction events such as login, page views, video access, problem attempts, discussion activity, navigation, and session closing. During preprocessing, irrelevant events are removed, timestamps are converted into a common format, and actions are aggregated into learner-course sessions. Missing or invalid records are also removed. The dataset is split into training, validation, and testing sets after pre-processing. The behavioral feature engineering stage converts the raw logs into weekly learner features. Such features are time indicators, weekly event counts, and deficit indicators. Event counts indicate what the learner did, temporal indicators show the pace of participation, and deficit indicators show what is missing when comparing the learner's behavior to the reference behavior of successful learners in the same course. In the prediction stage, the following machine-learning models are trained and compared: Logistic Regression, SVM, KNN, Decision Tree, Random Forest, Extra Trees, AdaBoost, Gradient Boosting, XGBoost, and LightGBM. Metrics for classification, like accuracy, ROC-AUC, and PR-AUC, are used to evaluate candidate models. The selected model is then used to produce an estimated dropout-risk score for each learner. The framework utilizes local SHAP explanation for learners having a predicted dropout probability above the risk threshold. This step identifies the features that exert the greatest influence on each individual prediction. Rather than a global model interpretation, L-AXAI is concerned with the local risk driver of each learner. Finally, the rule-based translation matrix maps the selected risk driver to a pedagogical action, such as a practice-oriented nudge, re-engagement message, video review recommendation, or help-seeking prompt. This design enables transparent and responsible learning analytics by connecting prediction, explanation, and intervention into one workflow. Fig. 1 illustrates the proposed L-AXAI workflow for MOOC dropout prediction and intervention.

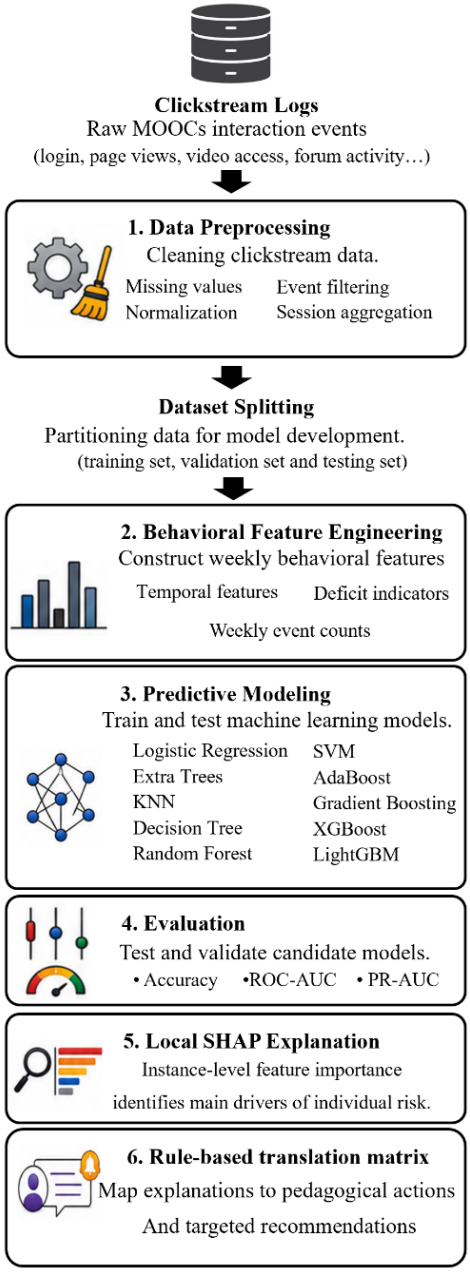

Fig. 1. Proposed L-AXAI workflow for MOOC dropout prediction and intervention.

B. Data Source and Event Representation

The experimental data are from the KDD Cup 2015 MOOC dropout prediction benchmark. The dataset contains learner-

course enrollment records and timestamped activity logs from a large MOOC platform. Each activity record contains an enrollment ID, a timestamp, a source object, and an event type. The task is formulated as a binary classification problem to predict if a learner-course enrollment will become inactive in the target period. The framework describes learner behavior through seven types of clickstream events: access, discussion, navigate, page_close, problem, video, and wiki. Access events indicate general connection to the course. Video events indicate consumption of instructional content, problem events indicate practice and assessment engagement, discussion events indicate social and help-seeking interactions, and wiki events indicate consultation of supplementary materials. Navigate and page_close events provide complementary information about movement between course objects and ending a session. Table I presents the interpretation of MOOC clickstream event types.

TABLE I. INTERPRETATION OF MOOC CLICKSTREAM EVENT TYPES

Event type	Behavioral meaning	Pedagogical interpretation
access	Opening or accessing course resources	General connection with the course
video	Interaction with course videos	Exposure to explanations and demonstrations
problem	Attempts or interaction with problems	Practice, self-testing, and assessment engagement
discussion	Forum or discussion activity	Help-seeking, peer interaction, and social learning
wiki	Consultation of wiki resources	Use of additional knowledge materials
navigate	Movement between course objects	Learning path and course exploration
page_close	Closing/leaving a course page	Session ending and possible early exit behavior

C. Preprocessing and Feature Engineering

The preprocessing step begins with a merge of the enrollment table and the log table on the enrollment identifier. Remove records that do not have key identifiers. Then each timestamp of learner action is transformed into a common datetime format and sorted chronologically. In this study, weekly observation windows are used instead of representing learner behavior with one global activity vector. Each week consists of seven days of activity. Thus, the actions of the learner are represented separately for week 1, week 2, week 3, and week 4. This design allows the model to learn the time variation of dropout risk. From the raw logs, three groups of features are derived.

The first type is weekly event counts including access, video, problem, discussion, wiki, navigate, and page_close events. For example, problem_w2_count is the number of problem-related actions taken by the learner in Week 2. This variable is coded as Week 2 problem activity in the analysis.

The second group includes temporal features on a weekly basis. Such features characterize the rhythm of the learner's participation. For example, active_days_w2 is the number of active days in Week 2, and maximum_daily_events_w2 is the maximum number of actions recorded in one day in Week 2. This variable is interpreted as Week 2's highest daily activity.

The third group is made up of weekly deficit indicators. A deficit is the difference between the learner's activity and the median activity of non-dropout learners in the same course. problem_w2_deficit shows the distance from the course reference for the learner's problem activity in Week 2. If the learner is not below the reference, then the deficit is set to zero. All course reference medians used for deficit computation were estimated only from the training set. For validation and test records, these training references were reused without accessing validation or test labels.

