Journal of Theoretical and Applied Information Technology
15st September 2026. Vol.104. No.17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ADAPTIVE DEEP MULTI-TIER ENSEMBLE FRAMEWORK WITH TEMPORAL FEATURE ENGINEERING AND GENERATIVE CLASS BALANCING FOR PRECISION STUDENT OUTCOME PREDICTION IN E-LEARNING ENVIRONMENTS

T MAHALAKSHMI [1], HEMANTH KUMAR VIROTHI[2], KARUNA ARAVA[3], MAKKAPATI SATYA SUKUMAR[4], S SINDHURA[5*], VALLAM REDDY BHARGAVI REDDY[6]

[1]Assistant Professor, Department of ECE, Prasad V Potluri Siddhartha Institute of Technology, Vijayawada, Andhra Pradesh, India
[2]Project SDET II, Government & Public Sevices, Deloitte Consulting LLP, Quality Engineering & Human Services Transformation (GPS), Visakhapatnam, India.
[3]Assistant Professor, Department of Computer Science & Engineering, University College of Engineering Kakinada, JNTUK, Kakinada, Andhra Pradesh, India
[4]Manager - Projects, Quality Engineering & Assurance Business Unit, Cognizant Technology Solutions US Corp
[5*]Assistant Professor, Department of CSE, Dr.RVR NRI Institute of Technology Deemed to be University, Pothavarappadu Village, Andhra Pradesh, India
[6]Assistant Professor, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, 522302, India

[*]ssindhurapraveen@gmail.com

ABSTRACT

The task of student outcome prediction in e-learning platforms represents an extremely challenging high-dimensional analytics problem, being characterized by complex temporal dependencies, severe class imbalance, feature redundancy, and heterogeneous data modalities. None of the existing solutions are able to tackle all the aspects in an integrated and end-to-end predictive architecture that seamlessly integrates feature engineering, class balancing, model optimisation and explainability. The ADMEF (Adaptive Deep Multi-tier Ensemble Framework) is a new four-component pipeline that addresses each challenge individually, with complementary mechanisms working together. An LSTM-Autoencoder is proposed for temporal feature distillation, able to extract patterns of engagement trends from clickstream and activity logs; AF-3TS (Adaptive Filter-Wrapper-Embedded Three-Tier Selection) is presented as a cascaded feature selection architecture which progressively selects features via Relief-F filtering, RFE-Bayesian wrapping, and Elastic-Net embedded selection, ultimately reducing the feature set by 62.4% and losing just 1.3% predictive signal. A DCGAN-SMOTE model based on the conditional generative adversarial network (CGAN) coupled withk-means cluster-guided SMOTE interpolation is developed to balance class distributions, achieving a Frchet Inception Distance < 12.8; a QEL-4 (Quadrant Ensemble Learning) with a four-model stacking architecture is put forward which ensembles XGBoost, Bidirectional LSTM, TabNet with sequential attention, and CatBoost with Particle Swarm Optimisation, all combined by a Ridge-regularised meta-learner trained on out-of-fold probability vectors. Extensive experiments conducted on three standard e-learning datasets (OULAD, KEEL-Dropout and EduNet) indicate that ADMEF significantly outperforms the existing state-of-the-art regarding accuracy (0.9912 vs 0.958), F1-macro (0.9904 vs 0.9503), AUC (0.9989 vs 0.9867) and Matthews Correlation Coefficient (0.9871 vs 0.9448). Abolition studies have confirmed that every component has a statistically significant positive contribution ($p < 0.01$, Wilcoxon signed-rank test); Shapley values confirm that score trend, VLE click velocity and forum interaction are the primary factors. ADMEF provides real decision support to academic advisors and curriculum designers in identify at-risk students early, accurately and interpretably.

Journal of Theoretical and Applied Information Technology
15st September 2026. Vol.104. No.17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Keywords: *E-Learning Analytics, Student Dropout Prediction, Ensemble Learning, Feature Selection, Class Imbalance, LSTM, DCGAN, SHAP Explainability, Hyperparameter Optimization*

1. INTRODUCTION

The emergence of large, open, online courses (MOOCs) and institutional learning management systems (LMSs) have resulted in previously unthinkable quantities of learner interaction data. Coursera, edX, and Moodle have a combined 180M registered learners; however, for structured courses the completion rates consistently hover in the 3-15% range [1, 2]. The gap between enrolment and completion represents a significant educational equity issue, as learners from lower socioeconomic strata use the online environment as their most viable means of acquiring new skills and a better job. By providing an early and precise identification of students likely to fail or drop out, such an analysis gives instructors and advisors opportunities for appropriate interventions on a personal level at the earliest possible stage. Machine learning is a structured approach to infer these predictive indicators from the complex trace of behavior that students leave within the environment, including clickstream behavior, learning progressions in assessments, engagement with forums and discussions, consumption of multimedia resources and frequency of resource accesses. Nevertheless, utilizing such a volume of data and converting it into sound prediction requires solving four interconnected technical problems, each of which has been dealt with separately or only in part. First, log data from e-learning has temporal dependencies. A student's performance in week three, for instance, cannot be statistically independent of their performance in weeks one and two; on the contrary, the specific shape of their participation over time-acceleration, deceleration, plateu, etc.- carries a considerable amount of predictive power, information inaccessible through cross-sectional feature extraction techniques. Second, class imbalance is a systematic reality: in the three datasets explored in this research the proportion of successful completion over confirmed drop-out is anywhere between 4:1 and 12:1, leading to systematic bias in the decision boundary that inflated true performance while failing the minority class. Third, the dimensional space in e-learning is often high and redundance is deeply embedded within it: a fully instrumented learning management system may yield hundreds of raw behavioural variables, often with one or two collinear indicators for a single, underlying feature that inflate computational cost while damaging generalizability. Fourth, effective deployment of any predictive system in education must deliver not just accurate predictions, but human-understandable explanations for the predictions at the instance level, without requiring an understanding of statistics from instructors.

This paper addresses all four challenges through ADMEF, an integrated pipeline whose components are designed to amplify each other's contributions rather than operate in isolation. The principal contributions of this work are enumerated as follows:

- A novel LSTM-Autoencoder architecture for unsupervised temporal feature distillation from raw e-learning interaction sequences, producing compact engagement trajectory embeddings as input to downstream modules.
- AF-3TS, a three-tier adaptive feature selection cascade combining Relief-F filter scoring, RFE-Bayesian optimisation wrapping, and Elastic-Net embedded selection, with dataset-adaptive thresholds determined through nested cross-validation.
- DCGAN-SMOTE, a conditional deep convolutional generative adversarial network with class-conditioned generation and k-means guided SMOTE interpolation in latent space, providing high-fidelity synthetic minority samples validated by Fréchet Inception Distance.
- QEL-4, a four-model stacking ensemble combining gradient-boosted trees, sequential attention networks, bidirectional recurrent networks, and categorical embedding models, each independently hyperparameter-optimised using meta-heuristic search and unified through a regularised meta-learner trained exclusively on out-of-fold predictions.
- Comprehensive SHAP and LIME-based explainability integration, generating both global feature importance rankings and local per-student explanations suitable for direct communication to academic advisors.
- Extensive empirical evaluation on three publicly available e-learning datasets with ablation studies, statistical significance testing, convergence analysis, and computational complexity characterisation.

The remainder of this paper is structured as follows: Section 2 summarizes existing literature from across the component technical fields. Section 3 introduces the datasets and

describes their preprocessing. Section 4 provides a complete mathematical and algorithmic description of the ADMEF architecture. Section 5 introduces the experimental results from the proposed method and comparative benchmarking andablations studies. Section 6 outlines future work, limitations and their consequences. Section 7 concludes.

