IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 19 May 2024, accepted 10 July 2024, date of publication 15 July 2024, date of current version 23 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3428332

RESEARCH ARTICLE

TCLPI: Machine Learning-Driven Framework for Hybrid Learning Mode Identification

CHAMAN VERMA[1], ZOLTÁN ILLÉS[1], AND DEEPAK KUMAR[2]
[1]Department of Media and Educational Informatics, Eötvös Loránd University, 1053 Budapest, Hungary
[2]Apex Institute of Technology, Chandigarh University, Sahibzada Ajit Singh Nagar 140413, India

Corresponding author: Chaman Verma (chaman@inf.elte.hu)

The work of Chaman Verma was supported in part by ÚNKP, Ministry of Innovation and Technology (MIT); in part by the National Research, Development, and Innovation (NRDI) Fund, Hungarian Government; and in part by under Grant ÚNKP-23-4-II-ELTE-97.

ABSTRACT Since the COVID-19 pandemic, teachers and students have started using online and hybrid learning in education. There might be several obstacles to adopting hybrid learning in theory classes or lab practice sessions. Based on student opinions, deciding what is appropriate for theoretical class and lab practice is challenging. We employed machine learning approaches to forecast the hybrid learning mode for theory classes and lab practices. We introduce a framework that utilizes machine learning to automate the identification of hybrid learning for Theory Class and Lab practice (TCLPI). Four machine learning models form the foundation of this framework: Random Forest (RDT), Support Vector Machine (SVN), Logistic Regression (LGR), and Extreme Gradient Boosting (XBT). In the context of Theory Class Identification (TCI), the SVN achieves a maximum test accuracy of 0.93, whereas the LGR achieves a minimum accuracy of 0.90. On the other hand, the Lab Practice Identification (LPI), XBT, RDT, and SVN achieved a test accuracy of 0.80. The outcome of trained algorithms is assessed using the Shapley Additive Explanation (SHAP), an explainable Artificial intelligence (AI) approach. This research found that student-teacher interaction decreased during lab practice, which is crucial. Internet disconnections, a lack of support during technological malfunctions, and the likelihood of cheating in exams without monitoring are also issues. We also found that students were accepting of hybrid learning for theory classes. Each model's intrinsic feature relevance and SHAP values helped prove this. Research shows that hybrid learning works more for theory classes; it is less needed for lab practice for students.

INDEX TERMS ATL, classification, hybrid learning, LPI, student, SHAP, TCLPI, Prediction.

I. INTRODUCTION AND RELATED WORK

Learning involves procuring memories, skills, knowledge, and even wisdom through experiences and education. Higher learning via the technologies of the internet has transformed modern education, and many universities offer online-based courses, while others have switched to fully online programs. Incorporating information, communication, and technology is crucial in promoting student participation and facilitating online learning, which involves ICT, independent, or supplementary learning with other physical presence classes referred to as hybrids [1]. According to previous research [2], online learning threatens conventional methods by allowing flexibility and fostering social interaction. During the pandemic, students and UAE residents were positive about online education. Online resources are available to remote learners for self-guided study and collaborative learning. The self-contained web-based coaching modules included the flexibility necessary for self-directed study away from conventional teacher-centered instruction [3]. Online course strategies seek to enhance content, empower students, and cultivate collaboration. These strategies encompass suggesting projects, organizing group assignments, facilitating discussions, and encouraging virtual interactions [4]. Hybrid learning integrates face-to-face and online methods to overcome specific limitations. Fixed

The associate editor coordinating the review of this manuscript and approving it for publication was Claudio Zunino.

schedules in traditional classes limit flexibility and can hinder progress, while online learning risks student disengagement. Hybrid courses, by combining classroom and online components, enhance teacher-student relationships. This approach provides improved performance, flexible scheduling, diverse learning options, community building, and increased interaction [5]. Evaluating the cost-effectiveness and directing investments in hybrid learning heavily depends on the satisfaction and experience of learners. Cognitive engagement is closely connected to both satisfaction and experience [6]. Remote access showed enhanced effectiveness and higher student satisfaction. Assessment results supported this, indicating a 27.69% average increase in grades, and 81.25% of students favored Web-lab [7]. However, the abrupt switch to remote learning presents challenges such as connectivity issues, socioeconomic disparities, and declining student motivation.

Student feedback is essential for determining its effectiveness, considering factors such as learning styles and motivations. Adapting learning environments to individual variances can lead to better long-term learning results [8]. In Jordan, teachers are happy with new restrictions that require some courses to be taught online, while others use a hybrid method [9]. Portugal's educational system proposes online and hybrid learning pedagogy instruction [10]. Portugal's educational system proposes online and hybrid learning pedagogy instruction [10]. A hybrid learning strategy with gamification has worked well for programming classes at a Croatian university [11]. Mobile learning was used in Abu Dhabi to maintain classroom education during the pandemic [12]. Online learning and courses affected the satisfaction of Egyptian university students towards hybrid learning [13].

Amid the COVID-19 pandemic, scholars delved into the obstacles experienced by English as a Foreign Language (EFL) students in online education. These hurdles included a need for more technological proficiency, challenges with completing assignments, and problems with internet connectivity. For educators and policymakers looking to improve online education, these findings offer insightful information [14]. A study focused on the pandemic's impact on education, highlighting the rise of virtual learning and increased online program enrollment and discussing the advantages and drawbacks [15]. A few issues with the pandemic-related shift to virtual learning were examined, along with how it affected soft skills and the necessity for creative teaching strategies. They also looked at issues like cheating on online tests, the long-term impacts on instructional activities and teachers' stress, and the impact of virtual tools on social skills [16]. Considering essential factors such as resource competition and student retention, it's wise to assess how technology impacts learning thoroughly. Supporters of hybrid learning advocate for its adoption with thoughtful consideration, offering valuable perspectives from different angles [17]. Researchers highlighted the crucial need to synchronize learning objectives between traditional and ICT-based programs, emphasizing collaborative and social learning. They emphasized the vital role of social affordances in crafting productive learning settings and endorsed hybrid learning, stressing the importance of collaboration among students, educators, and researchers [18]. The authors suggested using hybrid learning spaces for curriculum enhancement, highlighting the value of technology in enhancing student skills, especially for mobile learners. Hybrid learning gained traction during the COVID-19 pandemic, which led the University of Sciences and Humanities to introduce strategies for continuous education. Following the transition from an entirely virtual setup to a hybrid model, which experienced a significant 35% dropout rate and inadequate teacher preparation, connectivity issues became apparent [19]. The swift switch to online instruction during the COVID-19 pandemic disrupted the educational landscape, posing challenges for students and educators. This prompted a comprehensive assessment, highlighting the academic importance of hybrid methodologies [20]. Namyssova et al. underscored the necessity of thorough teacher training before introducing blended learning, noting the obstacles stemming from inadequate infrastructure and restricted technology access. Identified challenges include policy gaps, insufficient faculty support, and a shortage of technological resources [21]. Students have widely acclaimed hybrid learning for its accessibility, integrating online platforms like Blackboard with face-to-face elements. The innovative teaching methods, fostering connectivity and student engagement, received high praise, as did the flexibility provided by the online component's scheduling. Challenges, including varying levels of student engagement and the necessity for instructors to adeptly manage the hybrid format, highlighted the importance of gradually integrating technology and prioritizing pedagogy in hybrid course design [22]. Engaged students demonstrated a well-distributed allocation of time between in-person and virtual tasks, preferred outdoor activities, and prioritized measures to prevent musculoskeletal problems [23]. The previously developed and validated blended learning experiment was successful for Bangladeshi students. These findings revealed positive associations with attitudes toward online learning and openness to technology and negative correlations with technology skills, learning flexibility, and study management [24].

To investigate critical features for precise predictions, this research presented a hybrid machine-learning architecture called SHAP, which uses various algorithms and an explainable AI method. Eight significant sections organize the rest of the paper. Section II outlines the study's methodology and materials. The document outlines the TCPLI framework and includes sections on dataset descriptions, statistical features of factor analysis, prepossess, testing of datasets, testing, training, and tuning of machine learning algorithms. The experiments conducted using machine learning techniques present their results in Section IV. Section V delves into the

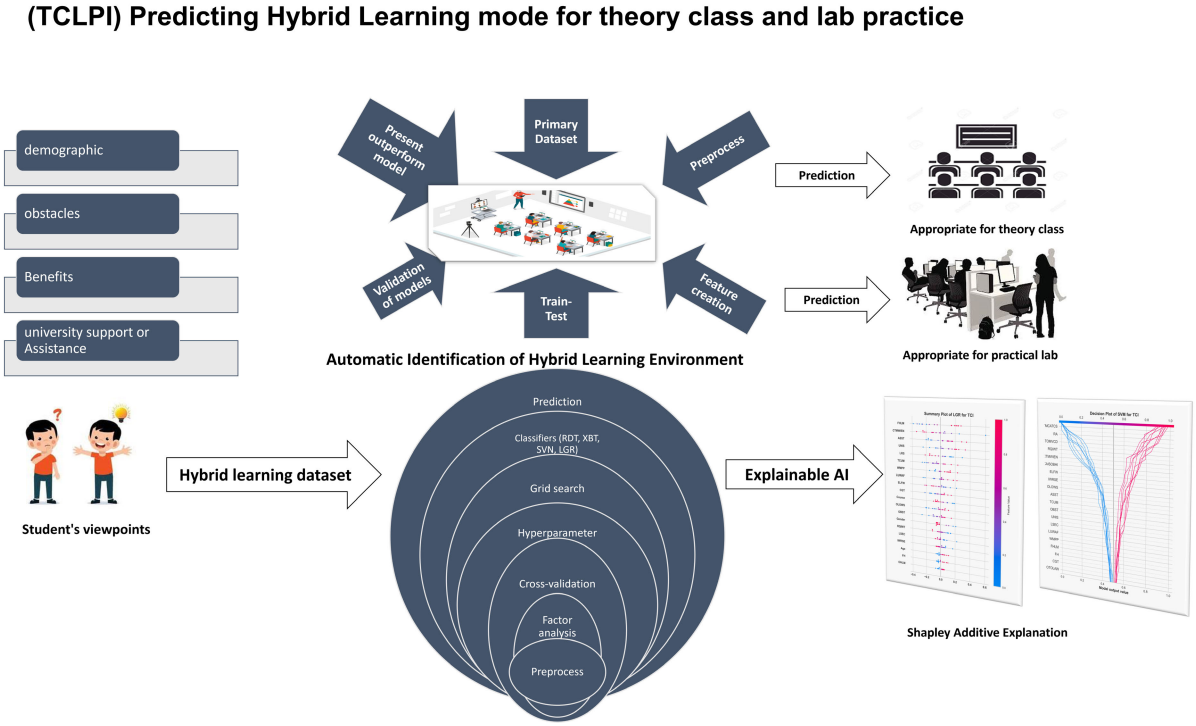


FIGURE 1. Framework for hybrid Learning mode Identification for Theory Classes and Lab Practices using Machine Learning (TCLPI).

significance of trained algorithms with SHAP features and discusses the most significant features with SHAP values. In Section VI, we offer the study's discussions, and in Section VII, we explain its findings as conclusions. Major future initiatives take up the final section VIII.

II. PROBLEM STATEMENT

A previous study did not provide a fully automated AI-based framework that predicts hybrid learning for theory classes and lab practices. Because of this, we showed the first hybrid framework with Theory Class Identification (TCI) and Lab Practice Identification (LPI), along with supervised machine learning algorithms that correctly predicted the features of hybrid learning. Further, the current research did not use the explainable artificial intelligence approach SHAP to investigate the characteristics of hybrid learning. As a result, we also investigated the benefits of using the SHAP methodology to apply machine learning algorithms. The primary performance metrics of each algorithm determine its effectiveness. After investigating novel key features and confirming their presence, we identified a hybrid learning style for theory classes and practical labs.