Event counts are what the learner did, temporal features are when and how often the learner participated, and deficit indicators are what is missing compared to successful learners. These deficit measures support the intervention layer by allowing a local risk driver to be translated into a candidate pedagogical nudge.

D. Machine Learning and Explainability Methods

The prediction task is formulated as a supervised binary classification problem. Let the processed dataset be represented as:

$$\mathcal{D} = \{(\mathrm{x}_i, y_i)\}_{i=1}^{N}, \quad \mathrm{x}_i \in \mathbb{R}^d, \quad y_i \in \{0,1\} \tag{1}$$

where, x_i is the feature vector of the learner-course record i, d is the number of engineered features, and y_i is the dropout label. The classifier learns a function that maps the behavioral representation of a learner to a dropout probability:

$$f: \mathbb{R}^d \to [0,1], \quad \hat{p}_i = f(\mathrm{x}_i) \tag{2}$$

where, $\hat{p}_i$ is the predicted probability that learner i will drop out. A binary decision is obtained by applying a risk threshold. τ:

$$\hat{y}_i = \begin{cases} 1, & \text{if } \hat{p}_i \geq \tau \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

In this study, the risk threshold τ was set to 0.5, meaning that learner-course records with a predicted dropout probability greater than or equal to 0.5 were classified as high-risk.

Logistic Regression is used as a simple baseline because it gives a transparent relationship between features and predicted probability. [15], [43]. It estimates dropout probability using the sigmoid function:

$$\hat{p}_i = \sigma(\beta_0 + \mathrm{x}_i^{\mathrm{T}}\beta), \quad \sigma(z) = \frac{1}{1+e^{-z}} \tag{4}$$

The Support Vector Machine (SVM) is included as a margin-based classifier. It separates the two classes by finding a decision boundary with a maximum margin. The K-Nearest Neighbors (KNN) classifier is used as a non-parametric baseline. It predicts the label of a learner according to the labels of the most similar learners in the training set:

$$\hat{y}_i = \mathrm{mode}\{y_l : \mathrm{x}_l \in \mathcal{N}_k(\mathrm{x}_i)\} \tag{5}$$

where $\mathcal{N}_k(\mathrm{x}_i)$ denotes the set of the k nearest training samples to x_i. Decision Tree is used as a single-tree baseline, while Random Forest and Extra Trees are used as randomized tree ensembles. These models combine several decision trees

and aggregate their predictions, commonly through majority voting [44]:

$$\hat{y}_i = \text{mode}\{h_t(\text{x}_i)\}_{t=1}^{T} \quad (6)$$

where, h_t is the prediction of tree t, and T is the total number of trees in the ensemble. Random Forest builds trees from bootstrap samples, while Extra Trees increases randomness by using more randomized split selection. Both strategies help reduce variance compared with a single decision tree. The remaining classifiers belong to the boosting family: AdaBoost, Gradient Boosting, XGBoost, and LightGBM. Boosting models build the final predictor additively by adding new weak learners that correct the errors of previous learners [33], [34]:

$$F_m(\text{x}_i) = F_{m-1}(\text{x}_i) + \eta h_m(\text{x}_i) \quad (7)$$

where, h_m is the weak learner added at iteration m, and η is the learning rate. XGBoost and LightGBM are included because they are efficient for structured tabular features and include regularization mechanisms to control overfitting. A general regularized boosting objective can be written as:

$$\mathcal{L}^{(m)} = \sum_{i=1}^{N} l\left(y_i, F_{m-1}(\text{x}_i) + \eta h_m(\text{x}_i)\right) + \Omega(h_m) \quad (8)$$

where, l is the predictive loss function and $\Omega(h_m)$ penalizes model complexity. After model training, SHAP is used to explain model outputs. SHAP is based on Shapley values and belongs to additive feature attribution methods [21], [23], [45]. For a learner i, the prediction is decomposed as:

$$g(\text{x}_i) = \phi_0 + \sum_{j=1}^{d} \phi_{i,j} \quad (9)$$

where, ϕ_0 is the expected model output and $\phi_{i,j}$ is the contribution of feature j to the prediction of learner i. The Shapley value of feature j is defined as:

$$\phi_{i,j} = \sum_{S \subseteq F_j} w(S)[f(S \cup j) - f(S)] \quad (10)$$

where, F_j denotes the set of all features excluding feature j. The weighting term is defined as:

$$w(S) = \frac{|S|!\,(d - |S| - 1)!}{d!} \quad (11)$$

In this expression, S is a subset of features that excludes feature j, and the term in brackets measures the marginal change caused by adding feature j. SHAP is suitable for this study because it provides both global and local explanations. Global explanations show which variables matter across the cohort, while local explanations show why a specific learner is classified as high-risk. In L-AXAI, only learners whose predicted dropout probability exceeds the threshold are passed to the action-generation step. For each high-risk learner, the framework selects the strongest positive actionable SHAP contribution:

$$r_i = \underset{j \in A}{\text{argmax}} \phi_{i,j}, \quad \phi_{i,j} > 0 \quad (12)$$

where, A is the set of actionable weekly features and r_i is the local risk driver that most strongly pushes the learner i toward the dropout class. The selected driver is then mapped to a pedagogical action using the rule-based translation matrix T:

$$a_i = T(r_i) \quad (13)$$

The final output of the proposed framework is therefore not only a predicted label. For each high-risk learner, L-AXAI returns a triplet composed of the estimated dropout-risk score, the local explanation, and a candidate pedagogical nudge template:

$$o_i = (\hat{p}_i, r_i, a_i) \quad (14)$$

This formulation shows the difference between standard predictive modeling and L-AXAI. A standard classifier stops at the predicted probability or class label. In contrast, L-AXAI links prediction to explanation and then to intervention, which supports localized and actionable learning analytics. The estimated score is used as an operational decision value under the threshold τ=0.5. Because probability calibration was not evaluated, it should not be interpreted as a validated empirical probability of dropout.

E. L-AXAI Rule-Based Translation Matrix

The mappings in Table II were constructed as an author-designed, literature-informed, and behavior-semantics-informed prototype rather than as the outcome of a formal expert-consensus process. Each rule links the behavioral meaning of the corresponding event or deficit to a support objective: practice for low problem activity, content review for low video activity, reconnection for low access, help-seeking for low discussion participation, regularity support for inactivity, diversified resource use for low event diversity, progression support for navigation deficits, and session-completion support for frequent page closing with few active days. The matrix is deliberately transparent and editable and should therefore be interpreted as a set of candidate nudge templates rather than as validated pedagogical prescriptions. Instructor review and learner-centered evaluation are required before operational deployment.