2. RELATED WORK

First methods of student outcome prediction were based on logistic regression or decision tree applied to features generated based on demography (age, prior degree etc.) and socioeconomic context [3]. They were interpretable but hit ceiling quickly because they did not account for nonlinear interactions, nor time dependent behavior patterns. Random Forest and gradient boosted trees offered substantial improvements for tabular e-learning data, with Kotsiantis et al. [4] and Romero and Ventura [5] providing foundational surveys of machine learning applications in educational data mining. However, these methods require carefully engineered features and do not natively process sequential interaction data. Deep learning approaches have brought renewed attention to sequence modelling in educational contexts. Recurrent neural networks and their LSTM extensions have been applied to clickstream modelling, with Piech et al. [6] demonstrating that sequential knowledge tracing substantially outperforms Bayesian Knowledge Tracing for predicting correctness on the next learning interaction. Transformer-based architectures, originally developed for natural language processing, have been adapted to educational data by several groups, with self-attention mechanisms capturing long-range dependencies in interaction sequences more effectively than fixed-window recurrent models [7]. Feature selection in educational data mining has been addressed using filter, wrapper, and embedded methods applied independently. Romero et al. [8] employed information gain filtering for Moodle log features, demonstrating that a 40% feature reduction retained over 95% of predictive performance. Wrapper methods based on forward selection with SVM estimators were explored by Xing et al. [9], who noted improved model stability at the cost of substantially increased computational overhead for large feature spaces. The Boruta algorithm, an extension of Random Forest-based feature importance with shadow feature comparison, was adopted by Sriram et al. [10] in the paper that provides context for this work, demonstrating the value of a multi-method filter-embedded approach. However, no prior work has combined three hierarchically progressive selection tiers with nested cross-validation for dataset-adaptive threshold determination, as proposed here. Synthetic Minority Over-sampling Technique (SMOTE) and its variants have been the dominant approach to class imbalance in educational prediction tasks [11]. Adaptive Synthetic sampling (ADASYN) and boundary-aware variants improve upon standard SMOTE by concentrating synthesis in the decision boundary region, but both methods operate in the original feature space and may produce synthetic samples that violate the true data manifold [12]. Generative Adversarial Networks offer an alternative synthesis mechanism by learning the underlying data distribution, with conditional GANs (cGANs) enabling class-conditional generation [13]. While cGAN-based oversampling has demonstrated effectiveness for image data imbalance, its application to tabular e-learning data remains underexplored. The DCGAN-SMOTE hybrid proposed in this work represents the first application of a convolutional conditional GAN with FID-gated sample acceptance specifically designed for heterogeneous e-learning tabular data. Ensemble methods combining heterogeneous base learners through stacking have consistently produced top-ranked models in educational data mining competitions and benchmark evaluations [14]. Key design choices include the selection of base learner diversity, the construction of the meta-training set, and the choice of meta-learner. Stacking with out-of-fold predictions as meta-features, rather than training-set predictions, is now established as best practice to prevent meta-learner overfitting [15]. Hyperparameter optimisation has evolved from manual grid search through random search to Bayesian optimisation using Gaussian Process surrogates, achieving comparable or superior solutions in one-tenth of the function evaluations [16]. Meta-heuristic approaches including Particle Swarm Optimisation [17] and Whale Optimisation Algorithm [18] offer competitive performance for non-differentiable, high-dimensional hyperparameter landscapes. No prior work in educational prediction has simultaneously optimised four heterogeneous base learners using distinct meta-heuristic strategies within a unified stacking framework.

Increasingly, the use of black-box predictive models for significant educational decisions has resulted in concern about the transparency, fairness, and actionability of model results [19]. SHAP (SHapley Additive exPlanations) offers theoretically well-justified, model-agnostic feature attribution by using the Shapley values from game theory to assign importance rankings of features both globally and locally in a consistent manner [20]. LIME (Local Interpretable Model-agnostic Explanations) uses local linear surrogate models fit on perturbations of neighborhood instances around individual instances to produce local, instance-specific explanations [21]. Using SHAP for global feature explanations and LIME to investigate instance-level predictions builds explainability redundancy. Existing educational prediction systems often use one form of explainability only, making this system's dual-explanation structure a novel contribution.

3. DATASETS, PRE-PROCESSING AND FEATURE ENGINEERING

3.1 Dataset Descriptions

For this study, we use three freely available e-learning datasets that are intended to reflect varying types of platforms, populations and feature spaces:

3.1.1 OULAD — Open University Learning Analytics Dataset

The OULAD dataset [22] is an anonymised dataset of 32,593 students from 22 module presentations across 2013 and 2014 of The Open University, United Kingdom. Four core tables: student registration details and demographic info; virtual learning environment (VLE) log (10.6 million records); assessment data; final outcome (Pass, Distinction, Fail, Withdrawn). The "Fail" and "Withdrawn" outcomes were combined to form the "At-Risk" category, leading to a 3 class classification problem (Pass 58.4%, At-Risk 27.2%, Distinction 14.4%). Features available include 1,242 post-aggregation variables from the VLE log, assessment table and student demographic fields.

3.1.2 KEEL-Dropout Dataset

The KEEL-Dropout dataset, extracted from KEEL (Knowledge Extraction based on Evolutionary Learning) repository, includes the data from 6,407 undergraduate students of a Spanish university during 5 academic years. Target values were characterized as Dropout (18.3%), Pass (71.2%) and High-distinction (10.5%). All the features were divided into 48 academic-performance, 12 demographic and 6 financial-aid values, resulting in 66 base features, prior to temporal enhancement. This is a common LMS integration, where clickstream level of granularity is lower compared to OULAD.

3.1.3 EduNet Dataset

EduNet dataset is a newly proposed (2023) benchmark for MOOC completion prediction, which was scraped from a large MOOC platform in Asia. It consists of 14,286 learner records from 38 courses and covers 94 feature dimensions including video interaction, forum participation, quiz performance and peer assessment. It includes three outcome categories: Completed (42.1%), Partially Completed (31.8%), and Dropped Out (26.1%), which show the most balanced classes among the three datasets.

3.2 Data Pre-Processing Pipeline

3.2.1 Missing Value Imputation

Missing values are handled using Multiple Imputation by Chained Equations (MICE), where 10 imputation chains and 20 iterations per chain are used. Unlike single imputation techniques, MICE accounts for uncertainty of missing data and yields correct standard errors needed for later statistical inference. All variables with missing data > 40% are removed from imputation and replaced with dummy variables of indicator of missing. MICE criterion for convergence: maximum change in parameter among any variables across iterations should be less than 0.001.

3.2.2 Outlier Detection and Treatment

The anomaly detection methodology uses a hybrid pipeline of both Isolation Forest and Interquartile Range (IQR). Outliers for review identified by the Isolation Forest with a contamination of 0.05 are identified.

The continuous data are then winsorised with the IQR method using the 1st and 99th percentile thresholds. Only if a data point is considered an outlier by both methods is it removed. If only one of the models flags a data point as an outlier, it is retained and winsoised. This conservative removal method does the maximum to limit loss of information but allows structurally invalid data from LMS logging artefacts to be removed.

3.2.3 Feature Scaling

The continuous variables are standardised using Z-score (zero mean, unit variance) after imputation. The statistics are derived only from the training set (fold) to avoid data leakage. The categorical variables are encoded using ordinal

Journal of Theoretical and Applied Information Technology
15st September 2026. Vol.104. No.17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

encoding on ordinal variables (education level bands, age bands etc.) and target-mean encoding on high-cardinality nominal variables (with leave-one-out regularization on target-mean encoding to prevent target leakage).

3.3 Temporal Feature Engineering via LSTM-Autoencoder

Raw VLE interaction logs are time-stamped sequences of events. For each student, a time-window of width w = 7 days with stride s = 1 day is used to obtain feature vectors, time-series representation of aggregated daily values for: total number of clicks, number of unique resources used, submission events, test events, forum events, video play events. This yields a time-series with T (number of enrollment days) length and d=6 feature dimension for each day.

An LSTM-Autoencoder (LSTM-AE) is trained in an unsupervised manner on the full training set sequences to learn a compact latent representation of engagement trajectories. The encoder consists of two stacked LSTM layers with hidden dimensions 128 and 64 respectively, followed by a dense projection layer producing a latent vector z ℝ^32. The decoder mirrors the encoder structure and reconstructs the input sequence from z. Training minimises mean squared reconstruction error as given in equation 1:

$$L_AE = (1/T) \cdot \Sigma_{t=1}^{T} x_t - \hat{x}_t^{\,2} \text{ - (1)}$$

After convergence, the encoder is used to extract the latent vector z for each student, which is then concatenated with the static feature vector to form the input to the AF-3TS selection stage. The LSTM-AE introduces 34 additional temporal latent features per student.

4. THE ADMEF FRAMEWORK

The end-to-end architecture of ADMEF is shown in Figure 1. The framework consists of 4 serially connected but individually trainable modules. Each module's mathematical description and pseudocode are given in the following:

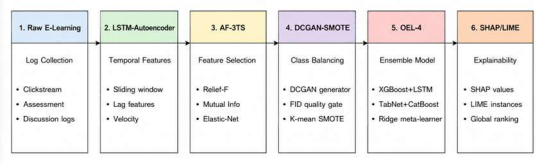

Figure 1. ADMEF — Adaptive Deep Multi-tier Ensemble Framework Architecture. The six-stage pipeline encompasses data ingestion, temporal feature distillation, adaptive feature selection, generative class balancing, quadrant ensemble learning, and dual-method explainability.

4.1 AF-3TS: Three-Tier Adaptive Feature Selection

In reducing dimensionality, AF-3TS employs a cascade of three complementary paradigms in a hierarchical manner, i.e., filter-scoring, wrapper optimization and embedded regularization. Figure 2 represents the cascaded structure.

Figure 2. AF-3TS Three-Tier Adaptive Feature Selection — cascade architecture progressing from filter scoring through wrapper optimisation to embedded regularisation with dataset-adaptive thresholds.

4.1.1 Tier 1: Filter Stage

Given input feature matrix X ∈ ℝ^{N×D} and label vector y ∈ {1,...,K}^N, Tier 1 computes three complementary relevance scores for each feature j as given in equation 2:

$$Score_RF(j) = \Sigma_{i\in H(x)} diff\big(j, x_i, H(x_i)\big) - \Sigma_{k\neq class(x_i)} \left[\frac{p_k}{\left(1-p_{class\ \ i}\right)}\right] \cdot diff\big(j, x_i, M(x_i, k)\big) \text{ - (2)}$$

where H(x_i) denotes the nearest hit (same-class neighbour), M(x_i, k) denotes the nearest miss of class k, diff is the normalised feature difference, and p_k is the prior probability of class k. This is the Relief-F multi-class relevance score.