III. METHODS AND MATERIALS

A. CONCEPTUAL FRAMEWORK

The suggested work's visual representation embodies an archetype of morality. For both theory and lab sessions, Figure 1 foresees a hybrid learning approach that is visually identifiable. At the outset, we gather and compile the primary dataset based on student responses regarding demographic features, challenges, benefits, and assistance throughout hybrid learning. In the subsequent preparation steps, we handled missing values and standardized and balanced the data. Next, we created essential factors using a factor analysis approach. Next, we test and train four supervised machine learning algorithms on the preprocessed and cleaned samples. Crucial parameters of each algorithm were identified by grid search cross-validation with a stratified method, allowing for maximum accuracy. We also graph, graphically compute, and display learning curves, confusion matrices, classification reports, recall, precision, gain, ROC, and learning scores to illustrate the comparison of algorithms. Coefficient computation and significance estimation show how each method uses intrinsic features. Finally, we implemented the SHAP model using trained algorithms to provide a detailed description of the features.

B. DATASET DESCRIPTION, PREPROCESS AND FACTOR ANALYSIS

We used primary data samples and conducted all tests in Python 3.9 using the necessary libraries [29]. After removing outliers, the current study used 99 out of 103 students' primary data samples. ELTE University enrolled the students in its informatics program. Every year, ELTE University enrolls roughly 30,000 students, of whom 2,500 study informatics with faculty members on average. Consequently, the samples obtained were adequate for analysis for both populations and factor analysis was used to confirm this [28].

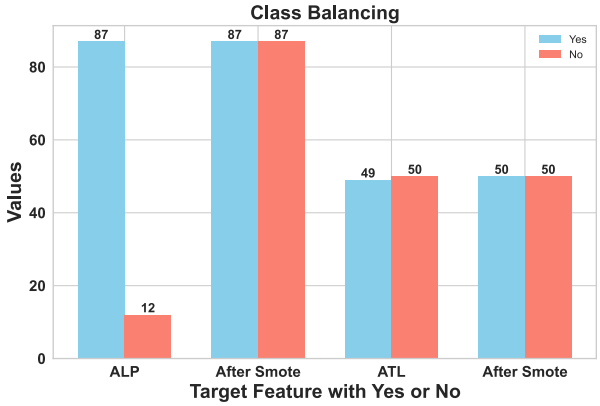


FIGURE 2. Record balance with SMOTE.

We used a random sampling strategy with Google Forms' assistance to gather samples [29]. The dataset has 58 students older than 21 and 41 younger than 20. There were 76 boys and 23 girls who participated. The master's program had thirty-one students, while the bachelor's course had sixty-eight. We selected the Google Forms questions after consulting with students and academic experts. We used the Likert scale (1-10) as the measurement tool, with 10 representing the highest level of agreement and 1 representing the lowest level of response to the questions. The dataset had no missing values. Added demographics are age, gender, and course. We divide the age group into those under 20 and those over 21. There are master's and bachelor's courses under the course feature. Eight binary features and one ternary feature were present. We split a ternary feature into three dummy features using one hot encoder.

We conducted an exploratory factor analysis to identify important factors warrant further investigation. The factor analysis utilizes the following parameters: Principal Component Analysis (PCA), Varimax Method, Anti-Image, convergence iterations set to 25, KMO (Kaiser-Meyer-Olkin), and Bartlett's test of sphericity. Table 1 displays the comprehensive sufficiency of the obtained samples for each variable, estimated at a value of 0.942. Therefore, the partial correlations among variables are minimal. The absence of an identifiable correlation matrix demonstrated a significant ($p < 0.05$) value of Bartlett's sphericity. The entire cumulative variation accounted for 72% of the participant's scores.

The dataset has 45 features. Out of 45, 19 were exempted from the factor analysis (see in Table 2), 23 were used to create 8 prominent factors (see in Table 3), and 3 were binary variables (age, gender, course). A total of 30 final features acted as predictors, and two target features were considered "ATL" and "ALP." Below, Figure 2 shows the records balanced based on target features with SMOTE [27]. In the initial imbalanced dataset, "No" is 0 and "Yes" is 1 for ATL and ALP features. To construct new synthetic training records, one or more k-nearest neighbors are randomly selected for each x record in minority class "No." We use the Euclidean distance to find neighbors. Synthetic records x_{synth}

TABLE 1. KMO and bartlett's test.

Particular	stat.
KMO	0.771
Approx. Chi-Square	1519.2
df	465
Sig.	< 0.01
Variance	67.8%

are generated randomly between the minority class's "no" record x and its i_{th} neighbor. See Equation 1 for the formula to construct x_{synth}, where x_i is the i_{th} neighbor of x.

$$x_{synth} = x + r \cdot (x_i - x) \tag{1}$$

$$z - score = \frac{x - \mu}{\sigma} \tag{2}$$

Equation 2 illustrates the StandardScaler() function's formula, where μ stands for the mean and σ for the standard deviation of the training samples. For this, a preprocessing package from the sklearn library has been used. This function has standardized the 19 features that factor analysis did not use.

Table 2 displays the statistical properties of those 19 features excluded from factor analysis [26]. We estimated the mean μ to summarize the features and the standard deviation σ to find dispersion in the features. Eight features are scaled on a 10-point Likert, so CAISOBHI has the highest mean ($\mu = 8.60$), and RQIWT has the lowest mean ($\mu = 6.51$). Online attendance was helpful for unwell students. The rest of the 11 features were binary, and students found them highly satisfied with hybrid learning ($\mu = 0.87$) and happy ($\mu = 0.68$), and they recommended them for future study ($\mu = 0.86$) and theory class ($\mu = 0.88$). Furthermore, we found that almost all features had less variation.

Table 3 displays the eight crucial variables based on significant metrics like Extraction (E) and Communalities (C) [26]. Factor analysis provides the communalities of chosen features. The communality is equal to the sum of the features' squared weights. We found that "OBST4" has the lowest communality of 0.594 and "LSEC1" has the highest communality of 0.779. The features with a communality greater than 0.58 have been selected. Thus, strong communality values described why the generated components accurately reflected these qualities. We employed varimax rotation with Kaiser normalization to reduce cross-loading and update extraction results. We no longer associate a single feature with two or three distinct factors. Consequently, we choose the extraction values using minimum cutoffs higher than 0.62.

Figure 3 displays the values generated from the rotated component matrix resulting from a PCA with a Varimax rotation and Kaiser normalization. PCA transforms variables into new factors, linear combinations of the original variables. By maximizing the variance of the squared loadings for each factor, the Varimax rotation method simplifies the interpretation of these components. The loadings indicate the

TABLE 2. Mean (μ), standard deviation (σ) of 19 features excluded from factor analysis.

Code	Features	μ	σ
FTFHL	First Time Facing Hybrid learning	0.66	0.48
TOMVCD	Turn on mic video camera disturbs	6.64	2.51
IWRSE	Isolate with real study environment	7.08	2.42
CTMWIEN	Canvas,Team, moodle are well organized and easy to navigate	7.77	1.97
ELFW	Encourgement to learn from websites	7.48	2.13
TMCATCS	Time management and convenient to attend two class at same time	6.96	2.47
RA	Reduction in absenteeism	6.82	2.32
CAISOBHI	Convenient to attend online in sickness or bad health issue	8.60	1.75
RQIWT	Reduction of Quality of Interaction with teachers	6.51	2.52
WMPP	Worthful from my parents prospective	0.82	0.39
ATL	Appropriate for Theory Lectures	0.88	0.33
ALP	Appropriate for Lab Practices	0.49	0.50
FHLM	Future Hybrid Learning Mode	0.86	0.38
LRS	Long run solution	0.73	0.45
OTOLAW	Online Theory Offline Lab Alternate Weekdays	0.47	0.50
OLOTAW	Online Lab Offline Theory Alternate Weekdays	0.15	0.36
OHLM	Online and Hybrid learning mode	0.37	0.49
FH	Feeling Happiness	0.68	0.47
OS	Overall Satisfaction	0.87	0.34

TABLE 3. Extraction (*E*) and communities (*C*) of 23 features under 8 factors.

Code	Features	C	E
OBST1	Internet disconnection during the lesson	.729	.806
OBST2	Less supportive during technical glitches or errors	.686	.742
OBST3	Less competitive environment	.670	.730
OBST4	Possiblity of cheating in exams in lack of surveillance	.594	.693
OBST5	Less Focused and Interactive	.649	.655
ASST1	Online lesson recordings are helpful	.739	.735
ASST2	Provided all course materials with recordings of lessons	.741	.686
ASST3	Video Recordings are more effective than a real classroom lecture	.660	.660
ASST4	Learnt time management skills	.596	.641
ASST5	Web resources are helpful	.727	.633
UNIS1	Worthful step taken by university	.703	.722
UNIS2	IT support and course coordinators helped to solve problems timely	.795	.703
UNIS3	Timely informed for lab and class	.682	.702
UNIS4	Appropriate for both theory and lab	.693	.638
TCUM1	Additional skills learning due to time flexibility	.621	.646
TCUM2	Better for Job oriented students	.617	.646
TCUM3	Reduction of Travel time and expenses	.601	.632
LSEC1	Less stressful than offline education	.779	.845
LSEC2	Easier to communicate with teacher	.666	.611
LUWAF	Less Interaction with another faculty	.750	.833
OLOWS1	No chance of Covid-19 epidemic in online class	.714	.690
OLOWS2	Excellent opportunity for learning online tools for freshers	.680	.640
CGT	Group tasks are challenging to manage	.649	.673

correlation between the original variables and the extracted factors. Higher absolute values indicate a stronger relationship. The loadings of OBST1-OBST5 on Factor 1 are 0.806, 0.74, 0.73, 0.69, and 0.66, suggesting a strong correlation between these variables and OBST, the first extracted factor. Similarly, ASST1-ASST5 has a loading of 0.73, 0.69, 0.66, 0.64, and 0.63 on Factor ASST, indicating a strong correlation among these variables and the second extracted factor. Also, the rest of the six factors have significant correlated variables. We also observe that the correlation values fall within the range of 0.63 to 0.85.

The eigenvalues shown in Figure 4 shed light on the degree of variation each factor can capture. The eigenvalues give information about how much variation each factor can capture. Higher eigenvalues imply greater importance in explaining the underlying data variability, while lower eigenvalues suggest less significance. OBST has the highest eigenvalue of 7.1, indicating that it captures the most

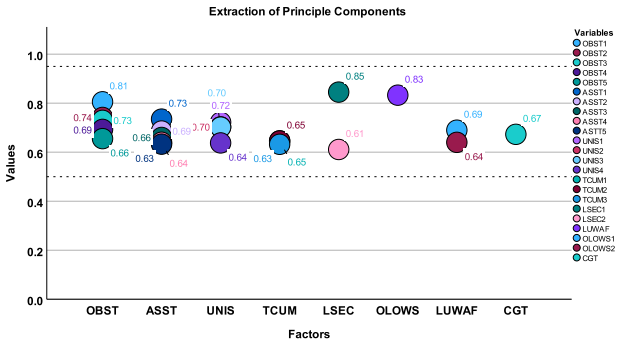


FIGURE 3. Significant factors.