TABLE II. MAPPING LOCAL SHAP RISK DRIVERS TO PROPOSED PEDAGOGICAL NUDGE TEMPLATES

Top local risk driver	Interpretation	Candidate pedagogical nudge
Low problem_count	The learner is not practicing enough	Try to complete one or two practice problems from this week. This will help you check your understanding before moving forward.
Low video_count	The learner may not be following explanations	Return to the main video of this unit and watch the key explanation before attempting the next activity.
Low access_count	The learner is disconnected from the course	You have not accessed the course regularly. Try to log in today and review the current unit for a few minutes.
Low discussion_count	The learner is not using peer or instructor support	Visit the discussion forum and read one question related to this unit. You can also post a short question if something is unclear.
High inactivity_ga	The learner has stopped regular	You had several days without activity. Restart with one small task

Top local risk driver	Interpretation	Candidate pedagogical nudge
p	participation	today rather than waiting until the end of the week.
Low event_diversity	The learner uses only one type of resource	Try to combine watching the video with solving one problem or reading one discussion thread.
Low navigate_count / high navigate_deficit	The learner is not progressing through course objects	Open the next course object and complete one small activity before moving to other pages
High page_close_count with low active_days	The learner may leave sessions quickly	Try to spend a little more time in one session and finish one complete activity before closing the page.

Algorithm 1 summarizes the main steps of the proposed Weekly L-AXAI framework. It starts by preparing clickstream data and converting learner actions into weekly behavioral features. Then, the predictive model estimates dropout probability for each learner and selects only learners above the risk threshold. For these high-risk learners, local SHAP values are computed to identify the main actionable risk driver, which is finally translated into a pedagogical nudge for the instructor dashboard.

Algorithm 1: Weekly L-AXAI for MOOC Dropout Intervention

Input: Enrollment table E, log table L, labels Y, number of weekly windows K, risk threshold τ, predictive model f, and matrix T.

Merge E and L using the enrollment identifier.

Convert timestamps to datetime format and order learner actions chronologically

For each learner-course enrollment i **do**

  For each weekly window k = 1 to K **do**

    Select learner actions inside the weekly window W_k.

    Count access, video, problem, discussion, wiki, navigate, and page_close events.

    Compute active days, average daily events, maximum daily events, inactivity days, and event diversity

    Store weekly features

  End

End

Split the processed dataset into training, validation, and test subsets.

Compute course-level reference medians using only non-dropout learners from the training set

For each learner i **do**

  Compute weekly deficit indicators by comparing x_i with the course reference

End

Train the predictive model f on the training set.

For each learner i in the test set **do**

  Estimate dropout probability $\hat{p}_i = f(x_i)$.

  If $(\hat{p}_i \geq \tau)$ **then**

    Compute local SHAP values φ_{ij} for learner i.

    Identify the strongest positive actionable risk driver r_i.

    Generate the pedagogical action $a_i = T(r_i)$.

    Add $(i, \hat{p}_i, r_i, a_i)$ to the intervention table.

  End

End

Return the intervention table.

IV. RESULTS

The processed dataset was divided into 70% training, 15% validation, and 15% testing using stratified random sampling with random seed 42. Stratification preserved the dropout/non-dropout distribution across the three subsets. All data-dependent operations, including course-reference medians and correlation-based feature filtering, were estimated from the training subset and then applied unchanged to validation and test data. The validation subset was used for hyperparameter selection, and the test subset was reserved for final evaluation. Experiments were performed in Google Colaboratory on an NVIDIA T4 GPU with 54 GB RAM. Gradient Boosting used 250 estimators, a learning rate of 0.05, and a maximum tree depth of 3. XGBoost used 500 estimators, a learning rate of 0.05, a maximum depth of 5, a subsampling ratio of 0.85, and a column-sampling ratio of 0.85. The operational risk threshold was fixed at $\tau = 0.5$. Parameters not explicitly modified retained their corresponding library defaults. The experiments were implemented in Python using scikit-learn, XGBoost, LightGBM, and SHAP. Parameters not explicitly modified retained their corresponding library defaults.

TABLE III. COMPARATIVE RESULTS OF DROPOUT PREDICTION MODELS

Model	Number of Features	Accuracy (%)	ROC-AUC (%)	PR-AUC (%)
Gradient Boosting	72	85.79	86.57	94.41
XGBoost	72	85.77	86.57	94.45
LightGBM	72	81.28	86.34	94.28
Logistic Regression	72	80.89	86.27	94.35
SVM	72	85.1	86.18	94.33
AdaBoost	72	84.82	86.14	94.28
Extra Trees	72	83.68	86.11	94.07
Random Forest	72	85.26	85.21	93.33
Decision Tree	72	80.78	84.33	92.92
KNN	72	84.97	84.09	93.03

Table III presents the performance of the tested machine learning models using the final weekly feature set. Gradient Boosting achieved the highest accuracy point estimate, while

XGBoost produced closely comparable performance, with 85.79% accuracy, 86.57% ROC-AUC, and 94.41% PR-AUC. XGBoost obtains almost the same performance, with 86.57% ROC-AUC. The difference between the two boosting models is small, but both are above simple baselines such as Decision Tree and KNN. Fig. 2 shows the accuracy comparison for the proposed models in this study.

Fig. 2 presents the accuracy comparison of the tested classifiers using the weekly feature representation. Under the original fixed split, Gradient Boosting achieved the highest accuracy point estimate at 85.79%, while XGBoost achieved 85.77% and the same ROC-AUC of 86.57%. The 0.02 percentage-point difference is too small to support a claim of definitive superiority without repeated experiments and statistical testing. The two boosting models are therefore interpreted as closely comparable under the present evaluation protocol. Gradient Boosting is retained as the predictive backbone used for the subsequent L-AXAI explanation stage, while the main contribution of the study remains the connection between estimated risk, local explanation, and candidate pedagogical action rather than a claim of universal classifier superiority.