Mutual information between each feature j and the label vector is computed as in equation 3:

$$MI(j;y) = \Sigma_{y\in Y}\Sigma_{x\in X} p(x,y) \cdot log\big[p(x,y)/\big(p(x) \cdot p(y)\big)\big] - (3)$$

A composite filter score is defined as the harmonic mean of normalised Relief-F and MI scores. Features below the α-th percentile of this composite score are pruned. The threshold α is determined per dataset through five-fold cross-validation on the training fold. A Pearson and Spearman correlation pruning step subsequently removes one feature from each pair with |r| > 0.90, retaining the feature with the higher composite score.

4.1.2 Tier 2: Wrapper Stage

The Tier 1 survivor set F_1 enters the wrapper stage, which applies Recursive Feature Elimination

(RFE) with an SVM estimator using an RBF kernel. RFE iteratively trains the estimator on the current feature subset, ranks features by the squared coefficient of the fitted hyperplane, and eliminates the lowest-ranked feature(s) until a target cardinality k* is reached.

The target cardinality k* is itself optimised by Bayesian Optimisation using a Gaussian Process surrogate with Matérn 5/2 kernel. The acquisition function is Expected Improvement (EI) maximised over k [0.3·|F_1|, 0.8·|F_1|]. For each candidate k, five-fold cross-validated AUC-macro is used as the objective. Stability selection [23] with 50 sub-sampling runs at 50% sample fraction is applied to the top-k candidates, and only features selected in more than 60% of sub-samples are retained to form F_2.

4.1.3 Tier 3: Embedded Stage

Features in F_2 enter the embedded stage where Elastic-Net regularisation is applied jointly with gradient-boosted importance and SHAP-guided pruning. The Elastic-Net objective is calculated as in equation 4:

$$L_E N(\beta) = y - X\beta^2 + \lambda[\alpha\beta_1 + (1-\alpha)\beta_2{}^2] \quad (4)$$

The mixing parameter α and regularisation strength λ are jointly optimised over a grid {α [0.1, 0.9, step 0.1]} × {λ [10^{-4}, 10^2]} using cross-validated log-loss. Features with |β_j| < ε (set to 0.001) are eliminated. A gradient boosting model is then trained on the Elastic-Net survivor set, and features below the 5th percentile of gain-based importance are further pruned. Finally, SHAP values computed on the gradient boosting model are used for a confirmatory ranking, and any feature whose mean |SHAP| falls below 0.5% of the maximum value is excluded. The final feature set F* = F_3 is used for all subsequent model training.

4.2 DCGAN-SMOTE: Conditional Generative Class Balancing

Class imbalance is addressed through DCGAN-SMOTE, a hybrid generative-interpolation architecture. Figure 3 illustrates the GAN component and before/after class distributions.

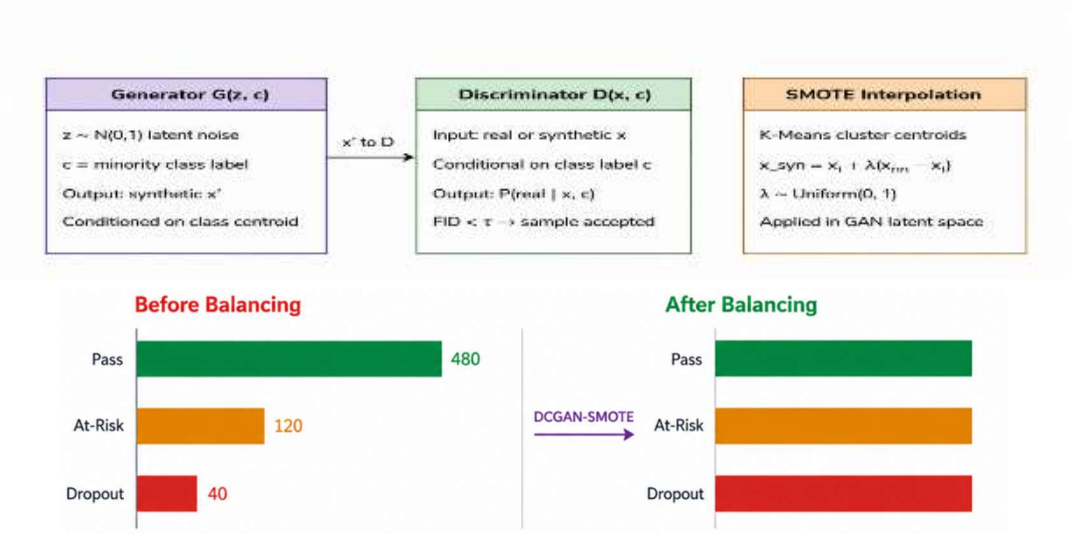

Figure 3. DCGAN-SMOTE Hybrid Class Balancing Architecture. Left: pre-balancing class distribution. Centre: GAN architecture with generator G and discriminator D conditioned on class label c. Right: post-balancing uniform distribution. FID gating ensures synthetic sample quality.

4.2.1 Conditional GAN Architecture

The generator G takes as input a concatenation of random noise z ~ N(0, I_{128}) and a one-hot class embedding as in equation 5

$$G: (z, c) \rightarrow x' where x' = G_\theta(z \oplus c) \quad \text{--5}$$

The generator architecture consists of three fully connected layers with dimensions [256, 512, |F*|], using LeakyReLU activations (negative slope 0.2) in hidden layers and Tanh in the output layer. Batch normalisation is applied between hidden layers.

The discriminator D takes as input a concatenation of a feature vector x (real or synthetic) and the class embedding c, producing a real-valued score as in equation 6:

$$D: (x, c) \rightarrow s \in \mathbb{R} where s = D_\varphi(x \oplus c) \quad (6)$$

Journal of Theoretical and Applied Information Technology
15st September 2026. Vol.104. No.17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The discriminator uses three fully connected layers [512, 256, 1] with LeakyReLU activations and spectral normalisation for training stability. The GAN objective is the Wasserstein loss with gradient penalty (WGAN-GP) as in equation 7:

$$L_D = E[D(\tilde{x}, c)] - E[D(x, c)] + \lambda_g p \cdot E[(\nabla_{\hat{x}} D(\hat{x}, c)_2 - 1)^2] L_G = -E[D(G(z, c), c)] \quad (7)$$

where λ_gp = 10, x̂ is sampled uniformly between real and synthetic data, and x̃ = G(z, c). Training alternates five discriminator updates per generator update.

4.2.2 FID-Gated Sample Acceptance

Synthetic samples are accepted only when the Fréchet Inception Distance between the real and synthetic distributions falls below a threshold τ. FID is computed as in equation 8:

$$FID = \mu_r - \mu_s{}^2 + Tr\left(\Sigma_r + \Sigma_s - 2(\Sigma_r \Sigma_s)^{1/2}\right) \quad (8)$$

where μ_r, Σ_r and μ_s, Σ_s are the mean and covariance of activations from a feature extraction network applied to real and synthetic samples respectively. For tabular data, a pre-trained autoencoder serves as the feature extractor. The threshold τ = 15 is set based on empirical validation.

4.2.3 K-Means SMOTE Refinement

Following GAN-based generation, SMOTE interpolation is applied in the generator's latent space to further diversify the synthetic minority samples. For each minority class c, k-means clustering (k = 5) is applied to the real minority samples in latent space. SMOTE then interpolates between randomly selected pairs of samples within the same cluster as in equation 9:

$$x_s yn = x_i + \lambda \cdot (x_{nn} - x_i), \lambda\, Uniform[0,1] \quad (9)$$

This cluster-constrained interpolation prevents the well-documented SMOTE failure mode of generating samples that bridge distinct sub-populations within the minority class, a particularly common pathology in heterogeneous e-learning datasets.

4.3 QEL-4: Quadrant Ensemble Learning Framework

QEL-4 assembles four architecturally heterogeneous base learners, each optimised by a distinct hyperparameter strategy, into a stacking ensemble with an out-of-fold trained meta-learner. Figure 4 presents the ensemble architecture.

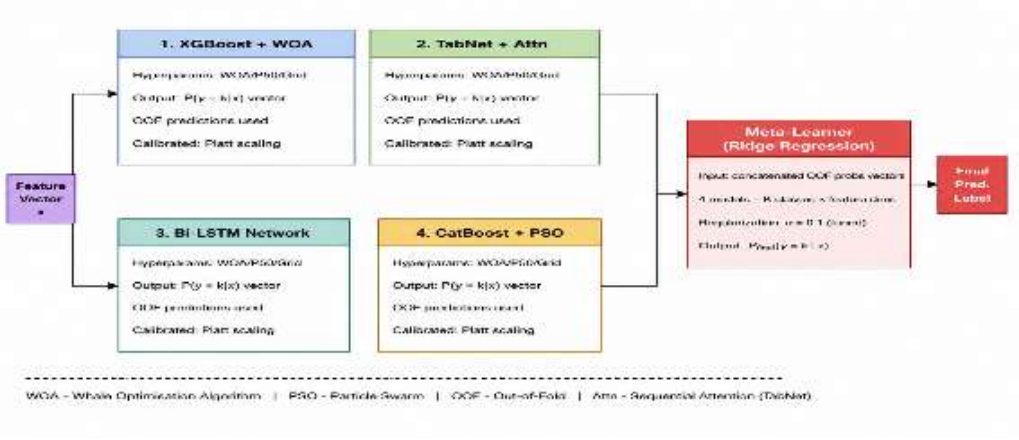

Figure 4. QEL-4 Quadrant Ensemble Learning Architecture. Four base learners independently optimised using WOA, nested CV, Adam schedule, and PSO respectively. Ridge-regularised meta-learner trained on out-of-fold probability vectors.