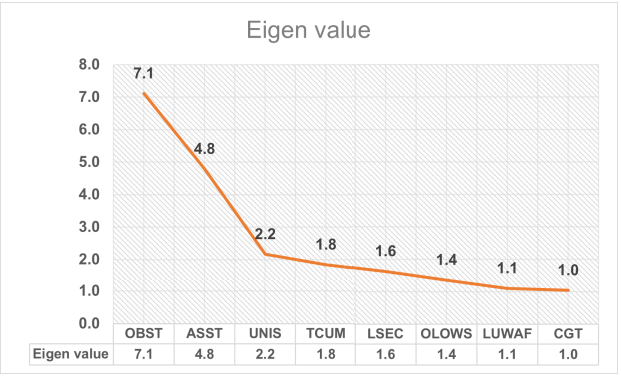


FIGURE 4. Eigen scores of factors.

variance or information among the variables. ASST has an eigenvalue of 4.8, suggesting it captures a significant amount of variance but less than OBST. UNIS, TCUM, LSEC, OLOWS, LUWAF, and CGT have progressively decreasing eigenvalues, indicating they capture successively less variance or information.

C. DATASET TRAINING AND TESTING

A well-liked method for dataset testing is Stratified Cross-Validation (SKV). To ensure that each fold has (almost) the same proportion of samples from both classes, it divides the dataset into k folds [30].

Algorithm 1 combined the SKV with TCLPI models to train and test the hybrid learning dataset (hld). A function named StratifiedKFold() holds three parameters: hld, k, and $TCLPI$. We divide the dataset hld into $k = 10$ folds, and set the variable Acc to store the accuracy. Using the *for* cycle, hld_{test} holds the test folds, and hld_{train} stores the difference between hld and hld_{test}. We trained our TCLPI four models with hld_{train} and tested them with hld_{test}. Finally, we estimate and average the accuracy of the TCLPI models using the mean() function.

These are the proposed TCI models that were tested with SKF data. Figures 5 (a) and (b) show how accurate they were during the testing and training phases. Figure 5 (a) visualizes testing accuracies obtained with four TCI models. The distribution of accuracies remains between 0.82 and 1

Algorithm 1 TCLPI With SKV

Require: Dataset hld, Number of folds k, $TCLPI$
Ensure: Average accuracy Acc

Function StratifiedKFold(hld, k, $TCLPI$)
Divide hld into k folds while preserving class distribution
$Acc \leftarrow 0$
for $i \leftarrow 1$ to k **do**
  $hld_{\text{test}} \leftarrow$ Fold i
  $hld_{\text{train}} \leftarrow hld - hld_{\text{test}}$
  Train $TCLPI$ using hld_{train}
  Test $TCLPI$ using hld_{test}
  $Acc \leftarrow Acc + \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$
end for
$Acc \leftarrow \frac{Acc}{k}$
return Acc
End Function

for XBT, SVN, and LGR. Only RDT has an accuracy of between 0.65 and 1. Figure 5 (b) depicts XBT training the dataset with an accuracy between 0.99 and 1. The SVN ranges from 0.98 to 0.99, the LGR is between 0.97 and 0.98, and the RDT is between 0.94 and 0.97. The average train accuracies of XBT, RDT, SVN, and LGR are 0.99, 0.96, 0.94, and 0.98, respectively, and the average test accuracies are 0.91, 0.90, 0.93, and 0.89. Figures 5 (c) and (d) show the accuracy distributions for both the training and testing stages of the suggested LPI models. As shown in Figure 5 (c), the testing accuracies for RDT and XBT consistently fall within the range of 0.60 to 0.90. Only LGRs and SVN test accuracies fall within the 0.60 to 0.99 range. As a result, SVN, RDT, and XBT all have average test accuracies of 0.80, while LGR comes in at 0.77. Both RDT and XBT trained the dataset with an accuracy ranging from 0.94 to 0.98, as shown in Figure 5 (d). While LGR trained samples in the 0.78-0.82 range, SVN trained samples in the 0.80-0.84 range. In descending order, the average trained accuracies of LGR, XBT, RDT, and SVN are 0.80, 0.83, 0.95, and 0.95, respectively. Therefore, both TCI and LPI models are trained and tested sufficiently.

Figure 6 visualizes the accuracy distribution of TCI and LPI models at each fold during testing and training sessions. Figure 6 (a) visualizes that in the LGR model, nine folds scored testing accuracy in the range of 0.82 to 0.94, and the last one fold scored a higher test accuracy of 1. SVN's eight folds scored around 0.94 test accuracy. RDT's four folds have 0.94, two have 1, and the rest have 0.82. The majority of folds in XBT's model also attained more than 0.8. As shown in Figure 6 (b), all of the folds in LGR achieved a score of 0.97 or higher, while the SVN achieved a score of 0.98 or higher. When it comes to XBT and RDT, the majority of folds are trained with an accuracy of 0.99. Figure 6 (c) shows the LGR model's test accuracy: three folds scored around 0.7, two folds had 0.6, and two folds had 0.9 accuracy. SVN's three

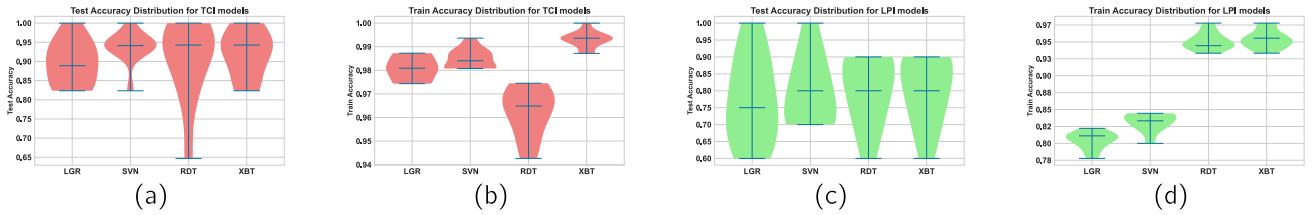


FIGURE 5. Accuracy Distribution of TCI models (a) Training (b) Testing.

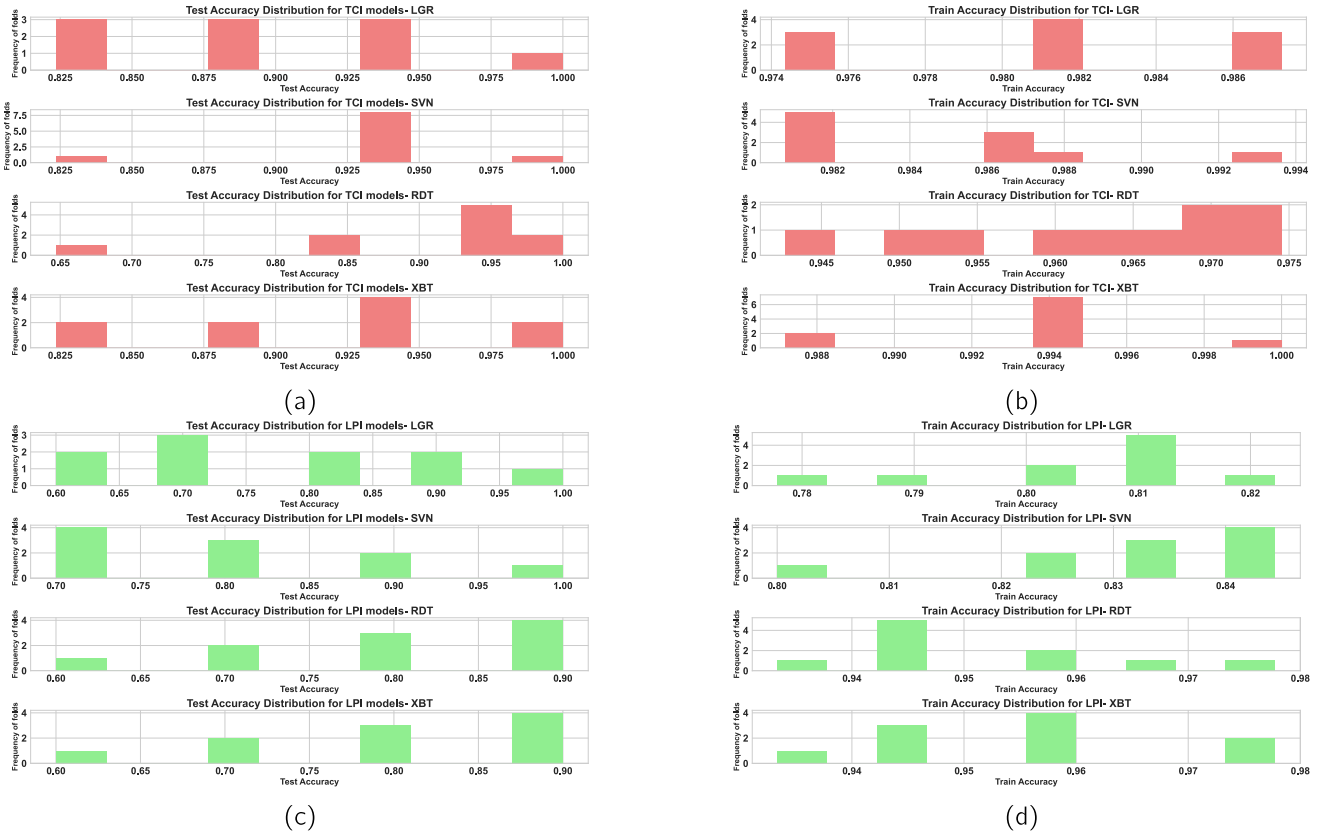


FIGURE 6. Accuracy distribution of TCI models on folds (a) Training (b) Testing.

folds have 0.8, the two folds have 0.9, and the four folds have 0.7 test accuracy. RDT's four folds have 0.9, three folds have 0.8, and two folds have 0.7 test accuracy. The four folds in XBT have a test accuracy of 0.9, whereas the three folds have a test accuracy of 0.8. Figure 6 (d) demonstrates that LGR trains maximum folds above 0.8 accuracy, while SVN maintains folds around 0.82. The folds of the XBT and RDT models exhibit significant accuracy around 0.94.

D. MACHINE LEARNING TECHNIQUES

1) LOGISTIC REGRESSION

The LGR finds the best feature weights to make a hyperplane that sorts samples into groups and reduces loss as much as possible [32]. LR uses explanatory variables to predict the logit transformation of a dependent variable. The equation restricts probabilities to a range of 0 to 1. Given the probability of outcome 1 as π_i, the probability of outcome 0 is $1-\pi_i$. Odds are calculated as $\frac{\pi_i}{1-\pi_i}$, and the logit is the logarithm of these odds [33] as illustrated in Equations 3,4.

$$\ln\left(\frac{\pi_i}{1-\pi_i}\right) = \alpha + \sum_{j=1}^{k} \theta_j z_j + \epsilon \tag{3}$$

- π_i is event probability. (y = 1).
- $1-\pi_i$ is the not event probability (y = 0).

$$\ln\left(\frac{\pi'}{1-\pi'}\right) = \alpha + \theta(z+1) = \alpha + \theta z + \theta \tag{4}$$

$$\begin{aligned} TCLPI = {} & \alpha + \theta_1 \cdot \text{Age} + \theta_2 \cdot \text{Gender} \\ & + \theta_3 \cdot \text{course} + \theta_4 \cdot \text{FTFHL} \\ & + \theta_5 \cdot \text{TOMVCD} + \theta_6 \cdot \text{IWRSE} \\ & + \theta_7 \cdot \text{CTMWIEN} + \theta_8 \cdot \text{ELFW} \end{aligned}$$

$$+\theta_9 \cdot \text{TMCATCS} + \theta_{10} \cdot \text{RA} + \theta_{11} \cdot \text{CAISOBHI} + \theta_{12} \cdot \text{RQIWT} + \ldots + \epsilon \quad (5)$$

Our framework, TCLPI, used Equation 5, where ATL and ALP are the target variables and others are predictors with coefficients (θ) and ϵ is the error term in Equation 5.