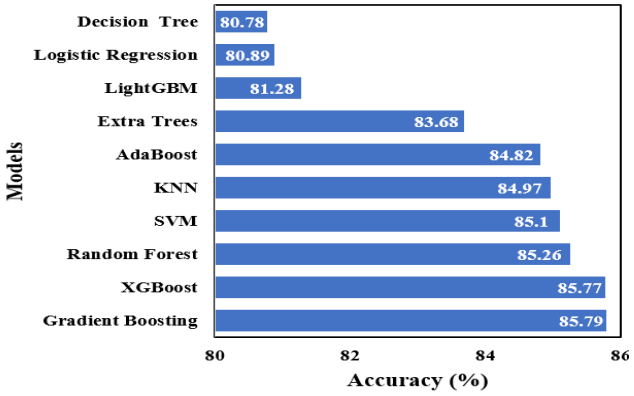

Fig. 2. Accuracy comparison of the tested classifiers.

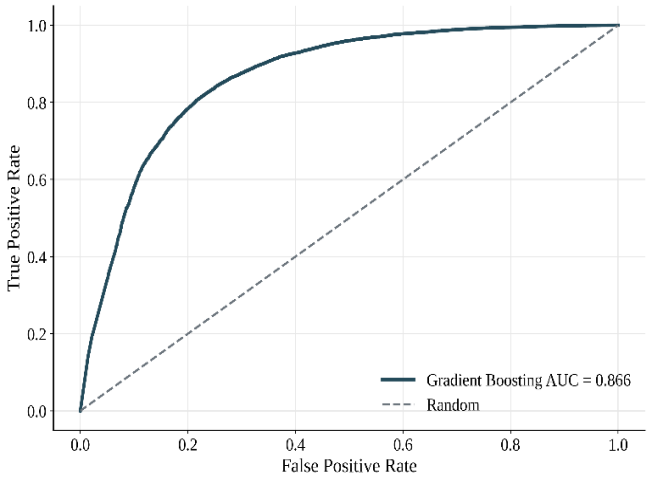

Fig. 3. ROC curve of the selected weekly model.

As shown in Fig. 3, the ROC curve confirms good discrimination between dropout and non-dropout learners. However, because the dataset is imbalanced, the precision-recall curve is more informative for the practical use of the model. The PR-AUC is high for the best models, showing that the system maintains useful precision even when the positive dropout class dominates the data.

A. Weekly Observation Horizon Analysis

The weekly horizon comparison evaluates whether useful dropout signals appear early or only after several weeks of activity. The results in Table IV show a clear improvement as the observation window grows from Week 1 to Week 4. Week 1 alone already provides useful predictive information, but the full four-week representation obtains the highest accuracy and ROC-AUC. These results show that extending the observation horizon from Week 1 to Week 4 provides additional predictive information under the current evaluation protocol.

TABLE IV. BEST MODEL FOR EACH WEEKLY OBSERVATION HORIZON

Horizon	Model	Number of Features	Accuracy (%)	ROC-AUC (%)	PR-AUC (%)
Week 1	XGBoost	20	81.28	80.33	91.95
Week 1-2	Gradient Boosting	39	83.46	84.22	93.56
Week 1-3	Gradient Boosting	58	85.26	85.92	94.17
Week 1-4	Gradient Boosting	72	85.79	86.57	94.41

B. Feature Filtering

Table V shows the highly correlated variables removed before final modelling. The weekly representation increases the number of variables because event counts, temporal indicators, and deficit variables are repeated over weeks. To reduce feature bloat, highly correlated variables were removed using a correlation threshold of 0.95. As expected, inactivity_days variables were perfectly correlated with active days variables and were removed from the final model. Several Week 4 deficit variables were also removed because they were almost redundant with access_w4_deficit.

TABLE V. HIGHLY CORRELATED VARIABLES REMOVED BEFORE FINAL MODELING

Dropped feature	Correlated with	Absolute correlation	Threshold
inactivity_days_w1	active_days_w1	1	0.95
inactivity_days_w2	active_days_w2	1	0.95
event_diversity_w2_deficit	navigate_w2_deficit	0.9548	0.95
inactivity_days_w3	active_days_w3	1	0.95
event_diversity_w3_deficit	access_w3_deficit	0.9569	0.95
inactivity_days_w4	active_days_w4	1	0.95
video_w4_deficit	access_w4_deficit	0.9751	0.95
problem_w4_deficit	access_w4_deficit	0.986	0.95
navigate_w4_deficit	access_w4_deficit	0.9844	0.95
page_close_w4_deficit	access_w4_deficit	0.9834	0.95
active_days_w4_deficit	access_w4_deficit	0.9821	0.95
event_diversity_w4_deficit	access_w4_deficit	0.9938	0.95

C. Explainability Analysis

The explainability analysis is structured on two levels. First, global SHAP analysis identifies the weekly variables that have the greatest influence on the model across the test cohort. Second, local SHAP analysis provides explanation for individual learner cases with waterfall plots and interaction plots. This setup is similar to the logic in explainability-oriented papers where global patterns and local case studies are explained at the same time.

1) Global SHAP importance: The global SHAP ranking in Table VI and Fig. 4 show that temporal intensity variables dominate the prediction. The most influential variable is "average daily events in week 3", followed by "average daily events in week 2", "active days in week 1", "average daily events in week 1", and "problem in week 1 count". This means that the model does not rely only on whether a learner logs in, but also on the intensity and regularity of weekly engagement.

TABLE VI. TOP 10 GLOBAL SHAP WEEKLY RISK DRIVERS

Feature	Mean absolute SHAP	Mean SHAP	Feature mean	Feature std
average_daily_events_w3	0.2681	-0.0274	1.265	3.794
average_daily_events_w2	0.2199	-0.041	1.455	4.121
active_days_w1	0.1953	-0.0243	1.637	1.099
average_daily_events_w1	0.182	-0.0464	5.05	6.596
problem_w1_count	0.1289	-0.0063	3.797	10.716
average_daily_events_w4	0.1287	0.0279	0.814	3.601
active_days_w4	0.0559	0.0034	0.22	0.765
access_w1_count	0.0527	-0.0029	13.308	18.743
active_days_w3	0.0404	0.0025	0.358	0.875
navigate_w2_deficit	0.0392	0.0119	0.743	0.437

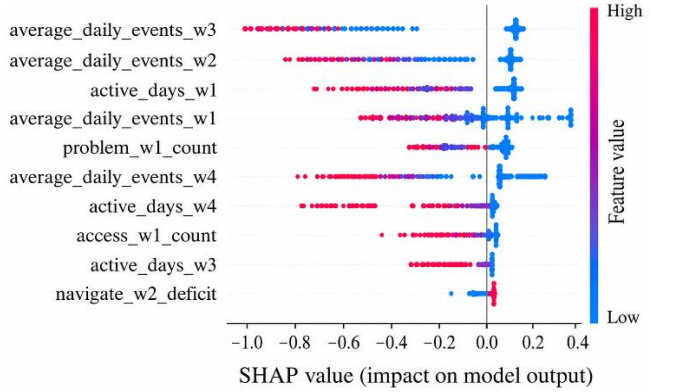

Fig. 4. Global SHAP summary plot for the selected weekly model.