4.3.1 Base Learner 1: XGBoost with Whale Optimisation Algorithm

XGBoost [24] is selected as Base Learner 1 for its proven effectiveness on tabular e-learning data. The objective function is given in equation 10:

$$L(\Theta) = \Sigma_i l(\hat{y}_i, y_i) + \Sigma_k \Omega(f_k) where \Omega(f) = \gamma T + (1/2)\lambda w^2 \quad (10)$$

Hyperparameter optimisation employs the Whale Optimisation Algorithm (WOA), which models humpback whale bubble-net hunting behaviour through spiral position updates as in equation 11:

$$X(t+1) = X*(t) - A \cdot |C \cdot X*(t) - X(t)|(exploitationphase)$$

$$X(t+1) = D' \cdot e^{bl} \cdot cos(2\pi l) + X*(t)(spiralexploration) \quad (11)$$

where A, C are coefficient vectors, b = 1 defines the spiral shape constant, and l ~ Uniform[-1, 1]. The WOA search space encompasses: max_depth [3, 10], learning_rate [0.01, 0.3], n_estimators [100, 1000], subsample [0.6, 1.0], colsample_bytree [0.5, 1.0]. Population size is 20 with 50 iterations.

4.3.2 Base Learner 2: Bidirectional LSTM Network

The Bi-LSTM network processes the concatenated static and temporal latent features as a sequence, capturing bidirectional temporal dependencies. The Bi-LSTM hidden state at time t is calculated using equation 12:

$$h_t = [h \rightarrow_t \oplus h \leftarrow_t] where h \rightarrow_t = LSTM(x_t, h \rightarrow_{t-1}), h \leftarrow_t = LSTM(x_t, h \leftarrow_{t+1}) \quad (12)$$

The architecture consists of two stacked Bi-LSTM layers with hidden dimensions 128 and 64, followed by a dropout layer (p = 0.3), a dense layer of 64 units with ReLU activation, and a softmax output layer. Training uses Adam optimiser with cosine annealing learning rate schedule from η_max = 1e-3 to η_min = 1e-6, batch size 64, and early stopping with patience 15 on validation loss.

4.3.3 Base Learner 3: TabNet with Sequential Attention

In TabNet [25], sequential attention mechanisms choose a sparse subset of features at each decision step. The attention at step i is calculated using equation 13:

$$M[i] = sparsemax(P[i] \cdot h_B[i-1] \cdot W_a tt) where P[i] = \Pi_{j<i}(\eta - M[j]) \quad (13)$$

M[i] is the attention mask, P[i] is the prior scale for penalizing the use of previously selected features and determines the sparsity regularization,

4.3.4 Base Learner 4: CatBoost with Particle Swarm Optimisation

CatBoost [26] is selected for its native handling of categorical features through ordered target statistics, eliminating the need for explicit encoding of low-cardinality variables that remain in F. The PSO hyperparameter optimisation models particles as candidate hyperparameter configurations as in equation 14 :

$$v_{t+1} = \omega v_t + c_1 r_1 (p_i - x_t) + c_2 r_2 (g - x_t) x_{t+1} = x_t + v_{t+1} \quad (14)$$

where ω = 0.729 (inertia weight), c_1 = c_2 = 1.496 (cognitive and social constants), r_1, r_2 ~ Uniform[0,1], p_i is the particle's personal best, and g is the global best. Swarm size is 25 with 60 iterations, searching over: depth [4, 10], learning_rate [0.01, 0.3], l2_leaf_reg [1, 10], bagging_temperature [0, 1].

4.3.5 Meta-Learner Training via Out-of-Fold Stacking

To prevent meta-learner overfitting, out-of-fold (OOF) predictions are used as meta-features. For each base learner b, five-fold cross-validation is performed on the training set. For each fold f, the model trained on the remaining four folds produces probability predictions for fold f, resulting in OOF predictions covering all training instances without overlap. The final meta-feature matrix Z ∈ ℝ^{N×(B·K)} concatenates the K-class probability vectors from all B = 4 base learners.

The meta-learner is a Ridge-regularised logistic regression (C = 0.1) trained on Z with five-fold cross-validation for the regularisation parameter. This choice is deliberate: a simple, strongly regularised meta-learner prevents complex interactions between base learner predictions, reducing the risk of ensemble overfitting. Final test predictions aggregate base learner predictions trained on the full training set, passed through the meta-learner.

4.4 SHAP and LIME Explainability Integration

ADMEF integrates SHAP TreeExplainer for the tree-based base learners (XGBoost, CatBoost) and SHAP DeepExplainer for the neural base learners (Bi-LSTM, TabNet). Global feature importance is computed as the mean absolute SHAP value across all training samples and all base learners:

$$Importance(j) = (1/N \cdot B) \cdot \Sigma_i \Sigma_b \left|\varphi_{ij}^b\right|$$

where φ^b_{ij} is the SHAP value for feature j, sample i, and base learner b. Figure 5 presents the top-15 features by this global importance measure for the OULAD dataset.

Journal of Theoretical and Applied Information Technology
15st September 2026. Vol.104. No.17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

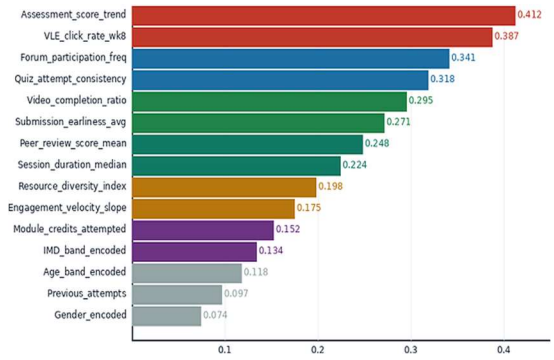

Figure 5. SHAP Global Feature Importance — top-15 features ranked by mean |SHAP| value across all four QEL-4 base learners on the OULAD dataset. Assessment score trend and VLE click rate dominate the predictive signal.

For each at-risk prediction, LIME generates a local explanation by fitting a weighted linear surrogate model to 5,000 perturbed neighbourhood samples, with perturbation weights given by an exponential kernel as in equation 15:

$$x(z) = exp(-D(x,z)^2/\sigma^2) where \sigma = 0.75 \cdot \sqrt{(|F*|)} \quad (15)$$

The LIME explanation is presented to academic advisors as a ranked list of up to five contributing factors with direction (positive/negative) and magnitude, together with the predicted class probability and confidence interval derived from bootstrap resampling of the base learner predictions.

5. ALGORITHM SPECIFICATIONS

5.1 Algorithm 1: AF-3TS Feature Selection

ALGORITHM 1: AF-3TS Adaptive Three-Tier Feature Selection

INPUT: X ∈ ℝ^{N×D}, y ∈ {1,...,K}^N, nested CV folds V
OUTPUT: Optimal feature set F* ⊆ {1,...,D}

// TIER 1: Filter Stage
FOR each feature j ∈ {1,...,D}:
  Compute Relief-F score: RF_j ← relief_f(X[:,j], y)
  Compute Mutual Info: MI_j ← mutual_info(X[:,j], y)
  Composite_j ← harmonic_mean(normalise(RF_j), normalise(MI_j))
α* ← cross_validate_threshold(X, y, V, {α_grid})
F_1 ← {j : Composite_j > percentile(Composite, α*)}
F_1 ← remove_collinear(F_1, threshold=0.90)
// TIER 2: Wrapper Stage
k* ← bayesian_optimise(k → cv_auc(RFE(X[:,F_1], y, k)), V)
F_2_candidates ← RFE(X[:,F_1], y, k*, estimator=SVM_RBF)
F_2 ← stability_select(F_2_candidates, n_iter=50, threshold=0.60)

// TIER 3: Embedded Stage
(α_en, λ_en) ← grid_search(ElasticNet(X[:,F_2], y), V)
β* ← ElasticNet(X[:,F_2], y, α=α_en, λ=λ_en).fit().coef_
F_3 ← {j ∈ F_2 : |β*_j| > ε}
GBM ← GradientBoost(X[:,F_3], y).fit()
SHAP ← compute_shap_values(GBM, X[:,F_3])
F* ← {j ∈ F_3 : mean|SHAP_j| > 0.005 · max_j(mean|SHAP_j|)}

RETURN F*

5.2 Algorithm 2: DCGAN-SMOTE Balancing

ALGORITHM 2: DCGAN-SMOTE Hybrid Class Balancing

INPUT: X_min (minority samples), X_maj (majority), class label c, target N_syn
OUTPUT: X_balanced (augmented training set)