2) SUPPORT VECTOR MACHINE

Support Vector Machine (SVN) maps data to a feature space and uses an optimal hyperplane to separate two classes with maximum margin.SVN solves an optimization problem known as the primal problem to create the optimal hyperplane [34].

$$\min_{w,b,\xi,m} \left(\frac{1}{2} w^T w + C \sum_{i=1}^{N} \xi_i + m \right) \quad (6)$$

$$\text{Subject to} \quad \begin{cases} y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i, & i = 1, \ldots, N \\ \xi_i \geq 0, & i = 1, \ldots, N \end{cases} \quad (7)$$

The SVN objective function in Equation 6 tries to find the lowest value of a function that balances model complexity, which is shown by $\frac{1}{2} w^T w$, with a penalty term that is based on the sum of slack variables ($C \sum_{i=1}^{N} \xi_i$) and an extra term m. The limits in Equation 7 ensure that for each data point i, the decision function meets a margin condition of at least $1 - \xi_i$. The slack variables $\xi_i \geq 0$ let us deal with data points in our dataset that are within the margin.

3) RANDOM FOREST

Breiman's random forest model constructs decision trees in three stages: using random subsets of training data, growing trees without pruning, and operating each tree independently with identical random inputs. Predictions involve passing an instance through each tree, with the forest selecting the class with the most votes. A probabilistic understanding of the forest prediction process is needed for feature contribution. Here, C denotes the classes and K the associated set [35].

$$\Omega_K = \left\{ (q_1, \ldots, q_K) : \sum_{j=1}^{K} q_j = 1 \text{ and } q_j \geq 0 \right\} \quad (8)$$

Equation 8 defines the set Ω_K as a collection of K-tuples $(q_1, \ldots, q_K)$ where the sum of all elements q_j equals 1, and each element q_j is non-negative. This represents a probability distribution over K classes. When tree t assumes that instance i is a member of class Ck, then the prediction is expressed as $\hat{Z}_{j,t} = e_k$, linking tree forecasts to C's K probability measurements.

$$\hat{Z}_j = \frac{1}{M} \sum_{m=1}^{M} \hat{Z}_{j,m} \quad (9)$$

Equation 9 calculates the average prediction for instance j, denoted as $\hat{Z}_j$, based on the predictions from a set of M models. It is computed as the mean of the predictions from each model $\hat{Z}_{j,m}$, where m ranges from 1 to M. This method aggregates multiple model predictions to improve overall prediction accuracy.

4) EXTREME GRADEINT BOOSTING

The boosting algorithm combines feeble classifiers to construct a robust classifier. Extreme Gradient Boosting (XBT) was derived from gradient boosting to enhance speed, scalability, and generalization. XBT starts with organizing data by converting categorical data to numeric form using One Hot Encoding. Data cleaning and feature engineering come next [36]. The following equation 10 describes the derivation of the estimated model from a general function.

$$\hat{z}_j^{(n)} = \sum_{m=1}^{n} g_m(w_j) \quad (10)$$

The estimated value $\hat{z}_j^{(n)}$ for an instance j at iteration n is calculated as the sum of function $g_m(w_j)$ across iterations from 1 to n as illustrated in Equation 10. The loss function is given a second-order increase. It considers the sum of gradients s_k and hessians t_k with respect to the input variables z_k, as well as the predicted output $v_p(z_k)$. Additionally, the equation includes regularization terms βR and $\frac{1}{2}\mu \sum v_l$ to control model complexity and overfitting, as illustrated in Equation 11.

$$\sum_{k=1}^{m} L(r) \approx \sum \left(s_k v_p(z_k) + \frac{1}{2} t_k v_p(z_k)^2 \right) + \beta R + \frac{1}{2} \mu \sum v_l \quad (11)$$

5) TUNE HYPERPAMETERS

Hyperparameter tweaking is an essential step in improving the performance of machine learning algorithms. We used a grid search approach with KCV, which specifies a grid of multiple hyperparameter values. Initially, we call TCLPI models by passing a random state parameter of 0. We then input a grid of vital parameters into each algorithm to assess the model's performance.

Eq. 12 depicts the elucidation of hyperparameter optimization.

$$\mathbf{y}^* = \arg\max_{\mathbf{y} \in \mathcal{D}} f(\mathbf{y}) \quad (12)$$

In this equation, $f(y)$ shows the objective score that should be used to cut down on errors in the validation set, y^* shows the hyperparameters that give the lowest score, and x can change within the domain D. [37].

Table 4 shows the essential parameters that played a vital role in improving the accuracy of each model. The LGR's `C` parameter adjusts regularization strength to minimize overfitting. `Max_iter` determines the maximum iteration count required for solver convergence. The penalty determines the type of regularization. Ridge regularization penalizes coefficient squared magnitude (''L2''). The solver parameter

TABLE 4. Model hyperparameter tuning.

Model	Best Parameters	μ Test Accuracy
TCI-SVN	C=0.5, gamma=0.1, kernel=rbf, random_state=0, probability=true	0.93
TCI-XBT	random_state=0, learning_rate=0.05, max_depth=5, n_estimators=50, subsample=0.8, min_child_weight=1, colsample_bytree=1	0.92
TCI-RDT	criterion=gini, max_depth=10, min_samples_leaf=5, min_samples_split=10, n_estimators=250, random_state=0	0.90
TCI-LGR	C=10, Max_iter=100, penalty=l2, solver=newton-cg, random_state=0	0.90
LPI-RDT	criterion=entropy, max_depth=4, min_samples_leaf=2, min_samples_split=6, n_estimators=300, random_state=6	0.80
LPI-XBT	random_state=0, learning_rate=0.1, max_depth=5, n_estimators=50, subsample=1.0, min_child_weight=3, colsample_bytree=0.8	0.80
LPI-SVN	C=0.1, gamma=scale, kernel=poly, random_state=0, coef0=1.0, degree=4	0.80
LPI-LGR	C=0.01, max_iter=300, penalty=l2, solver=newton-cg, random_state=0	0.77

specifies the optimization algorithm. Newton-Conjugate Gradient algorithm for optmization. The `random_state` parameter seeds random number creation, assuring repeatability. In RDT, the criteria parameter specifies the split quality function. Entropy splits based on information acquisition. The `max_depth` parameter sets the maximum decision tree depth in the forest. Smaller values constrain tree depth, preventing overfitting. The `min_samples_leaf` argument determines the minimum leaf node sample count. Setting a minimum leaf node size helps control overfitting. The `min_samples_split` determines the minimum internal node splitting sample count. Setting a minimum sample size for splits helps prevent overfitting. The forest's tree count is `n_estimators`. XBT's `learning_rate` regulates step size for each iteration to minimize the loss function. Lower learning rates hinder model convergence, but they improve generalization. `Max_depth` sets the tree's maximum depth. Increasing it allows the model to capture more complex data connections, but it also increases the risk of overfitting. The number of boosting rounds (trees) is `n_estimators`. The subsample establishes the fraction of samples used to fit the base learners. All samples are given a 1.0 value. A child's instance weight minimum is `min_child_weight`. It controls overfitting. Higher values prevent models from overlearning noisy and small data. The `colsample_bytree` parameter sets each tree's random feature sampling percent. At 0.8, each tree randomly selects 80% of features during construction. The `C` parameter affects SVN's regularization, which balances margin and classification error. `Gamma` measures the impact of a single training example. It sets the decision limit by determining the training sample reach. The `kernel` is `Poly` for non-linear connections and `rbf` for linear relationships. The `coef0` governs higher-degree polynomials' decision boundary effects. Degree determines choice boundaries in polynomial degree.

IV. RESULTS

This section showed experimental results from model training using best hyperparameters. We predicted students' hybrid learning opinions for theoretical class and lab practice using LR, XBT, RDT, and SVN algorithms.

Figure 7 visualizes vital performance metrics such as learning, roc and gain curves of TCPLI models of the test data set with SKV. Figure 7(a) shows how the amount of training data added through cross-validation affects the accuracy of TCI models' predictions. It achieves this by utilizing a learning curve. At 20 samples, the training score was initially more significant than the cross-validation scores. When the sample size approaches 100, all TCI models achieve training scores of 0.90 and cross-validation scores greater than 0.82. It increases further to 0.89 at a sample size of 160. The reduced difference between the test and train results suggests that sufficient samples have undergone testing and training for classification tasks. We discovered that the samples had enough training to suit the four TCI models and predict the target feature, ATL. At several thresholds, Figure 7 (b) assesses the weights of the actual positive rate versus the false positive rate. Besides the LGR, every other has an area under the curve that separates both classes by greater than 0.91. Because the results were closer to 1 and greater than 05, all TCI models completed the classification tasks appropriately. No significant difference is observed in the area under the curves of XBT, and RDT models. The true positive rates of all models are high at 0.5 cutoffs, except LGR model. At a cutoff of 0.2, the true positive rate of SVN is 1; for RDT and XBT, it is 0.98, proving a significant classification. Figure 7 (c) displays the TCI model's efficiency using the gain curve. The cumulative true positive observation probability is calculated based on the percentage of tested samples. XBT and SVN performed better than the others in explaining 80% of tested samples, with a cumulative true positive rate of 87%. The LGR model provides the least explanation for the tested samples. As a result, if the TCI models are applied to 90% of the population, the likelihood of observation is approximately 87%. Figure 7(d) is drawn for LPI models for test and train scores on a separate sample count. We observed that the average train accuracy for LGR, SVN, RDT, and XBT is 0.80, 0.83, 0.95, and 0.95, respectively, with 10 KCV. LGR's average test accuracy is 0.77, while others are 0.80. The ROC curve for LPI model comparisons in ALP feature

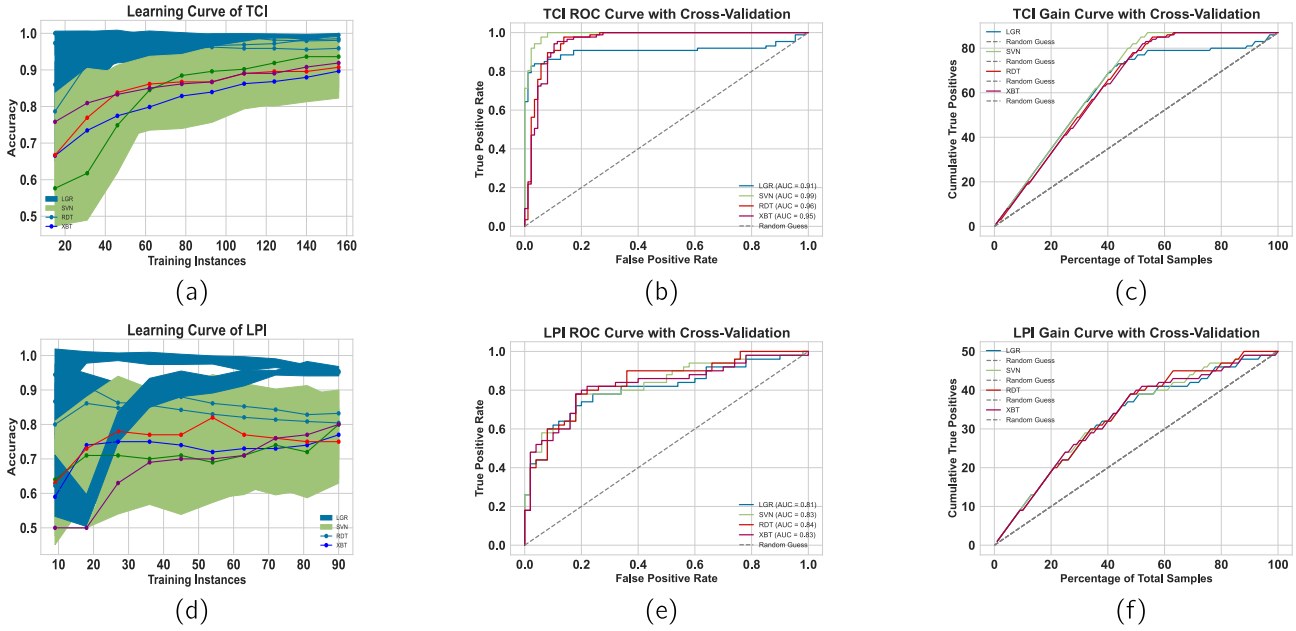


FIGURE 7. Performance of TCPLI: (a) TCI-Learning Curve (b) TCI-ROC Curve (c) TCI-Gain Curve (d) LPI-Learning Curve (e) LPI-ROC Curve (f) LPI-Gain Curve.

categorization is shown in Figure 7 (e). Each model has an AUC value above 80, indicating a balanced categorization for the binary class. The RDT model has a huge AUC curve of 0.84 among the models. The XBT and RDT models' areas under the curves show no discernible differences. At 0.5 cutoffs, all models have significant true-positive rates. The genuine positive rate is greater than 0.7 at cutoff 0.2. Due to their highest AUC and true positive rate, we discovered that the two models—XBT and RDT—are competitive. Therefore, the model's significance demonstrated its helpful categorization. Figure 7 (f) illustrates the performance of the LPI model utilising the gain curve. We calculate the cumulative probability of correctly identifying positive observations based on the examined samples' proportions. XBT and RDT outperformed the other methods by accurately explaining 100% of the studied samples, achieving a cumulative true positive rate of 50%. The LGR model offers the minimum level of explanation for the studied samples.