2) Temporal Global SHAP evolution: In Table VII, the temporal SHAP evolution analysis shows how the type of predictive signal changes across weeks. In Week 1, temporal variables contribute 57.12% of the weekly SHAP mass, event counts contribute 35.23%, and deficit variables contribute 7.66%. In Weeks 3 and 4, temporal features become even more dominant, reaching 81.29% and 77.42%, respectively. Fig. 5 illustrates temporal SHAP evolution by weekly feature group.

TABLE VII. TEMPORAL SHAP CONTRIBUTION BY WEEKLY FEATURE GROUP

Feature type	Week	Mean absolute SHAP	Relative contribution (%)
Deficit	Week 1	0.0558	7.66
	Week 2	0.0866	19.34
	Week 3	0.0063	1.48
	Week 4	0.0008	0.29
Event count	Week 1	0.2565	35.23
	Week 2	0.0853	19.05
	Week 3	0.0731	17.24
	Week 4	0.064	22.29
Temporal	Week 1	0.4158	57.12
	Week 2	0.2758	61.6
	Week 3	0.3449	81.29
	Week 4	0.2224	77.42

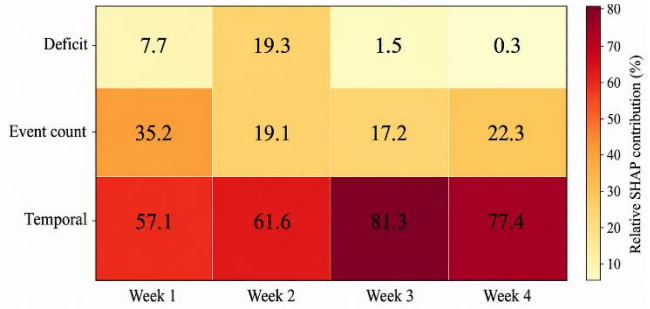

Fig. 5. Temporal SHAP evolution by weekly feature group.

3) Local SHAP waterfall analysis: Tables VIII and IX show four local cases explained with SHAP waterfall analysis. Cases A and B were both predicted as high-risk with a dropout probability of 95.98%. Cases A and B are behavioral twins selected to illustrate that identical clickstream patterns may lead to different final outcomes. Their main risk drivers are low activity in the first weeks, especially "average daily events in Week 1", "average daily events in Week 2", "average daily events in Week 3", and "problem Week 1 count". This means that early low engagement strongly pushed the model toward dropout.

Because Cases A and B have the same engineered clickstream vector, the deterministic classifier assigns them the same predicted probability and the same SHAP decomposition. Their different observed outcomes do not indicate an inconsistency in the model calculation. Instead, they show that the current feature representation is incomplete. Learner goals, prior knowledge, motivation, external time constraints, offline study, and interactions not recorded by the platform may influence persistence even when the observed clickstream behavior is identical. The pair is therefore treated as an

illustration of residual uncertainty and unobserved heterogeneity.

Cases C and D were predicted as low-risk. Their negative SHAP values show that higher weekly activity reduced the predicted dropout probability, especially high "average daily events in week 3", "average daily events in week 4", and several active days in Weeks 2 and 4. Case C is a false negative, which suggests that some dropout decisions may be caused by factors not visible in clickstream logs.

TABLE VIII. LOCAL CASE PROFILES SELECTED FOR SHAP WATERFALL ANALYSIS

Case	Case description	Drop out	Predicted dropout probability	Prediction
A	High-risk dropout	1	95.98	1
B	High-risk non-dropout	0	95.98	1
C	Low-risk dropout	1	2.46	0
D	Low-risk non-dropout	0	1.52	0

TABLE IX. TOP FIVE LOCAL SHAP CONTRIBUTIONS FOR SELECTED LEARNER CASES

Case	Rank	Feature	Feature value	SHAP value
A	1	average_daily_events_w1	0.143	0.3998
	2	average_daily_events_w3	0	0.1517
	3	active_days_w1	1	0.1378
	4	average_daily_events_w2	0	0.1245
	5	problem_w1_count	0	0.1089
B	1	average_daily_events_w1	0.143	0.3998
	2	average_daily_events_w3	0	0.1517
	3	active_days_w1	1	0.1378
	4	average_daily_events_w2	0	0.1245
	5	problem_w1_count	0	0.1089
C	1	average_daily_events_w3	16.143	-0.8571
	2	active_days_w4	5	-0.7091
	3	average_daily_events_w4	16.429	-0.5911
	4	average_daily_events_w2	16.857	-0.4854
	5	active_days_w2	6	-0.4313
D	1	average_daily_events_w3	27.714	-0.9057
	2	active_days_w4	5	-0.702
	3	average_daily_events_w4	21.429	-0.5737
	4	average_daily_events_w2	7	-0.4725
	5	active_days_w2	6	-0.438

Fig. 6-(a) and Fig. 6-(b) show that the high-risk cases are mainly explained by low engagement indicators, especially low average daily events, limited problem activity, and weak access behavior. These features push the prediction score upward and increase the predicted dropout risk. In contrast, Fig. 6-(c) and Fig. 6-(d) show that stronger weekly engagement reduces the prediction score and is associated with low-risk learner profiles. Overall, the SHAP waterfall plots demonstrate how the model explains dropout risk at the individual learner level by identifying the main behavioral factors that increase or decrease the prediction.