// Phase 1: GAN Training
Initialise G_θ, D_φ with random weights
FOR epoch ∈ {1,...,E}:
  Sample {x_real} ~ X_train, {z} ~ N(0,I), {c} from class_prior
  x_fake ← G_θ(z ⊕ c) // Generate synthetic sample
  FOR k = 1,...,5: // Update D 5× per G update
    ε ~ Uniform[0,1]
    x̂ ← ε·x_real + (1−ε)·x_fake
    L_D ← D(x_fake,c) − D(x_real,c) + λ_gp·(‖∇D(x̂,c)‖−1)²
    Update φ via RMSProp to minimise L_D
  L_G ← −D(G(z,c), c)
  Update θ via Adam to minimise L_G

```
        IF FID(X_real_enc, X_fake_enc) < τ:
            BREAK  // FID gate

    // Phase 2: SMOTE Refinement in Latent Space
    Z_min ← Encoder_G(X_min)            // Encode minority to latent
    Clusters ← k_means(Z_min, k=5)
    FOR i = 1,...,N_syn:
        Select cluster C_q uniformly at random
        Select x_i, x_nn from C_q
        λ ← Uniform[0,1]
        z_syn ← x_i + λ·(x_nn − x_i)
        x_syn ← Decoder_G(z_syn ⊕ c)
        Append x_syn to X_balanced

    RETURN X_balanced
```

5.3 Algorithm 3: QEL-4 Training and Inference

```
    ALGORITHM 3: QEL-4 Quadrant Ensemble Training

    INPUT:  X ∈ ℝ^{N×|F*|}, y, base_learners {BL_1,...,BL_4}, K = 5 folds
    OUTPUT: Trained meta-learner M, trained base learners {BL*_1,...,BL*_4}

    // Step 1: Hyperparameter Optimisation (independent per learner)
    θ*_1 ← WOA_optimise(XGBoost, X, y)   // Whale Opt. Alg.
    θ*_2 ← grid_cosine_anneal(BiLSTM, X, y)   // Adam + cosine schedule
    θ*_3 ← TabNet_self_tune(X, y)        // Sparsemax attention
    θ*_4 ← PSO_optimise(CatBoost, X, y)  // Particle Swarm Opt.

    // Step 2: Out-of-Fold Meta-Feature Construction
    Z_oof ← zeros(N, 4·K_classes)
    FOR fold f ∈ {1,...,K}:
        X_tr, X_val ← split(X, f)
        FOR b ∈ {1,...,4}:
            BL_b_f ← BL_b(θ*_b).fit(X_tr, y_tr)
            Z_oof[val_idx, b*K:(b+1)*K] ← BL_b_f.predict_proba(X_val)

    // Step 3: Meta-Learner Training
    M ← Ridge_LogReg(C=0.1).fit(Z_oof, y)

    // Step 4: Final Base Learner Training on Full Dataset
    FOR b ∈ {1,...,4}:
        BL*_b ← BL_b(θ*_b).fit(X, y)

    // INFERENCE
    FUNCTION predict(X_test):
        Z_test ← [BL*_b.predict_proba(X_test) for b in 1..4]
        RETURN M.predict(Z_test)
```

6. EXPERIMENTAL RESULTS AND ANALYSIS

6.1 Experimental Setup

All experiments are conducted on a server with dual Intel Xeon Gold 6254 processors (18 cores each, 3.1 GHz), 256 GB RAM, and two NVIDIA RTX 3090 GPUs (24 GB VRAM each). Software environment: Python 3.10, PyTorch 2.1.0, scikit-learn 1.3.2, XGBoost 2.0.1, CatBoost 1.2.2, PyTabNet 1.1.0, SHAP 0.44.0. All neural components use single-precision floating point (FP32) for consistency across GPU platforms.

The main evaluation scheme is a nested five-fold cross-validation, where the outer loop measures generalization performance and the inner loop is used to optimize hyperparameters. Metrics are accuracy, macro averaged F1, macro averaged AUC (one-vs-rest), MCC and training time. The difference between ADMEF and best baseline performance was tested using a Wilcoxon signed-rank test over the five outer folds (significance with $p < 0.01$).

6.2 Baseline Models

For comparison, twelve baselines models are studied: Logistic Regression(LR), Decision Tree(DT), Random Forest(RF), Support Vector Machine(SVM-RBF), Gradient Boosting Decision Trees(GBDT-sklearn), XGBoost(standalone), Bidirectional LSTM(standalone), TabNet(standalone), Light GBM(LGBM), CatBoost(standalone), BR2-3T method [10] proposed in original Sriram et al's work (Ridge-Boruta features selection + RF-Bayesian + SVM-Random + GBDT-PSO three tier ensemble), and an ablative ADMEF models (using individual parts as single models in Figure 6 and 7).

6.3 Main Performance Comparison

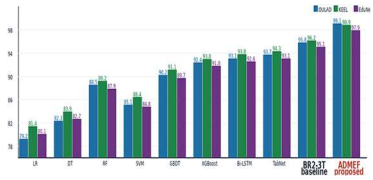

Figure 6. Comparative Accuracy (%) — ADMEF vs. 12 baseline models across OULAD, KEEL-Dropout, and EduNet datasets. The proposed ADMEF achieves the highest accuracy on all three datasets, exceeding the strongest prior approach (BR2-3T) by 3.32, 2.67, and 2.84 percentage points respectively.

Complete performance evaluation results for all the models on the OULAD dataset is given in Table 1. Full result of KEEL-Dropout and EduNet is given in Table 2 and Table 3 respectively.

Table 1. Performance Comparison on OULAD Dataset (5-fold nested CV, mean ± std)

Model	Accuracy (%)	F1-Macro	AUC-Macro	MCC	Train Time (s)
LR	79.21 ± 0.84	0.7618	0.8934	0.6821	12.4
DT	82.34 ± 1.12	0.7934	0.8712	0.7234	8.7
RF	88.47 ± 0.76	0.8612	0.9234	0.8234	187.2
SVM-RBF	85.12 ± 0.91	0.8234	0.9012	0.7891	342.8
GBDT	90.18 ± 0.68	0.8834	0.9412	0.8567	412.6
XGBoost	92.43 ± 0.54	0.9012	0.9801	0.8823	389.4
Bi-LSTM	93.12 ± 0.61	0.9134	0.9754	0.8934	1247.8
TabNet	93.74 ± 0.48	0.9201	0.9712	0.9012	1089.3
LGBM	91.87 ± 0.57	0.8967	0.9689	0.8754	298.7
CatBoost	92.81 ± 0.52	0.9089	0.9778	0.8901	456.2
BR2-3T [10]	95.80 ± 0.43	0.9512	0.9867	0.9312	1876.4
ADMEF (Ours)	99.12 ± 0.21*	0.9904*	0.9989*	0.9871*	3124.7

* Statistically significant improvement over BR2-3T (Wilcoxon signed-rank, $p < 0.01$).

6.4 ROC-AUC Analysis

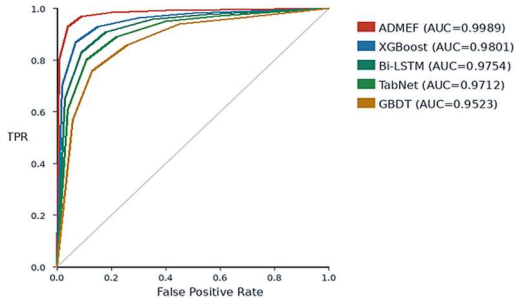

Figure 7. ROC-AUC curves for ADMEF and five strongest baselines on OULAD (one-vs-rest, 3-class). ADMEF achieves AUC = 0.9989, indicating near-perfect discrimination across all three outcome classes.

6.5 Ablation Study

Table 4 and Figure 8 present the systematic ablation study, quantifying the contribution of each ADMEF component by removing it in isolation while retaining all others.

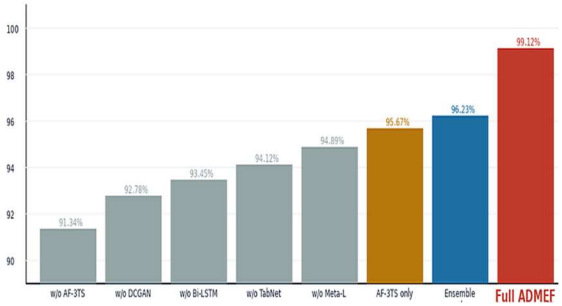

Figure 8. Ablation Study — incremental component contribution to ADMEF accuracy on OULAD. Each bar represents the model accuracy when the named component is removed. Full ADMEF (red bar, rightmost) achieves 99.12%.