Figure 8 (a) shows the heatmap of the classification report of four TCI models. On average, SVN has the identical highest accuracy, f1-and recall of 0.94. The precision score for the "No" class is 1, and the precision score for the "Yes" class is 0.89. The average accuracy for LGR is 90%, RDT is 90%, and XBT is 0.92%. Compared to SVN and RDT, LGR has less recall, f1-score, and precision. Recall is defined as the ratio of true positive predictions to real positive cases. We found higher values for all the models. As a result, the TCI model feels suitable for forecasting real positives. The f1-score measures harmonic precision and recall. It provides a reasonable assessment of false positives and false negatives. More than 90% of the f1-score is found for the "Yes" and "No" categories. Figure 8(b) the heatmap of the classification report of four LPI models. The maximum test accuracy, f1-score, and recall discovered were 0.80, which SVN, LGR, and XBT all offer. These models have identical f1-scores; precision and recall scores were estimated for both classes, "Yes" and "No". Therefore, classification reports exhibited significant results for the overall performance of TCLPI models.

As illustrated in the Figure 9, the confusion matrix of TCPLI models was utilised to assess the efficacy of each classification algorithm in discerning the ALP and ATL features. The model estimated both the accurate and inaccurate classification of dataset records. Figure 9 (a)(b)(c)(d) shows that the LGR, RD, XBT, and SVN algorithms accurately identified the total number of records 156, 157, 160, and 163. On the other hand, these algorithms get records 18, 17, 14, and 11 wrong. The algorithms LGR, RD, XBT, and SVN precisely determined the total number of records 77, 80, 80, and 80, as seen in Figure 9 (e) (f) (g) (h). Conversely, these algorithms incorrectly identify records 23, 20, 20, and 20. Figure 9 (a) reflects that 85 records in the LGR model are accurately categorized as "No" whereas 71 records in the ATL fall into the "Yes" category. There are just 02 records that are wrongly categorized in the "No" class and 16 in the "Yes" class. According to Figure 9 (a), 81 samples in the RDT model can be correctly classified as "Yes" while 76 observations belong to the "No" group. Just 06 observations are incorrectly classified as belonging to the "No" class, whereas 11 are in the "Yes" class. Out of the 87 observations in Figure 9 (c) that are correctly classified as "Yes" 76 are classified as "No" in the SVN model. There are 11 observations in the "Yes" class, and none are incorrectly classified in the "No" class. Figure 9 (d) has 82 instances

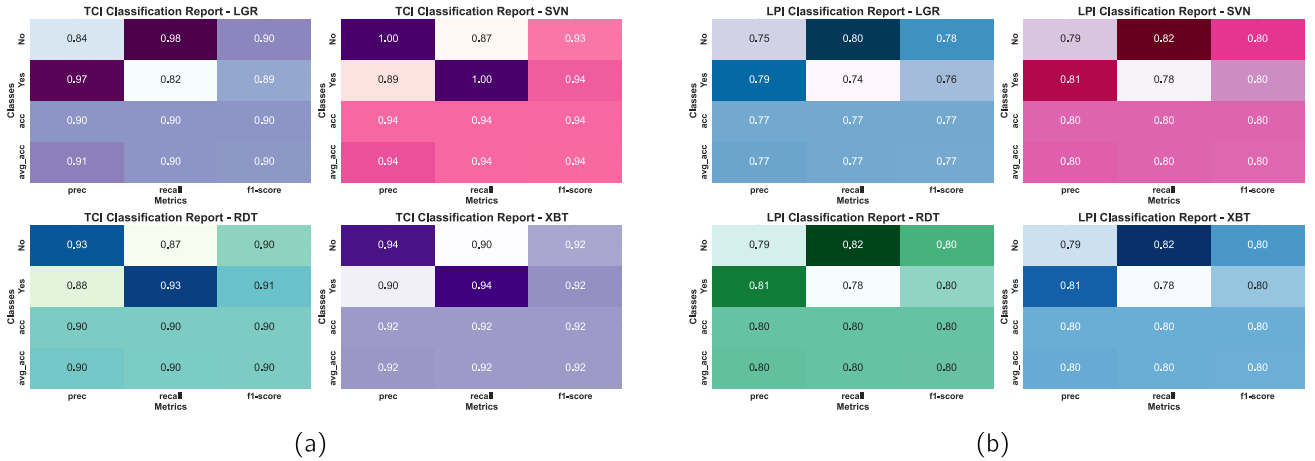


FIGURE 8. Classification Reports (a) TCI for ATL (b) LPI for ALP.

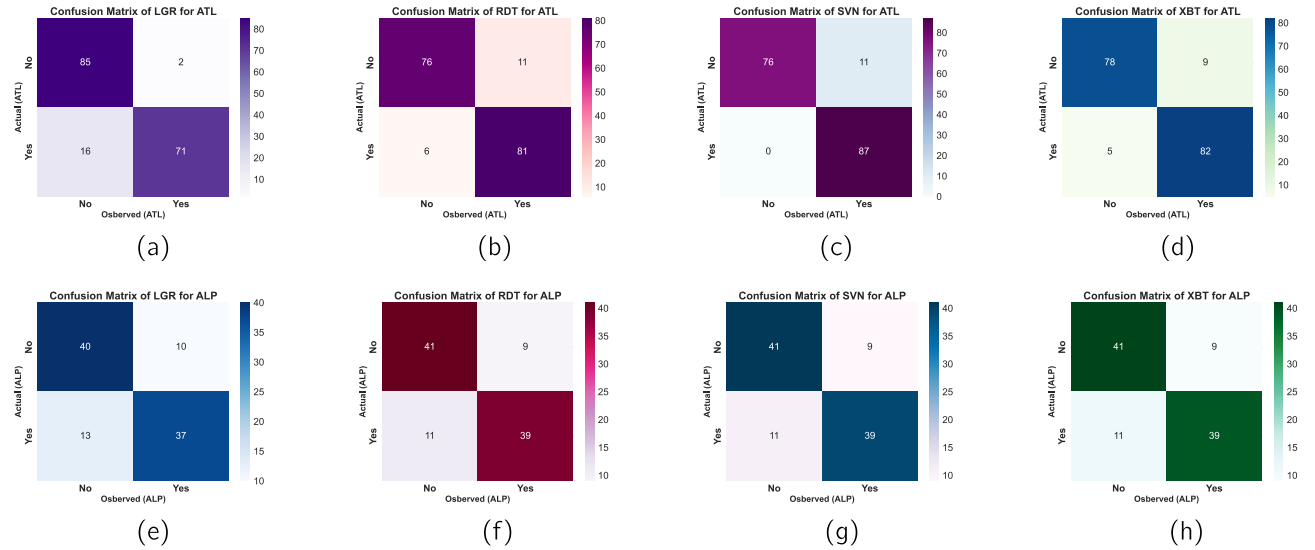


FIGURE 9. Confusion matrix of TCPLI models: (a)ATL-LGR (b) ATL-RDT (c) ATL-SVN (d) ATL-XBT (e) ALP-LGR (f) ALP-RDT (g) ALP-SVN (h) ALP-XBT.

in the XBT model accurately classified as ''Yes'' whereas 78 instances fall into the ''No'' category. There are 9 instances found misclassified as the ''No'' class and 5 misclassified as the ''Yes'' class. In Figure 9, it is evident that 40 records in the LGR model are correctly classified as ''No,'' but 37 records in the ALP model are classified as ''Yes.'' There are only 10 records that have been incorrectly classified as belonging to the ''No'' class, whereas 13 records belong to the ''Yes'' class. Looking at the information in Figure 9 (f) (g) (h), we can see that 39 of the samples in the RDT, SVN, and XBT models were correctly labeled as ''Yes,'' while 41 were labeled as ''No'' Only 11 observations are wrongly labelled as belonging to the ''No'' class, while 09 observations are classified as belonging to the ''Yes'' class.

Figure 10 shows the top ten crucial features of the TCLPI model according to the intrinsic feature estimation approach. Furthermore, the models advise against these. The LGR provided the most prominent and positive coefficient for FHLM, WMPP, UNIS, TCUM, ASST, LUWAF, and CGT, as shown in Figure 10 (a). Only three features have negative coefficient values: LRS, gender, and course. The graphs in Figure 10 (b) and (c) show the importance values of features found by RDT and XBT, respectively. Both models found 5 essential features: FHLM, WMPP, FH, Age, and IWRSE. The rest of the features found were different and unique, and the TCI classes were identified. Figure 10 (d) shows the coefficient of LGR model for LPI, where six features have positive values, and four have negative values. We observed differences in features in the top ten rankings calculated with LGR in the LPI and TCI models. Figure 10 (e) shows that RQIWT and OBST played the most crucial roles in predicting hybrid learning for lab practice using RDT, with 0.1174 and 0.1169 importance scores, respectively. The following essential features are found in

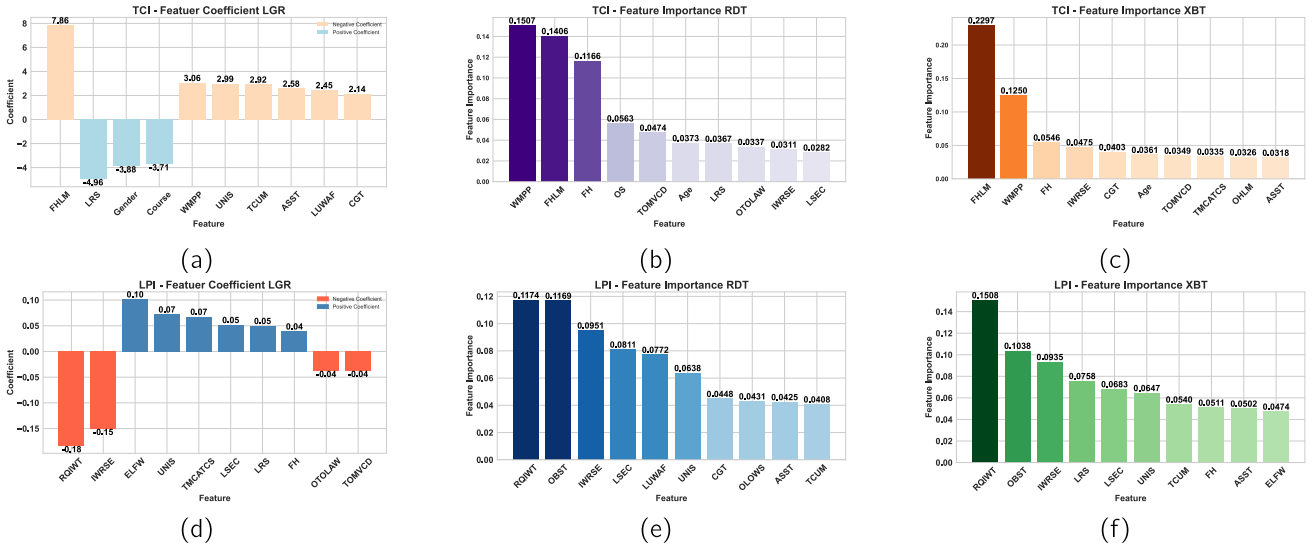


FIGURE 10. Significant intrinsic Feature: (a) TCI-LGR coefficient (b) TCI-RDT Importance, (c) TCI RDT-importance (d) LPI-LGR coefficient (e) LPI-RDT Importance (f) LPI-XDT Importance.