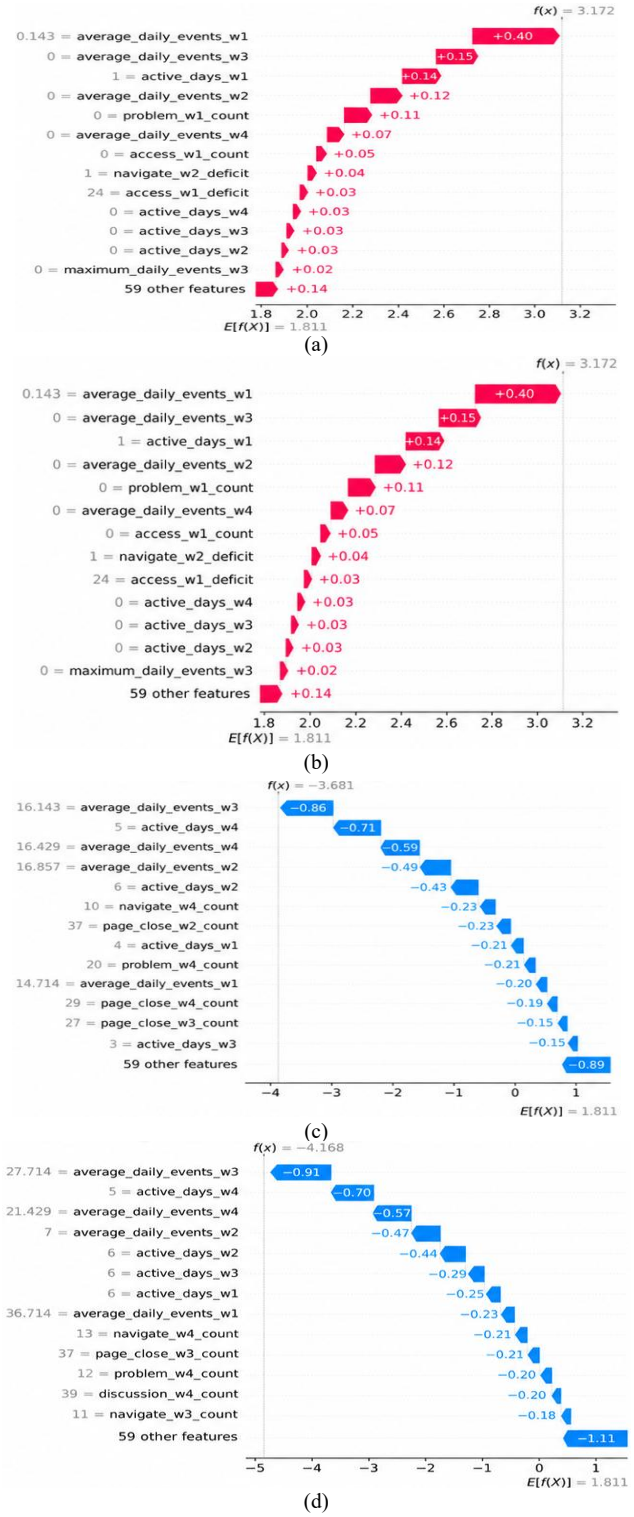

Fig. 6. SHAP waterfall plot for (a) High-risk dropout case, (b) High-risk non-dropout case, (c) Low-risk dropout case, and (d) Low-risk non-dropout case.

4) Weekly L-AXAI intervention output: As shown in Table X and XI, the weekly L-AXAI module generated 13,237 high-risk intervention records on the test set. Approximately 11,610 corresponded to actual dropout records and 1,627 to non-dropout records. The generated messages are candidate decision-support suggestions and were not evaluated with learners or instructors. Navigation nudges represented 74.62% of all candidate nudges and 89.76% of the Week 2 nudges. This concentration reflects both the behavior represented in the test set and the current one-to-one mapping rule, which selects only the strongest positive actionable SHAP driver. It may also be reinforced by correlation filtering: event_diversity_w2_deficit was removed because it was highly correlated with navigate_w2_deficit ($|r| = 0.9548$), leaving the retained navigation feature to represent part of this related Week 2 signal. The distribution should therefore not be interpreted as a universal pedagogical pattern. For future deployment, correlated drivers should be grouped into broader support families, the top-k actionable drivers may be considered, course-specific thresholds should be used, repeated delivery of the same message should be limited, and instructor approval should be retained. These safeguards are proposed design recommendations and were not empirically tested in the present study.

TABLE X. OVERALL DISTRIBUTION OF WEEKLY L-AXAI NUDGE TYPES

Nudge Type	Count	Percentage
Navigation nudge	9,877	74.62
Re-engagement nudge	2,300	17.38
Session completion nudge	962	7.27
Regularity nudge	57	0.43
Content review nudge	33	0.25
Balanced engagement nudge	8	0.06

TABLE XI. WEEKLY DISTRIBUTION OF L-AXAI NUDGE TYPES

Week	Nudge Type	Count	Percentage within week
1	Balanced engagement nudge	8	0.39
	Content review nudge	33	1.6
	Re-engagement nudge	1,142	55.28
	Session completion nudge	883	42.74
2	Navigation nudge	9,812	89.76
	Re-engagement nudge	1,116	10.21
	Regularity nudge	1	0.01
	Session completion nudge	2	0.02
3	Navigation nudge	65	32.83
	Regularity nudge	56	28.28
	Session completion nudge	77	38.89
4	Re-engagement nudge	42	100

V. DISCUSSION

The results describe the predictive performance of the complete weekly feature representation under a stratified train-validation-test evaluation setting. Stratification preserves the dropout and non-dropout distribution in the three subsets. This is important since the dataset is imbalanced. Thus, the reported performance measures the ability of the model to distinguish dropout from non-dropout students with similar class distributions during the training, validation, and testing phases.

The evaluation should be interpreted as a stratified within-benchmark and within-distribution comparison. Training-only estimation of course-reference medians and feature filters prevents direct use of validation or test labels, but the random split does not provide chronological, learner-independent, or course-independent separation. Learner-course records with similar behavioral patterns may therefore occur across subsets, and the reported metrics may be optimistic relative to deployment on completely unseen courses or future cohorts. The findings do not establish cross-platform generalizability. Transfer to another MOOC environment would require event-schema harmonization, verification of the dropout definition and observation horizon, re-estimation of reference medians, probability recalibration, and contextual review of the translation matrix.

The best predictive performance was obtained by tree-based boosting models. Gradient Boosting and XGBoost performed very similarly, and Logistic Regression and SVM stayed competitive. This suggests that weekly clickstream features contain strong predictive signals that can be exploited by linear and non-linear models.