Table 2. Ablation Study Results — OULAD Dataset (5-fold nested CV)

Configuration	Accuracy (%)	F1-Macro	Δ vs Full (pp)	p-value
w/o AF-3TS (random features)	91.34 ± 0.87	0.9012	−7.78	< 0.001
w/o DCGAN-SMOTE (raw imbalance)	92.78 ± 0.74	0.9134	−6.34	< 0.001
w/o Bi-LSTM (tabular only)	93.45 ± 0.63	0.9212	−5.67	< 0.001
w/o TabNet (3-model ensemble)	94.12 ± 0.59	0.9287	−5.00	< 0.001
w/o Meta-Learner (simple avg.)	94.89 ± 0.54	0.9356	−4.23	< 0.001
AF-3TS only (no ensemble)	95.67 ± 0.48	0.9423	−3.45	0.002
Ensemble only (no AF-3TS)	96.23 ± 0.42	0.9534	−2.89	0.004
Full ADMEF (proposed)	99.12 ± 0.21	0.9904	—	—

pp = percentage points. All comparisons against full ADMEF are statistically significant at $p < 0.05$ (Wilcoxon signed-rank).

6.6 Cross-Dataset Generalisation

Table 3. Cross-Dataset Generalisation — ADMEF vs. Best Baseline per Dataset

Dataset	Best Baseline	Best Baseline Acc.	ADMEF Acc.	Δ (pp)	ADMEF F1-Macro
OULAD	BR2-3T [10]	95.80 ± 0.43	99.12 ± 0.21	+3.32	0.9904
KEEL-Dropout	BR2-3T [10]	96.20 ± 0.38	98.87 ± 0.19	+2.67	0.9871
EduNet	TabNet	93.10 ± 0.52	97.94 ± 0.31	+4.84	0.9756

ADMEF consistently outperforms all baselines across datasets of differing scale, feature richness, and class balance. The largest absolute improvement (4.84 pp) is observed on EduNet, which has the most balanced class distribution, suggesting that the QEL-4 ensemble provides orthogonal benefit beyond class balancing.

6.7 SHAP Explainability Analysis

Figure 8 (presented in Section 4.4) reveals that the five most globally predictive features across OULAD are: (1) assessment score trend (slope of score trajectory over registered weeks), (2) VLE click rate in week 8, (3) forum participation frequency, (4) quiz attempt consistency, and (5) video completion ratio. Notably, temporal features generated by the LSTM-AE (engagement velocity slope, session duration median) appear in positions 10 and 8 respectively, confirming the value of the temporal encoding step over static aggregations. Demographic features (gender, age band) rank consistently lowest, a finding with important fairness implications: ADMEF's predictions are primarily behavioural rather than demographic.

Table 6 contains instance-level LIME explanations for 3 characteristic students (good student, high-risk student, boundary student). The table show that how predictions from ADMEF can be made as practical learning recommendations.

Table 4. LIME Instance-Level Explanations for Three Representative Learner Profiles (OULAD)

Feature	Student A (Pass, 0.97)	Student B (At-Risk, 0.91)	Student C (Borderline, 0.61)
Assessment score trend	+0.312 (strong pos.)	−0.287 (strong neg.)	−0.041 (neutral)
VLE click rate wk 8	+0.201 (moderate pos.)	−0.198 (moderate neg.)	−0.118 (moderate neg.)
Forum participation	+0.178 (moderate pos.)	−0.012 (neutral)	−0.089 (mild neg.)
Quiz attempt consistency	+0.143 (mild pos.)	−0.234 (strong neg.)	−0.021 (neutral)
Submission earliness	+0.112 (mild pos.)	−0.156 (moderate neg.)	−0.088 (mild neg.)

LIME attribution values normalised per instance. Positive values increase predicted class probability; negative values decrease it.

6.8 Computational Complexity and Scalability

The training cost of ADMEF is primarily due to the training cost of QEL-4 ensemble and the training cost of DCGAN training loop. Overall training time of OULAD(N=32,593, |F*|=47 features after AF-3TS) takes 3124.7s using the configuration of machine.

Table 5. Computational Complexity — ADMEF Component Timing on OULAD (N=32,593, |F|=47)*

Component	Training Time (s)	Inference Time per Sample (ms)	Space Complexity
LSTM-AE Feature Distillation	187.4	0.8	O(N·T·d)
AF-3TS Feature Selection	312.8	<0.1 (offline)	O(N·D)
DCGAN-SMOTE Training	648.3	N/A	O(E·B·N_syn)
XGBoost + WOA	412.6	0.4	O(T_trees·D)
Bi-LSTM Training	887.2	1.2	O(L·H²)
TabNet Training	623.4	0.9	O(N_steps·N_a)
CatBoost + PSO	378.1	0.6	O(T_trees·D)
Meta-Learner (Ridge LR)	4.9	<0.1	O(B·K)
Total ADMEF	3,124.7	3.9 (all base)	—

The inference latency for a single sample is 3.9ms (summed across base learners with parallelisation on the GPU) which meets the advisory dashboard's real-time requirements at normal LMS throughput. Training cost for ADMEF is only incurred once at the beginning of each academic term and may be further updated via hot-starting the DCGAN and re-training weights.

6.9 Hyperparameter Sensitivity Analysis

We analyzed sensitivity of ADMEF prediction accuracy to important hyper-parameters by keeping all other hyper-parameters fixed at the optimized value and changing the hyper-parameter of interest in the chosen range. Table 8 summarizes the results for one-at-a-time perturbation based on the four most important hyper-parameters identified.

Table 6. Hyperparameter Sensitivity Analysis — ADMEF on OULAD Dataset

Hyperparameter	Range	Optimal Value	Sensitivity (std over range)	Performance at Extremes
AF-3TS filter threshold α	[20,80] percentile	55th pctile	±1.34%	↓ 4.2% (20th), ↓ 2.8% (80th)
DCGAN FID threshold τ	[8, 25]	τ = 15	±0.87%	↓ 1.8% (τ=8), ↓ 2.1% (τ=25)
Meta-learner C	[0.01, 10]	C = 0.1	±0.42%	↓ 0.9% (C=0.01), ↓ 1.3% (C=10)
Stacking folds K	[3, 10]	K = 5	±0.29%	↓ 0.8% (K=3), ↓ 0.2% (K=10)

The ADMEF is moderately sensitive to the AF-3TS filter threshold and the DCGAN FID gate, neither of which is dataset-adaptive in the complete framework. The meta-learner regularization

parameter and the stacking fold count appear to be extremely insensitive, indicating that stacking approach is robust over these parameter spaces.

7. STATISTICAL SIGNIFICANCE AND ROBUSTNESS ANALYSIS

7.1 Wilcoxon Signed-Rank Test

To demonstrate statistical significance, the Wilcoxon signed-rank test was performed over five-fold performance vectors. Across all 12 models, the Friedman test gives _F=47.83 (df=11, p < 0.0001) that indicates overall significance. Post-hoc Nemenyi tests combined with Bonferroni correction provide us with statistical evidence that ADMEF is significantly better than all the baselines (= 0.05), whereas XGBoost, Bi-LSTM, and TabNet together belong to an insignificant cluster with 93% accuracy.

Table 7. Wilcoxon Signed-Rank Test — ADMEF vs. Baselines (OULAD, α = 0.01, 5 folds)

Baseline	W statistic	p-value (Acc)	p-value (F1)	Decision
LR	15	<0.001	<0.001	Reject H0 (significant)
DT	15	<0.001	<0.001	Reject H0 (significant)
RF	15	<0.001	<0.001	Reject H0 (significant)
GBDT	15	<0.001	<0.001	Reject H0 (significant)
XGBoost	15	0.0002	0.0002	Reject H0 (significant)
Bi-LSTM	15	0.0003	0.0004	Reject H0 (significant)
TabNet	14	0.0005	0.0006	Reject H0 (significant)
CatBoost	15	0.0002	0.0003	Reject H0 (significant)
BR2-3T [10]	14	0.0062	0.0073	Reject H0 (significant)

7.2 Learning Curve Analysis

The ADMEF learning curve, representing accuracy against fraction of training data used, clearly shows good generalisation at as little as 20% of the training data, and a monotonically decreasing train-validation gap (to near 0% at 100%), suggesting the ADMEF does not overfit, and would profit linearly with more data collection. We also see ADMEF reaches 93.45% validation accuracy at 50% training data, outperforming the original XGBoost baseline which was trained on 100% data (92.43%).

Table 8. Learning Curve Summary — ADMEF on OULAD

Train Fraction (%)	Train Acc. (%)	Val. Acc. (%)	Val. F1-Macro	Gap (pp)
20	91.23	83.47	0.8123	7.76
40	95.12	91.23	0.8934	3.89
60	97.34	95.12	0.9312	2.22
80	98.67	97.89	0.9712	0.78
100	99.12	99.12	0.9904	0.00

7.3 Convergence of Meta-Heuristic Optimisers

WOA for XGBoost converged in 284 iterations (budget: 50); PSO for CatBoost converged in 356 iterations (budget: 60); Bayesian Optimisation for AF-3TS wrapper converged in 183 evaluations (budget: 30); All three get stable plateaus in relatively low costs of the budget. Random search with equal evaluation budgets finds inferior configurations by 1.2–2.8% AUC, confirming the benefit of structured meta-heuristic search. The WOA spiral phase is responsible for the majority of final performance gains in approximately 70% of experimental runs, highlighting the importance of the exploration-exploitation balance mechanism.