IWRSE, LSEC, and LUWAF, with significant importance scores of 0.0951, 0.0811, and 0.0772, respectively. The rest of the four features, UNIS, CGT, OLOWS ASST, and TCUM, also played a vital role in the classification task. As shown in Figure 10 (f), RQIWT and OBST predicted the most hybrid learning for lab practice, with XBT significance scores of 0.1508 and 0.1038, respectively. IWRSE, LRS, and LSEC have the following critical properties with significance values of 0.0935, 0.0758, and 0.0683. Four other features—UNIS, FH, ASST, ELFW, and TCUM-were also crucial in classification.

V. SHAPLEY ADDITIVE EXPLANATIONS

A SHAP that can assign a relevance value to each feature for a specific forecast [38] offers a single framework for interpreting predictions. We explain the results of the SHAP model from a game-theoretic perspective. The shape approach explores individual prediction and exhibits an increase or decrease in prediction. Local explanations and optimal credit allocation are linked by applying classic Shapley game theory values [39].

$$\phi_i(TCPLI; x) = \sum_{z_0 \subseteq x_0} \frac{|z_0|!(M - |z_0| - 1)!}{M!} [model_x(z_0) - model_x(z_0 \setminus i)] \quad (13)$$

Equation 13 figures out the average amount that each feature contributes to the variation in the TCPLI's predictions for all possible feature value combinations and orderings. Here, $\phi\_i(TCPLI; x)$ denotes the SHAP values for feature i in the TCPLI. Here, x is the input value of that feature i. $\sum_{z_0 \subseteq x_0}$ represents that z_0 is a portion of the set x_0. $|z_0|!$ is the factorial of the portion size z_0. $(M - |z_0| - 1)!$ is the factorial of the complement size of the portion z_0, and $M!$ the factorial of the total number of features. $TCPLI_x(z_0)$ represents the model prediction when only the features in the portion z_0. $TCPLI_x(z_0 \setminus i)$ denotes the TCPLI's prediction when the feature i removed from portion z_0. Importing the shap package allows the TreeExplainer class to explain the TCPLI model and calculate the SHAP values. Afterward, the SHAP values for each feature's cross-validated data sample were estimated using the shap_values() method.

Figure 11 displays the summary plot of vital features to identify the theory class with shap values. The x-axis shows the shape values, and the y-axis shows the names of features. The shap values are displayed as feature contributions for both classes "Yes" and "No". The features that have made the biggest contributions to the prediction of TCI are FHLM, CTMWIEN, ASST, UNIS, LRS, TCUM, WMPP, and LUWAF. Figure 11 (a) displays the LGR model's shape values. In contrast, CTMWIEN, LRS, ELFW, gender, and IWRSE maximum voted for the NO class, while the FHLM, ASST, UNIS, and TCUM favored the "Yes" class. Figure 11 (b) shows that the RDT model predicted that FHLM, WMPP, OS, OTOLAW, and RA all contributed equally to the prediction task and did a great job of finding the "Yes" class. Figure 11 (c) demonstrates that the characteristics that SVN recommended: TMCATCS, RQIWT, RA, and TOMVCD—are strongly contributed. It also confirms that the features that contributed less were FHLM, course, and FH. RA, TMCATCS, and IWARSE favored the "Yes" class, while TOMVCD, ELFW, and ASTT supported the "No" class. The features with low shape values contributed equally to both classes. Figure 11 (d) shows that the XBT model found that FHLM, WMPP, FH, TOMVCD, and OTOLAW were very important in predicting the "Yes" class, even though other factors also played a part. Figures 11 (e), (f), (g), and (h) show shap summary charts

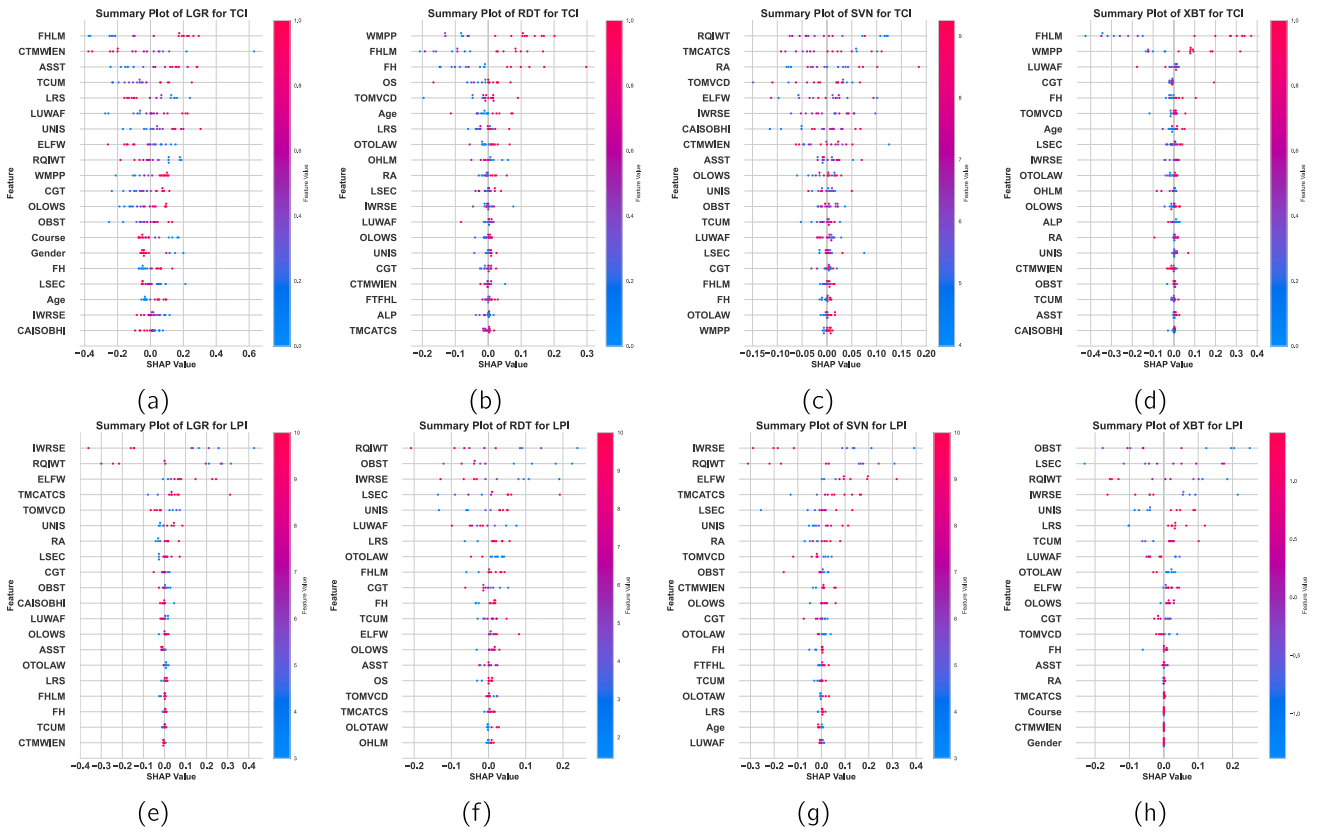


FIGURE 11. SHAP Summary of TCPLI: (a) TCI-LGR (b) TCI-RDT (c) TCI-SVN (d) TCI-XBT (e) LPI-LGR (f) LPI-RDT (g) LPI-SVN (h) LPI-XBT.

of four LPI models each. Figure (e) (g) shows that the LGR and SVN investigated the same and most influential aspects: IWRSE, RQIWT, ELFW, TMCATCS, UNIS, and RA. Blue dots indicate that these qualities contributed significantly to the "Yes" class, whereas red dots supported the "No" class. The RDT demonstrated several vital properties for classifying the two classes. Figures 11 (f) (g) show that both RDT and XBT supported RQIWT, OBST, IWRSE, LSEC, and UNIS as important features.

Figure 12 shows decision plots to show the real effect of the features on the model's prediction of theory class for each instance. The bottom x-axis plots the log odds against probabilities or model output. The y-axis presents the TCI's features in descending order of estimated significance over the plotted records. The top x-axis reflects the SHAP values that show the real impact of features on the model's performance. A colored line represents the prediction for each record. Every line in the plot's upper region crosses the x-axis at the matching record's predicted value. We increase the model's base value by adding the SHAP values for each feature, working from the bottom to the top of the plot. It illustrates the individual impact of each attribute on the accuracte prediction. Figure 12 (a) shows that LGR's SHAP values lie between -10 and 10, favoring features FHLM, CTMWIEN, ASST, UNIS, LRS, and others. Predictions are higher for higher feature values, but those pointing to the left imply the contrary. Figure 12 (b) graphs the plot for the RDT model to identify the ATL with influence features FHLM, WMPP, FH, OS, OTOLAW, and others; the range of these features lies in between 0 to 0.8. Figure 12 (c) visualizes SVN's SHAP values lying between 0 and 1 that explored TMCATCS, RA, TOMVCD, CTMWIEN, RQIWT, and others. The SVN's ability to classify the ATL feature was excellent. When these feature values raise the model's average prediction value, the line turns red. Conversely, if the average prediction value decreases, the line changes to blue. Essentially, every line in the plot represents a distinct record of ATL. Figure 12 (d) displays the shape values of XBT within a range of -3 to 2 to pretend FHLM, WMPP, FH, TOMVCD, and others to identify the ATL. To detect lab practices, plots of Figures 12 (e), (f), (g), and (h) for LPI models were used. Figure 12 (e) shows that the LGR's SHAP values are -2 to 2. These values favor traits like RQIWT, IWRSE, ELFW, TMCATCS, TOMVCD, and others. These are extremely distributed to estimate larger shap values. The elements of the RDT model with higher shape values include OBST, IWRSE, RQIWT, LSEC, UNIS, and others, as shown in Figure 12 (f). These characteristics have a range of 0.2 to 0.8. The SHAP values for SVN, which range from 0 to 1, are shown in Figure 12 (g). These values cover RQIWT, IWRSE, ELFW, TMCATCS, LSEC, and other features. The shape values of XBT are shown in Figure 12 (h) in a range of -3