The explainability analysis identifies model-associated behavioral patterns within the present dataset. The SHAP summary plot and local case analysis indicate how weekly activity intensity, active days, and early problem engagement contributed to the selected model's predictions. These associations should not be interpreted as causal determinants of dropout because clickstream logs omit learner goals, prior knowledge, offline study, motivation, and contextual constraints. Explanation stability across repeated training runs, permutation-based validation, and comparison with alternative explainability techniques were not evaluated. The reported SHAP patterns should therefore be interpreted as descriptive explanations of the selected model under the current split rather than as universally stable behavioral findings.

The temporal SHAP analysis describes how the selected model's attribution patterns vary across weekly feature groups. Temporal features are more important in later weeks, while the role of event-count and deficit features depends on the observation window. Therefore, the dropout risk should not be seen as a constant. The features that explain dropout risk in Week 1 are not the same features that explain dropout risk in Weeks 3 or 4. These findings provide a temporal description of the selected model's explanations, but they do not independently establish the predictive contribution of each feature family.

Interpretation at the learner level is emphasized in the local waterfall analyses. Two learners may have similar high-risk probabilities, but their explanations may reflect different combinations of weak activity, limited practice, or low weekly intensity. The high-risk non-dropout case also illustrates a limitation of prediction using clickstream data alone: some learners can behave like dropout learners but still persist. L-AXAI nudges should therefore be seen as supportive signals, not as punitive decisions.

Intervention analysis shows that local explanations can be translated into pedagogical actions at scale. But the strong dominance of navigation nudges should be treated with caution. This could be a real behavior pattern in the test set, or it could be that the translation matrix is giving a lot of priority to navigate_w2_deficit . The intervention matrix should be tuned in future work using feedback from instructors, learner-response data, and A/B testing to ensure that the nudges generated are varied, useful, and not repetitive.

VI. Conclusion

A weekly Localized Actionable Explainability framework for MOOC dropout prediction and intervention was proposed. Clickstream behavior is represented through weekly event counts, temporal indicators, and course-referenced deficit features, after which estimated dropout risk is linked to a local SHAP driver and a transparent candidate nudge template. Under the fixed stratified test split, Gradient Boosting achieved 85.79% accuracy, 86.57% ROC-AUC, and 94.41% PR-AUC, while XGBoost produced closely comparable results. Because repeated-seed experiments and statistical significance tests were not conducted, the small performance differences should be interpreted cautiously. The weekly-horizon analysis indicates that additional weeks provide progressively more predictive information, but a complete feature-family ablation was not performed. SHAP identified model-associated patterns involving activity intensity, active days, and early problem engagement; these findings are descriptive and were not validated through repeated-run stability or alternative explainability methods. The intervention layer generated candidate nudges dominated by navigation, re-engagement, and session-completion categories. This concentration requires diversified driver selection, repetition limits, and instructor-controlled delivery. Because no instructor panel, learner study, or controlled intervention trial was conducted, pedagogical effectiveness and perceived pressure remain unverified. The findings are limited to the KDD Cup 2015 benchmark and a stratified within-distribution protocol. Future work should examine repeated-seed, learner-grouped, course-held-out, and chronological evaluation; formal probability calibration; component-wise ablation; explanation-stability analysis; cross-platform event harmonization; expert co-design of the translation matrix; and real-world intervention studies.

References

[1] G. Siemens and R. S. J. d. Baker, “Learning analytics and educational data mining: towards communication and collaboration,” in Proceedings of the 2nd International Conference on Learning Analytics and Knowledge, 2012, pp. 252–254. doi: 10.1145/2330601.2330661.

[2] R. Ferguson, “Learning analytics: drivers, developments and challenges,” International Journal of Technology Enhanced Learning, vol. 4, no. 5/6, pp. 304–317, 2012, doi: 10.1504/IJTEL.2012.051816.

[3] D. Clow, “An overview of learning analytics,” Teaching in Higher Education, vol. 18, no. 6, pp. 683–695, 2013, doi: 10.1080/13562517.2013.827653.

[4] A. Pardo and G. Siemens, “Ethical and privacy principles for learning analytics,” British Journal of Educational Technology, vol. 45, no. 3, pp. 438–450, 2014, doi: 10.1111/bjet.12152.

[5] S. Slade and P. Prinsloo, “Learning analytics: Ethical issues and dilemmas,” American Behavioral Scientist, vol. 57, no. 10, pp. 1510–1529, 2013, doi: 10.1177/0002764213479366.

[6] H. Drachsler and W. Greller, “Privacy and analytics: it’s a DELICATE issue,” in Proceedings of the 6th International Conference on Learning Analytics and Knowledge, 2016, pp. 89–98. doi: 10.1145/2883851.2883893.

[7] C. Romero and S. Ventura, “Educational data mining and learning analytics: An updated survey,” Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 10, no. 3, p. e1355, 2020, doi: 10.1002/widm.1355.

[8] R. S. Baker and P. S. Inventado, “Educational data mining and learning analytics,” in Learning Analytics: From Research to Practice, Springer, 2014, pp. 61–75. doi: 10.1007/978-1-4614-3305-7_4.

[9] R. F. Kizilcec, C. Piech, and E. Schneider, “Deconstructing disengagement: analyzing learner subpopulations in massive open online courses,” in Proceedings of the 3rd International Conference on Learning Analytics and Knowledge, 2013, pp. 170–179. doi: 10.1145/2460296.2460330.

[10] D. F. O. Onah, J. Sinclair, and R. Boyatt, “Dropout rates of massive open online courses: behavioural patterns,” in EDULEARN14 Proceedings, 2014, pp. 5825–5834. [Online]. Available: https://wrap.warwick.ac.uk/id/eprint/65543/

[11] S. Halawa, D. Greene, and J. Mitchell, “Dropout prediction in MOOCs using learner activity features,” in Proceedings of the Second European MOOC Stakeholder Summit, 2014, pp. 58–65.

[12] M. Kloft, F. Stiehler, Z. Zheng, and N. Pinkwart, “Predicting MOOC dropout over weeks using machine learning methods,” in Proceedings of the EMNLP 2014 Workshop on Analysis of Large Scale Social Interaction in MOOCs, 2014, pp. 60–65. doi: 10.3115/v1/W14-4111.

[13] C. Taylor, K. Veeramachaneni, and U.-M. O’Reilly, “Likely to stop? Predicting stopout in massive open online courses,” 2014. [Online]. Available: https://arxiv.org/abs/1408.3382

[14] M. Fei and D.-Y. Yeung, “Temporal models for predicting student dropout in massive open online courses,” in Proceedings of the IEEE International Conference on Data Mining Workshop, 2015, pp. 256–263. doi: 10.1109/ICDMW.2015.174.