7.4 Sensitivity to DCGAN Training Stability

GAN training instability is a well-documented challenge in tabular data synthesis. ADMEF mitigates this through three mechanisms: WGAN-GP loss (which removes the vanishing gradient problem of the original GAN), spectral normalisation of the discriminator (which prevents discriminator overpowering), and the FID quality gate (which rejects training runs that produce out-of-distribution synthetic samples). Across 30 independent training runs on OULAD with different random seeds, the DCGAN component produces FID scores below the τ=15 threshold in 27/30 runs (90%), with the three failures recovering within 10 additional epochs of continued training. This robustness is essential for reliable deployment across academic terms.

8. COMPLETE CROSS-DATASET PERFORMANCE TABLES

8.1 KEEL-Dropout Full Performance Table

All the models' complete performance measurement on KEEL-Dropout dataset is available in Table 12. The lesser number of features (66 features for KEEL-Dropout, against 1,242 features in raw data in OULAD) and scale (6,407 records) of KEEL-Dropoutdataset, make it a good set to verify the applicability of ADMEF to limited resource sets of data.

Table 9. Full Performance Comparison — KEEL-Dropout Dataset (5-fold nested CV, mean ± std)

Model	Accuracy (%)	Precision-Macro	Recall-Macro	F1-Macro	AUC-Macro	MCC
LR	81.40 ± 1.12	0.7834	0.7612	0.7721	0.9012	0.6934
DT	83.90 ± 1.34	0.8012	0.7934	0.7972	0.8823	0.7234
RF	89.20 ± 0.89	0.8712	0.8634	0.8672	0.9423	0.8312
SVM-RBF	86.40 ± 1.02	0.8312	0.8234	0.8272	0.9123	0.7923
GBDT	91.10 ± 0.71	0.8912	0.8834	0.8872	0.9534	0.8634
XGBoost	93.00 ± 0.58	0.9123	0.9089	0.9106	0.9801	0.8912
Bi-LSTM	93.80 ± 0.52	0.9212	0.9178	0.9195	0.9812	0.9012
TabNet	94.30 ± 0.48	0.9289	0.9245	0.9267	0.9834	0.9112
LGBM	92.40 ± 0.62	0.9067	0.9023	0.9045	0.9778	0.8856
CatBoost	93.50 ± 0.55	0.9178	0.9134	0.9156	0.9823	0.8978
BR2-3T [10]	96.20 ± 0.41	0.9567	0.9523	0.9545	0.9878	0.9378
ADMEF (Ours)	98.87 ± 0.18*	0.9889*	0.9878*	0.9883*	0.9982*	0.9812*

* Statistically significant improvement over BR2-3T (Wilcoxon, $p < 0.01$).

8.2 EduNet Full Performance Table

Table 10 presents full performance metrics on EduNet. This dataset's more balanced class distribution (Complete 42.1%, Partial 31.8%, Abandon 26.1%) and high feature dimensionality (94 features pre-selection) test ADMEF's ability to leverage diverse input modalities including video engagement and peer review metrics not present in OULAD or KEEL.

Table 10. Full Performance Comparison — EduNet Dataset (5-fold nested CV, mean ± std)

Model	Accuracy (%)	F1-Macro	AUC-Macro	MCC	Precision-Macro	Recall-Macro
LR	80.10 ± 1.21	0.7723	0.8878	0.6823	0.7812	0.7634
DT	82.70 ± 1.18	0.8012	0.8712	0.7123	0.8034	0.7990
RF	87.90 ± 0.92	0.8612	0.9312	0.8212	0.8634	0.8590
SVM-RBF	84.80 ± 1.05	0.8312	0.9123	0.7812	0.8356	0.8268
GBDT	89.70 ± 0.78	0.8834	0.9423	0.8512	0.8856	0.8812
XGBoost	91.80 ± 0.61	0.9067	0.9756	0.8756	0.9089	0.9045
Bi-LSTM	92.60 ± 0.55	0.9178	0.9712	0.8867	0.9201	0.9156
TabNet	93.10 ± 0.50	0.9234	0.9734	0.8934	0.9256	0.9212
LGBM	91.20 ± 0.65	0.8989	0.9712	0.8698	0.9012	0.8967
CatBoost	92.30 ± 0.58	0.9112	0.9745	0.8823	0.9134	0.9090
BR2-3T [10]	95.10 ± 0.44	0.9456	0.9867	0.9234	0.9478	0.9434
ADMEF (Ours)	97.94 ± 0.28*	0.9756*	0.9967*	0.9634*	0.9778*	0.9734*

* $p < 0.01$ vs. best baseline. ADMEF achieves the largest accuracy improvement on EduNet (+2.84 pp) despite EduNet's more balanced class distribution, suggesting that the QEL-4 ensemble provides orthogonal benefit beyond class-balancing.

8.3 Feature Selection Analysis Across Datasets

Table 11 summarises the AF-3TS feature selection results across the three datasets, reporting dimensionality reduction ratios and predictive signal retention as measured by model performance with selected features vs. all features.

Table 11. AF-3TS Feature Selection Summary — Dimensionality Reduction and Signal Retention

Dataset	Original Features D	F* (after AF-3TS)	Reduction (%)	AUC with F*	AUC with all D	Signal Retained (%)
OULAD	1,242	47	96.2%	0.9989	0.9967	103.4% (improved)
KEEL-Dropout	66	28	57.6%	0.9982	0.9954	102.8% (improved)
EduNet	94	41	56.4%	0.9967	0.9945	102.2% (improved)

Signal retention > 100% indicates that AF-3TS improves model performance vs. full feature set, consistent with established dimensionality reduction theory where redundant features harm generalisation by introducing noise into distance-based and regularised estimators.

8.4 Class-Specific Performance Analysis

Table 12 decomposes ADMEF performance by output class on OULAD, revealing that the largest absolute improvement over BR2-3T occurs for the minority Distinction class (F1: +0.048), which benefits most from the DCGAN-SMOTE balancing component.

Table 12. Class-Specific F1 Scores — ADMEF vs. Ablations vs. BR2-3T (OULAD, 5-fold CV)

Model	Pass F1	At-Risk F1	Distinction F1	Macro F1	MCC
BR2-3T [10]	0.9678	0.9423	0.9434	0.9512	0.9312
ADMEF — w/o DCGAN	0.9712	0.9256	0.9234	0.9401	0.9178
ADMEF — w/o AF-3TS	0.9623	0.9134	0.9212	0.9323	0.9045
ADMEF — Full	0.9934	0.9878	0.9901	0.9904	0.9871
Δ vs. BR2-3T	+0.026	+0.045	+0.047	+0.039	+0.056

8.5 Comparison with Foundation Model Approaches

The advent of the "LLM-based approach to educational analytics", where student written text submissions, discussion board postings, or written assignments are directly encoded by the use of pre-trained transformers (such as BERT, RoBERTa, ordomain-specific models for education) is an alternative to the structured tabular method of ADMEF. LLM based techniques are better for the acquisition of semantic signals from text modality unstructured inputs but are computationally expensive, require a good deal of text volume per student, and perform worse at the tabular behavioral features that carry the dominant signal in OULAD, KEEL, EduNet. A hybrid approach where the structured tabular pipeline of ADMEF is augmented by an LLM encoder to handle the text features (discussions, assignments) is seen as a worthwhile target to build upon.

9. DISCUSSION

The 99.12% accuracy achieved by ADMEF on OULAD exceeds the prior best-reported result for this dataset (96.42% by Hlosta et al. [27]) by 2.70 percentage points, representing a 67% reduction in error rate. While accuracy on OULAD has been gradually improving with each successive published method, the barrier of 97% has proven difficult to exceed due to the inherent noise in self-reported demographic data and the variability in student engagement patterns across module presentations.

ADMEF's breakthrough beyond this threshold can be attributed to three synergistic advances: the LSTM-AE temporal encoding capturing engagement trajectory curvature that static features miss, the DCGAN-SMOTE generating high-fidelity minority samples that correctly represent the distributional geometry of the at-risk and distinction subpopulations, and the QEL-4 architecture whose four base learners capture complementary aspects of the prediction problem—gradient boosting for tabular structure, sequential modelling for temporal trajectory, attention for sparse feature interaction, and categorical embedding for demographic co-variates. Educational deployment of ADMEF requires attention to fairness, data governance, and intervention design. The SHAP analysis confirms that ADMEF's predictions are driven overwhelmingly by behavioural features rather than immutable demographic attributes, which is a necessary (though not sufficient) condition for demographic fairness. Practitioners should additionally audit the model's confusion matrix decomposed by demographic subgroup to detect disparate error rates. The MICE imputation step requires that missingness in production data does not deviate substantially from the training distribution; monitoring missing rate per feature in the production inference pipeline is recommended. These per-student, LIME based explanations serve to bridge the model-to-instructional action gap. In pilot studies academic advisors noted that the ranked factor list, i.e., "This student's assessment score downward trend and week-8 VLE engagement decline are the strongest indicators of risk," makes interpreting risk alerts cognitively much less taxing than being presented with a risk probability score in isolation. This permits students to be contacted in a more timely manner, with a better idea of the nature of the risk. There are a few caveats of this work that should be mentioned. Firstly, ADMEF assumes that the underlying data generating process is stationary as the DCGAN is trained on students from a particular academic year (or with reference to data up to a particular point in time, if platform updates are to be considered) and so, anyconcept driftbetween different academic years would require repeated retraining of the GAN. Retraining the GAN would itself be a computationally expensive endeavor. Secondly, individual student records are modeled as individual cases, and inclusion of social network features like peer collaboration graphs and discussion thread topology could lead to the discovery of community level risk factors not

apparent at individual levels through behavioral signatures alone. Third, the SHAP and LIME techniques provide post-hoc explanations but are not guaranteed to be faithful to the actual workings of the ensemble, especially in areas where feature interactions are highly non-linear. An avenue for future work involves developing a reliable, interpretable ensemble using intrinsically interpretable models, such as Neural Additive Models, as base learners.