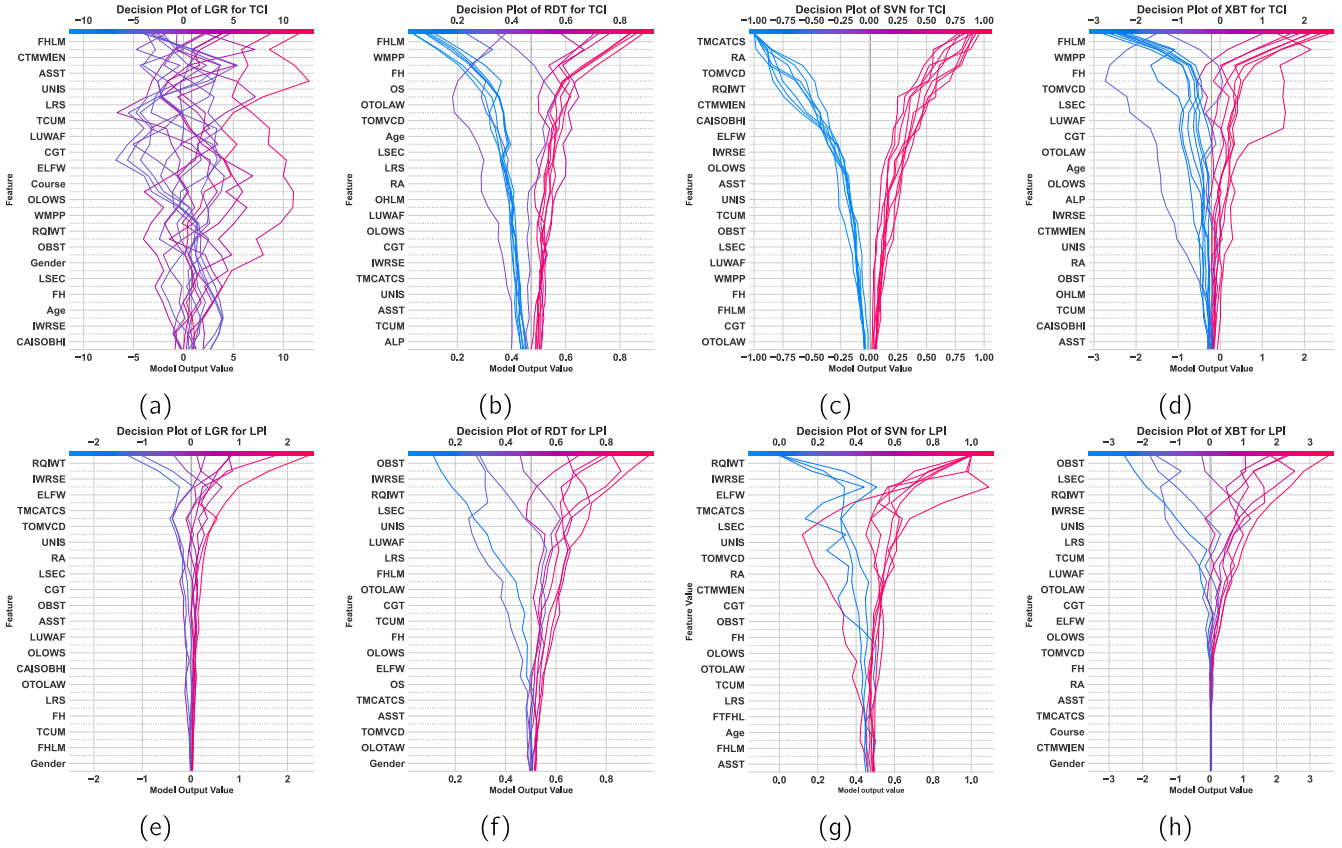


FIGURE 12. SHAP Decision TCPLI: (a) TCI-LGR (b) TCI-RDT (c) TCI-SVN (d) TCI-XBT (e) LPI-LGR (f) LPI-RDT (g) LPI-SVN (h) LPI-XBT.

to 3, which is used to mimic OBST, LSEC, RQIWT, IWRSE for the identification of lab.

VI. DISCUSSION

This study uses machine learning and explainable AI to predict informatics students' hybrid learning for theoretical class and lab practice. All the algorithms achieved training scores of 0.90 and cross-validation scores higher than 0.82, while the sample size was approaching 100—the process of identifying the theory class accomplished this. For the lab identification task, 90 samples are sufficient to train 0.95 to 0.80 accuracy scores. Hence, more samples can reduce this gap between test and train scores. All models estimated more than 0.90 AUC values to identify the theoretical class while retaining 0.81-0.84 AUC ratings to identify lab practice. In theory class identification, XBT and SVN did better than others, explaining 80% of samples and having a cumulative actual positive rate of 87%. In lab practice identification, XBT and RDT were 100% accurate, with a cumulative true positive rate of 50%. For theory class prediction, SVN, XBT, and RDT had the same most excellent mean accuracy, f1-score, and recall of 0.94; for lab practice prediction, it was 0.80. In theory class prediction, LGR, RD, XBT, and SVN algorithms correctly identified 156, 157, 160, and 163, respectively. However, these algorithms misinterpret records 18, 17, 14, and 11, respectively. The algorithms LGR, RD, XBT, and SVN accurately determined the total number of records as 77, 80, 80, and 80, respectively, for lab identification. However, these algorithms wrongly identified records as 23, 20, 20, and 20, respectively.

When making predictions in theory class, factors such as gender, course, future hybrid learning mode, and long-term solutions are essential. Additionally, my parents' valuable perspective also plays a significant role. It is crucial to consider factors such as reduced quality of interaction with teachers, hurdles encountered, isolation from a natural study environment, lower stress levels, ease of communication, and university assistance when predicting laboratory outcomes. Significant features emerge from these factors. The SVN model's SHAP scores in theoretical class prediction, teacher interaction quality, and absence reduction were substantial. Other noteworthy aspects included time management, the convenience of attending two classes at once, the ability to turn on mic video camera interrupts, the encouragement of website learning, and the isolation of a natural study environment. The XBT, RDT, and SVN model SHAP values forecast lab practice, which isolates university initiatives and support within a genuine study environment. Lower teacher interaction quality, less stress, and more accessible communication were all essential features.

VII. CONCLUSION

In this study, we employed machine learning techniques to identify the possibilities of incorporating hybrid learning into the education of informatics students, whether in lab practice or theory classes. Furthermore, we evaluate the explanatory power of the proposed models using an explainable AI approach (SHAP). We also presented a combined framework that identifies the applicability of hybrid learning with significant accuracy. The factor analysis technique created novel and essential factors for the experiments. The SVN algorithms achieved a test accuracy of 0.93% in predicting hybrid learning for theoretical classes, while XBT achieved the second-greatest accuracy of 92%. RDT and LGR achieved an identical accuracy rate of 90%. We demonstrated that students and parents eagerly accept hybrid learning for theory classes. Each model's intrinsic feature importance and SHAP values proved this. While a viable solution for theory classes, hybrid learning lacks significance for lab practice. Using hybrid learning makes students less stressed and makes it easy to join two different classes simultaneously. They also believed the hybrid learning mode separated them from the natural study environments. To predict the hybrid learning for lab practice, RDT, XBT, and SVN scored a homogeneous accuracy of 80%, and LGR scored the lowest accuracy of 0.77%. During the lab practice, there was a decrease in the quality of interaction between the students and the teachers, which is an important aspect. Furthermore, students face challenges, including intermittent Internet access, inadequate assistance in the event of technical difficulties, and the risk of exam cheating as a result of insufficient monitoring. They experience a decrease in their ability to concentrate and engage with people.

VIII. FUTURE STUDY

The present research has identified the features of hybrid learning that played a vital role in predicting its applicability in informatics students' lab practice and theory classes. Future experiments could involve students from other domains like medicine, economics, social science, or physiology. We can also implement other algorithms like artificial neural networks and discriminant analysis. Future hybrid learning mode features allow for the extraction of additional features. Furthermore, we can apply more feature selection algorithms, like gain ratio and info-gain, to the samples. Local Interpretable Model-Agnostic Explanations (LIME) is a new machine-learning explanation method that can provide more dataset insights. After increasing accuracy, we must deploy the models as Python-based applications to put this research into actual use.

CONFLICTS OF INTEREST

The authors declare that there is no conflict of interest regarding the publication of this article.

ACKNOWLEDGMENT

Chaman Verma like to thank the Faculty of Informatics, Eötvös Loránd University, Budapest, Hungary, for supporting and encouraging him to work on this research. This paper belongs to his postdoctoral study.

REFERENCES

[1] A. T. T. Wong, "5i: A design framework for hybrid learning," in *Hybrid Learning and Education* (Lecture Notes in Computer Science), Hong Kong, Aug. 2008, pp. 147–156. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-540-85170-7_13

[2] T. M. Osaili, L. C. Ismail, H. M. ElMehdi, A. A. Al-Nabulsi, A. O. Taybeh, S. T. Saleh, H. Kassem, H. Alkhalidy, H. I. Ali, A. S. Al Dhaheri, and L. Stojanovska, "Comparison of students' perceptions of online and hybrid learning modalities during the COVID-19 pandemic: The case of the University of Sharjah," *PLoS ONE*, vol. 18, no. 3, Mar. 2023, Art. no. e0283513, doi: 10.1371/journal.pone.0283513.

[3] N. Z. Zacharis, "A multivariate approach to predicting student outcomes in web-enabled blended learning courses," *Internet Higher Educ.*, vol. 27, pp. 44–53, Oct. 2015, doi: 10.1016/j.iheduc.2015.05.002.

[4] A. Zapalska and D. Brozik, "Learning styles and online education," *Int. J. Inf. Learn. Technol.*, vol. 24, no. 1, pp. 1–6, 2007, doi: 10.1108/10650740610714080.

[5] B. El Mansour and D. M. Mupinga, "Students' positive and negative experience in hybrid and online classes," *College Student J.*, vol. 41, no. 1, pp. 242–248, 2007.

[6] J. Xiao, H. Sun-Lin, T. Lin, M. Li, Z. Pan, and H. Cheng, "What makes learners a good fit for hybrid learning? Learning competences as predictors of experience and satisfaction in hybrid learning space," *Brit. J. Educ. Technol.*, vol. 51, no. 4, pp. 1203–1219, May 2020, doi: 10.1111/bjet.12949.

[7] P. C. S. Euphrásio, L. A. Faria, J. S. E. Germano, and D. Hirata, "Improving teaching–learning process in MIL-STD-1553B bus classes using a new hybrid web-lab methodology," *IEEE Trans. Educ.*, vol. 63, no. 4, pp. 291–298, Nov. 2020, doi: 10.1109/TE.2020.2984882.

[8] B. Uğur, B. Akkoyunlu, and S. Kurbanoğlu, "Students' opinions on blended learning and its implementation in terms of their learning styles," *Educ. Inf. Technol.*, vol. 16, no. 1, pp. 5–23, Oct. 2009, doi: 10.1007/s10639-009-9109-9.