[15] J. Whitehill, K. Mohan, D. Seaton, Y. Rosen, and D. Tingley, “Delving deeper into MOOC student dropout prediction,” 2017. [Online]. Available: https://arxiv.org/abs/1702.06404

[16] J. Gardner and C. Brooks, “Dropout model evaluation in MOOCs,” 2018. [Online]. Available: https://arxiv.org/abs/1802.06009

[17] S. Nagrecha, J. Z. Dillon, and N. V. Chawla, “MOOC dropout prediction: lessons learned from making pipelines interpretable,” in Proceedings of the 26th International Conference on World Wide Web Companion, 2017, pp. 351–359. doi: 10.1145/3041021.3054162.

[18] W. Feng, J. Tang, and T. X. Liu, “Understanding dropouts in MOOCs,” in Proceedings of the AAAI Conference on Artificial Intelligence, 2019, pp. 517–524. doi: 10.1609/aaai.v33i01.3301517.

[19] A. Alamri, Z. Sun, A. I. Cristea, G. Senthilnathan, L. Shi, and C. Stewart, “Is MOOC learning different for dropouts? A visually-driven, multi-granularity explanatory ML approach,” 2020. [Online]. Available: https://arxiv.org/abs/2008.05209

[20] E. Er, “An explainable machine learning approach to predicting and understanding dropouts in MOOCs,” Kastamonu Education Journal, vol. 31, no. 1, pp. 143–154, 2023, doi: 10.24106/kefdergi.1246458.

[21] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in Advances in Neural Information Processing Systems, 2017. [Online]. Available: https://arxiv.org/abs/1705.07874

[22] M. T. Ribeiro, S. Singh, and C. Guestrin, “Why should I trust you? Explaining the predictions of any classifier,” in Proceedings of the 22nd

ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1135–1144. doi: 10.1145/2939672.2939778.

[23] L. S. Shapley, "A value for n-person games," in Contributions to the Theory of Games, Princeton University Press, 1953, pp. 307–317.

[24] S. Wachter, B. Mittelstadt, and C. Russell, "Counterfactual explanations without opening the black box: automated decisions and the GDPR," Harvard Journal of Law & Technology, vol. 31, no. 2, pp. 841–887, 2017, [Online]. Available: https://jolt.law.harvard.edu/assets/articlePDFs/v31/Counterfactual-Explanations-without-Opening-the-Black-Box-Sandra-Wachter-et-al.pdf

[25] R. K. Mothilal, A. Sharma, and C. Tan, "Explaining machine learning classifiers through diverse counterfactual explanations," in Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, 2020, pp. 607–617. doi: 10.1145/3351095.3372850.

[26] S. Verma, J. P. Dickerson, and K. Hines, "Counterfactual explanations for machine learning: A review," 2020. [Online]. Available: https://arxiv.org/abs/2010.10596

[27] V. Swamy, R. Kämpgen, S. Bataille, and T. Käser, "Can explainable AI methods help understand student performance prediction models?," in Proceedings of the 15th International Conference on Educational Data Mining, 2022, pp. 206–217. [Online]. Available: https://educationaldatamining.org/edm2022/proceedings/2022.EDM-long-papers.9/

[28] C. Molnar, Interpretable Machine Learning. Self-published, 2022. [Online]. Available: https://christophm.github.io/interpretable-ml-book/

[29] Z. C. Lipton, "The mythos of model interpretability," Queue, vol. 16, no. 3, pp. 31–57, 2018, doi: 10.1145/3236386.3241340.

[30] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," Nature Machine Intelligence, vol. 1, pp. 206–215, 2019, doi: 10.1038/s42256-019-0048-x.

[31] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," 2017. [Online]. Available: https://arxiv.org/abs/1702.08608

[32] S. Alghamdi, "A comprehensive review of dropout prediction methods in Massive Open Online Courses," Multimodal Technologies and Interaction, vol. 9, no. 1, p. 3, 2025, doi: 10.3390/mti9010003.

[33] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[34] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," The Annals of Statistics, vol. 29, no. 5, pp. 1189–1232, 2001, doi: 10.1214/aos/1013203451.

[35] I. K. Nti and S. Ramanayake, "Explainable machine learning for student dropout prediction and tailored interventions in online personalized education," Discover Artificial Intelligence, 2026, doi: 10.1007/s44163-026-01016-6.

[36] S. Boyer and K. Veeramachaneni, "Transfer learning for predictive models in massive open online courses," in Artificial Intelligence in Education, Springer, 2015, pp. 54–63. doi: 10.1007/978-3-319-19773-9_6.

[37] B. Jeon, N. Park, and S. Bang, "Dropout prediction over weeks in MOOCs via interpretable multi-layer representation learning," 2020. [Online]. Available: https://arxiv.org/abs/2002.01598

[38] B. Jeon and N. Park, "Dropout prediction over weeks in MOOCs by learning representations of clicks and videos," 2020. [Online]. Available: https://arxiv.org/abs/2002.01955

[39] Y. Liu, "MOOC dropout prediction via a dilated convolutional learning behavior model," Informatics, vol. 12, no. 4, p. 127, 2025, doi: 10.3390/informatics12040127.

[40] A. Ajayi, "Predicting student dropout risk in online learning using stacked ensemble learning and explainability," International Journal of Computer Applications, vol. 187, no. 40, pp. 1–8, 2025, [Online]. Available: https://ijcaonline.org/archives/volume187/number40/ajayi-2025-ijca-925707.pdf

[41] M. Nagy, R. Molontay, and M. Szabo, "Interpretable dropout prediction: towards XAI-based personalized intervention," International Journal of Artificial Intelligence in Education, 2024, doi: 10.1007/s40593-023-00331-8.

[42] P. Buñay-Guisñan, M. A. Al-Fedaghi, and J. Sánchez-Gordón, "Group counterfactual explanations: A use case to support students at risk of dropout," Electronics, vol. 15, no. 1, p. 51, 2026, doi: 10.3390/electronics15010051.

[43] C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995, doi: 10.1007/BF00994018.

[44] L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[45] S. M. Lundberg et al., "From local explanations to global understanding with explainable AI for trees," Nature Machine Intelligence, vol. 2, pp. 56–67, 2020, doi: 10.1038/s42256-019-0138-9.