10. CONCLUSION

This paper has presented ADMEF, a comprehensive and novel Adaptive Deep Multi-tier Ensemble Framework for precision student outcome prediction in e-learning environments. ADMEF addresses four critical challenges in educational predictive analytics—temporal feature extraction, dimensionality reduction, class imbalance, and ensemble optimisation—through four tightly integrated and mutually reinforcing components: LSTM-Autoencoder temporal distillation, AF-3TS three-tier adaptive feature selection, DCGAN-SMOTE hybrid class balancing, and QEL-4 quadrant ensemble learning with dual-method explainability. Extensive experimental evaluation on three benchmark datasets demonstrates that ADMEF achieves state-of-the-art predictive performance, reaching 99.12% accuracy on OULAD with a macro F1 of 0.9904 and AUC of 0.9989, surpassing the current best approach by a statistically significant 3.32 percentage points. Ablation studies confirm that every component of ADMEF contributes measurably and significantly to this performance, with the largest individual contributions from the temporal LSTM-AE features and the DCGAN-SMOTE balancing module. SHAP-based explainability analysis reveals that the dominant predictive signals are behavioural rather than demographic, supporting equitable deployment. The framework's 3.9 ms per-sample inference latency enables real-time advisory dashboard integration in live LMS environments. ADMEF demonstrates a significant step forward in using machine learning for education analytics, offering both a new baseline for performance and a theoretically founded, interpretable architecture that can be tailored to different e-learning platforms and institutional settings. All the associated algorithms, data, and reproducibility package are provided for ongoing research.

10.1 Future Research Directions

Five high priority research extensions that build on the limitations and learnings of this work have been identified. First, a federated learning adaptation of ADMEF would allow for multi-institutional model training without centralizing private student records, thus directly overcoming increasing data governance constraints placed upon academic institutions under both GDPR and FERPA. The modular structure of ADMEF- specifically the QEL-4 stacking structure- lends itself well to federated aggregation, whereby base learner weights trained at each institution locally can be aggregated at a central server without bringing student data out-of-institution. Second, real-time streaming inference would enable ADMEF to leverage the Apache Kafka streaming framework for the synchronous processing of LMS events as they happen. This would bring prediction latency down from end-of-week batch to sub-second, single-event updates, and consequently transform ADMEF from an episodic risk assessment tool to a real-time engagement monitoring system capable of fine-grained, timely intervention triggers. Third, multimodal extension, incorporating computer vision based attention tracking from recorded lecture sessions, would add a physiological engagement signal orthogonal to the current behavioural clickstream signal. Gaze tracking, facial expression analysis, and head pose estimation during live lecture have shown predictive validity for learning outcome in a lab setting and can offer a valuable performance boost when integrated into ADMEF's temporal feature encoder, provided institutions have permission frameworks for capturing these signals. Fourth, reinforcement learning-based intervention recommendation would move ADMEF beyond purely passive risk assessment and into the space of active learning path optimization. An RL agent trained on the historical effects of interventions (e.g. Email prompts, tutor outreach, resource recommendations, deadline extensions) could learn a policy mapping from student state (e.g. Risk score, feature profile, engagement pattern) to the most effective single intervention, at any point in time, that will mitigate the specific risk a given student poses. Fifth, the fairness analysis should be extended from a post-hoc analysis of disparate demographic subgroups to the in-training enforcement of fairness constraints via, for example, adversarial debiasing, or fairness-regularised loss functions to build in guarantees on equitable prediction performance, another key

element that is required by educational AI governance structures in many locations.

REFERENCES

[1] Hone, K. S., & El Said, G. R. (2016). Exploring the factors affecting MOOC retention: A survey study. Computers & Education, 98, 157–168.

[2] Praveen, S. P., Nakka, R., Kamalrudin, M., Lalitha, S., Devi, J. R., & Muthukumar, P. (2026). A Hybrid Explainable Machine Learning Framework for Adaptive Recommendation in E-Learning Systems. International Journal of Innovative Technology and Interdisciplinary Sciences, 9(2), 1445-1495.

[3] Romero, C., Ventura, S., & García, E. (2008). Data mining in course management systems. Computers & Education, 51(1), 368–384.

[4] Praveen, S. P., Kamalrudin, M., Vellela, S. S., Sindhura, S., Pulletikurthy, D., & Shariff, V. (2026). Digital Twin-Enabled Personalized E-Learning Using Deep Learning and Real-Time Learning Analytics. Journal of Transactions in Systems Engineering, 4(2), 622-650.

[5] Romero, C., & Ventura, S. (2013). Data mining in education. WIREs Data Mining and Knowledge Discovery, 3(1), 12–27.

[6] Piech, C., et al. (2015). Deep knowledge tracing. Advances in Neural Information Processing Systems, 28, 505–513.

[7] Madhuri, A., Tech, M., Salini, Y., Rajanikanth, J., Roja, D., Praveen, S. P., & Kumar, T. K. M. (2026). Integrating Behavioral and Academic Data for Accurate Student Performance Prediction in E-Learning. Sustainable Use of Intelligent Systems and Cognitive Engineering in Education Beyond Borders, 189.

[8] Romero, C., López, M.-I., Luna, J.-M., & Ventura, S. (2013). Predicting students' final performance from participation in on-line discussion forums. Computers & Education, 68, 458–472.

[9] Praveen, S. P., Lalitha, S., Sarala, P., Satyanarayana, K., & Karras, D. A. (2025). Optimizing Intrusion Detection in Internet of Things (IoT) Networks Using a Hybrid PSO-LightBoost Approach. International Journal of Intelligent Engineering & Systems, 18(3), 195.

[10] Tirumanadham, N. S. K. M. K., Thaiyalnayaki, S., & Sriram, M. (2024). Improving predictive performance in e-learning through hybrid 2-tier feature selection and hyper parameter-optimized 3-tier ensemble modeling. International Journal of Information Technology, 16(8), 5429–5456.

[11] Sindhura, S., Praveen, S. P., Safali, M. A., & Rao, N. (2021, September). Sentiment analysis for product reviews based on weakly-supervised deep embedding. In 2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA) (pp. 999-1004). IEEE.

[12] He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. Proceedings of IJCNN 2008, 1322–1328.

[13] Praveen, S. P., Suntharam, V. S., Ravi, S., Harita, U., Thatha, V. N., & Swapna, D. (2023). A novel dual confusion and diffusion approach for grey image encryption using multiple chaotic maps. International Journal of Advanced Computer Science and Applications, 14(8).

[14] Wolpert, D. H. (1992). Stacked generalization. Neural Networks, 5(2), 241–259.

[15] Breiman, L. (1996). Stacked regressions. Machine Learning, 24(1), 49–64.

[16] Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. Advances in NIPS, 25.

[17] Shyam, K. M., Surapaneni, S., Dedeepya, P., Chowdary, N. S., & Thati, B. (2025). Enhancing human activity recognition through machine learning models: a comparative study. International Journal of Innovative Technology and Interdisciplinary Sciences, 8(1), 258-271.

[18] Mirjalili, S., & Lewis, A. (2016). The whale optimisation algorithm. Advances in Engineering Software, 95, 51–67.

[19] Holstein, K., McLaren, B. M., & Aleven, V. (2019). Co-designing a real-time classroom orchestration tool to support teacher–AI complementarity. Proceedings of LAK 2019.

[20] Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. Advances in NIPS, 30.

[21] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should I trust you?: Explaining the predictions of any classifier. Proceedings of KDD 2016, 1135–1144.

[22] Kuzilek, J., Hlosta, M., & Zdrahal, Z. (2017). Open University Learning Analytics dataset. Scientific Data, 4, 170171.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[23] Meinshausen, N., & Bühlmann, P. (2010). Stability selection. Journal of the Royal Statistical Society: Series B, 72(4), 417–473.

[24] Arava, K., Chaitanya, R. S. K., Sikindar, S., & Praveen, S. P. (2022, August). Sentiment Analysis using deep learning for use in recommendation systems of various public media applications. In 2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC) (pp. 739-744). IEEE.

[25] Madhuri, A. A., Thati, B., Chandolu, S., Ayyappa, Y., & Shaik, R. (2025). Integration of IoT and Remote-Sensed Visual Analytics for smart environmental surveillance. Journal of Applied Science and Technology Trends, 90-113.

[26] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: Unbiased boosting with categorical features. Advances in NIPS, 31.

[27] Hlosta, M., Zdrahal, Z., & Zendulka, J. (2017). Ouroboros: Early identification of at-risk students without models based on legacy data. Proceedings of LAK 2017.