[9] X. Xu, "A survey on students' satisfaction with college English IV course in blended learning mode," in *Proc. 2nd Int. Conf. Big Data Eng. Educ. (BDEE)*, Aug. 2022, pp. 221–227. [Online]. Available: https://ieeexplore.ieee.org/document/9980848/

[10] I. Pereira, J. Ferreira, M. Francisco, C. Rodrigues, C. Esperança, J. Mineiro, S. Pedro, C. Maximiano, and M. Almeida, "Teacher training as a starting point to support the initial challenges of emergency remote learning–case study," in *Proc. Int. Symp. Comput. Educ. (SIIE)*, Sep. 2021, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/9583660

[11] G. Đambic, T. Keščec, and D. Kučak, "A blended learning with gamification approach for teaching programming courses in higher education," in *Proc. 44th Int. Conv. Inf., Commun. Electron. Technol. (MIPRO)*, Sep. 2021, pp. 843–847. [Online]. Available: https://ieeexplore.ieee.org/document/9597167

[12] K. Alhumaid, M. Habes, and S. A. Salloum, "Examining the factors influencing the mobile learning usage during COVID-19 pandemic: An integrated SEM-ANN method," *IEEE Access*, vol. 9, pp. 102567–102578, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9488198

[13] D. El Kayaly, N. Hazem, and I. Fahim, "Online teaching at Egyptian private universities during COVID 19: Lessons learned," in *Proc. 6th Int. Conf. e-Learn. (econf)*, Dec. 2020, pp. 22–27. [Online]. Available: https://ieeexplore.ieee.org/document/9385458

[14] S. Khatoony and M. Nezhadmehr, "EFL teachers' challenges in integration of technology for online classrooms during Coronavirus (COVID-19) pandemic in Iran," *Asian J. English Lang. Pedagogy*, vol. 8, no. 2, pp. 89–104, 2020. [Online]. Available: https://ejournal.upsi.edu.my/index.php/AJELP/article/view/3523

[15] B. Surkhali and C. K. Garbuja, "Virtual learning during COVID-19 pandemic: Pros and cons," *J. Lumbini Med. College*, vol. 8, no. 1, pp. 154–155, 2020. [Online]. Available: https://www.jlmc.edu.np/index.php/JLMC/article/view/363

[16] M. M. Ahamed and S. Siddiqui, "From the normal to the new normal: Reflecting metamorphosis in pedagogies," *Arab World English J.*, vol. 7, no. 1, pp. 419–434, Jul. 2021. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3906339

[17] S. Wainwright, "Hybrid learning: The perils and promise of blending online and face-to-face instruction in higher education," *J. Phys. Therapy Educ.*, vol. 25, no. 1, p. 73, 2011. [Online]. Available: https://journals.lww.com/jopte/fulltext/2011/10000/hybrid_learning

[18] D. Bennett, E. Knight, and J. Rowley, "The role of hybrid learning spaces in enhancing higher education students' employability," *Brit. J. Educ. Technol.*, vol. 51, no. 4, pp. 1188–1202, Jul. 2020. [Online]. Available: https://bera-journals.onlinelibrary.wiley.com/doi/full/10.1111/bjet.12931

[19] J. Corzo-Zavaleta, R. Yon-Alva, J. Icho-Yacupoma, Y. Principe Somoza, L. Andrade-Arenas, and N. I. Vargas-Cuentas, "Hybrid learning in times of pandemic COVID-19: An experience in a Lima University," *Int. J. Eng. Pedagogy*, vol. 13, no. 1, pp. 65–81, Feb. 2023, doi: 10.3991/ijep.v13i1.36393.

[20] F. Nechita, G. G. Răţulea, M. Borcoman, D. Sorea, and L. M. Leluţiu, "Hybrid events as a sustainable educational approach for higher education," *Trends Higher Educ.*, vol. 2, no. 1, pp. 29–44, Jan. 2023. [Online]. Available: https: https://www.mdpi.com/2813-4346/2/1/3

[21] G. Namyssova, G. Tussupbekova, J. Helmer, K. Malone, M. Afzal, and D. Jonbekova, "Challenges and benefits of blended learning in higher education," *Int. J. Technol. Educ.*, vol. 2, no. 1, pp. 22–31, 2019. [Online]. Available: https://eric.ed.gov/?id=EJ1264247

[22] O. Lin, "Student views of hybrid learning: A one-year exploratory study," *J. Comput. Teacher Educ.*, vol. 25, no. 2, pp. 57–66, Dec. 2008. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/10402454.2008.10784610

[23] G. B. Glazkova, E. A. Lubyshev, O. V. Mamonova, and M. N. Pukhovskaya, "Students' attitude to physical activity under blended learning," *Theory Pract. Phys. Culture*, vol. 1, pp. 52–54, Jan. 2023. [Online]. Available: https://cyberleninka.ru/article/n/students-attitude-to-physical-activity-under-blended-learning

[24] S. I. Shakeel, M. F. A. Haolader, and M. S. Sultana, "Exploring dimensions of blended learning readiness: Validation of scale and assessing blended learning readiness in the context of TVET Bangladesh," *Heliyon*, vol. 9, no. 1, Jan. 2023, Art. no. e12766, doi: 10.1016/j.heliyon.2022.e12766.

[25] C. Verma. (2023). *Hybrid Learning Dataset*. [Online]. Available: https://forms.gle/wx1N6XewB8oAwTVa8

[26] C. Verma, Z. Illés, and D. Kumar, "An investigation of novel features for predicting student happiness in hybrid learning platforms—An exploration using experiments on trace data," *Int. J. Inf. Manage. Data Insights*, vol. 4, no. 1, Apr. 2024, Art. no. 100219. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2667096824000089

[27] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002.

[28] *Survey Monkey*. Accessed: Jan. 1, 2024. [Online]. Available: https://www.surveymonkey.com/mp/sample-size-calculator/

[29] (2024). *Google Form*. Accessed: Jan. 1, 2024. [Online]. Available: https://forms.gle/wx1N6XewB8oAwTVa8

[30] *Sklearn*. Accessed: Apr. 2, 2024. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.StratifiedKFold.html

[31] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, pp. 5–32, Oct. 2001.

[32] J. Sheng, S. Wu, Q. Zhang, Z. Li, and H. Huang, "A binary classification study of Alzheimer's disease based on a novel subclass weighted logistic regression method," *IEEE Access*, vol. 10, pp. 68846–68856, 2022. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9809963

[33] H. P. Singh and H. N. Alhulail, "Predicting student-teachers dropout risk and early identification: A four-step logistic regression approach," *IEEE Access*, vol. 10, pp. 6470–6482, 2022. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9676659

[34] S. Kalyani and K. S. Swarup, "Binary SVM approach for security assessment and classification in power systems," in *Proc. Annu. IEEE India Conf.*, Dec. 2009, pp. 1–4, doi: 10.1109/INDCON.2009.5409433.

[35] A. Palczewska, J. Palczewski, R. M. Robinson, and D. Neagu, "Interpreting random forest classification models using a feature contribution method," in *Integration of Reusable Systems*, vol. 193. El Segundo, CA, USA: Springer, 2014, pp. 193–218. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-04717-1_9

[36] S. B. S. Lai, N. H. N. B. M. Shahri, M. B. Mohamad, H. A. B. A. Rahman, and A. B. Rambli, "Comparing the performance of AdaBoost, XGBoost, and logistic regression for imbalanced data," *Math. Statist.*, vol. 9, no. 3, pp. 379–385, May 2021.

[37] D. Kumar, C. Verma, S. Dahiya, P. K. Singh, M. S. Raboaca, Z. Illés, and B. Bakariya, "Cardiac diagnostic feature and demographic identification (CDF-DI): An IoT enabled healthcare framework using machine learning," *Sensors*, vol. 21, no. 19, p. 6584, Oct. 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/19/6584

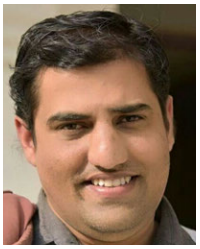
[38] S. M. Lundberg and S. Lee, "A unified approach to interpreting model predictions," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Long Beach, CA, USA, 2017, pp. 4765–4774.

[39] *SHAP Documentation*. Accessed: Feb. 1, 2024. [Online]. Available: https://shap.readthedocs.io/en/latest/

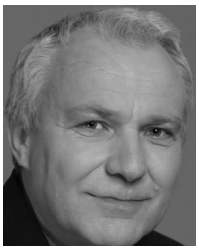
[40] C. Verma, V. Stoffova, and Z. Illes, "Gender prediction of the European school's teachers using machine learning: Preliminary results," in *Proc. IEEE 8th Int. Advance Comput. Conf. (IACC)*, Dec. 2018, pp. 213–220, doi: 10.1109/IADCC.2018.8692100.

[41] A. Field, *Discovering Statistics Using IBM SPSS Statistics*. Newbury Park, CA, USA: Sage, 2013.

[42] D. George and P. Mallery, *SPSS for Windows Step by Step: A Simple Guide and Reference, 17.0 Update*, 10th ed., Boston, MA, USA: Pearson, 2010.



CHAMAN VERMA received the Ph.D. degree in informatics from the Doctoral School of Informatics, Eötvös Loránd University, Budapest, Hungary. He is currently an Assistant Professor with the Department of Media and Educational Informatics, Faculty of Informatics, Eötvös Loránd University. He is performing his additional duty as the Project Leader in his own project supported by the Digital Education Development Competence Centre Project (DOKK), NRDI, Hungary. He is also a Postdoctoral Researcher of statistical and machine learning, financed under the NKP Scholarship by the Ministry of Innovation and Technology and the National Research, Development and Innovation Office (NRDIO) Fund, Government of Hungary. He has around ten years of experience in teaching and industry. He reviews many scientific journals, including IEEE, Springer, Elsevier, Wiley, and MDPI. He has Scopus citations of 1121 with an H-index of 18. He has Web of Science citations 91 with an H-index of nine. He has more than 100 scientific publications in IEEE, Elsevier, Springer, IOP Science, MDPI, and Walter de Gruyter. His research interests include data analytics, the IoT, feature engineering, real-time systems, and educational informatics. He is a Life Member of ISTE, New Delhi, India; a member of the Editorial Board; and a reviewer of various international journals and scientific conferences. He was a recipient of the Best Scientific Publication Award from the Faculty of Informatics, Eötvös Loránd University, from 2021 to 2024. He has been awarded three times the NKP Scholarship for research by the Ministry of Innovation and Technology and NRDIO Fund, from 2021 to 2023. During the Ph.D. degree, he won the EFOP Scholarship, Co-Founded by European Union Social Fund and the Government of Hungary, as a Professional Research Assistant in a real-time system, from 2018 to 2021, and the Stipendium Hunagricum Scholarship funded by the Tempus Public Foundation, Government of Hungary. He also received the Stipendium Hungaricum Dissertation Scholarship of the Tempus Public Foundation, Government of Hungary, from 2021 to 2022. He has been awarded several Erasmus scholarships for conducting international research and academic collaboration with European and non-European universities. He was the Guest Editor of several Springer and MDPI journals from 2022 to 2024. He is the Co-Editor of the series of conference proceedings of ICRIC-2021–2024 published by Springer, Singapore.



ZOLTÁN ILLÉS received the Ph.D. and Habilitation degrees in mathematics and physics from Eötvös Loránd University. Later, he took up the computer science supplementary course, which was started at that time. His Ph.D. dissertation was titled "Implementation of Real-Time Measurements for High-Energy Ion Radiations," in 2001. After graduating in 1985, he joined the Department of Computer Science, Eötvös Loránd University. In 2004, at the request of Jedlik Publisher, he also wrote a textbook on the C programming language. This book has a second, expanded edition in 2008. He received a scholarship from the Slovak Academy of Sciences in 2007 and he spent six months conducting research and instructing at Constantine the Philosopher University, Nitra. The NJSZT awarded the Rezs Tarjn Prize, in 2016, for the success of the joint work that has been going on ever since. He and his colleagues also researched the issue of mobile devices and applications in the framework of a tender won, in 2014. Based on their research findings, he launched a pilot project to support real-time, innovative performance management. The first results of this research are an integral part of the habilitation dissertation. He got a Hungarian Republic Scholarship based on the outstanding academic achievements during university studies. Since 2020, he has been an invited speaker at several international conferences and an Amity University Advisory Board Member.

DEEPAK KUMAR is currently a Professor with Chandigarh University, Sahibzada Ajit Singh Nagar. He also been the Ex-Director/Principal with the Bharat Group of Colleges, Sardulgarh, Mansa, Punjab. He has 16 years of teaching experience. He also successfully organized the Winning Camp (Technical Placement Preparation Event), from July 2022 to July 2022, with the Apex Institute of Technology, Chandigarh University. He has made significant contribution through various approaches, such as directly supervising the Ph.D. student, having acquired one patent, publishing more than 18 publications of research papers in international and national journals/conferences viz. MDPI and Springer. His research interest includes machine learning. He has been invited as a keynote speaker at various prestigious FDP in Akal University, Bathinda. He took parts in various international conference as the session chair.