Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X
Volume 3, Issue 12, 2025

SLA-AWARE PREDICTIVE RESOURCE ALLOCATION IN 6G RADIO ACCESS NETWORKS USING CLIENT-TEMPORAL TELEMETRY LEARNING

Samina Rajper[*1], Rafaqat Hussain Arain[2], Riaz Ahmed Shaikh[3], Attique Ur Rehman[4], Bushra Shaikh[5], Irfan Dharejo[6]

[*1,2,3,4,5,6]*Institute of Computer Science, Shah Abdul Latif University, Khairpur, Pakistan*

[*1]samina.rajper@salu.edu.pk, [2]rafaqat.arain@salu.edu.pk, [3]riaz.shaikh@salu.edu.pk,
[4]attiqueurrehman.cs@gmail.com, [5]shykhbushra23@gmail.com , [6]dharejoirfan89@gmail.com

DOI: https://doi.org/10.5281/zenodo.20080277

Keywords
6G networks, resource allocation, SLA prediction, radio access network telemetry, machine learning, network slicing, quality of service, predictive resource management, calibration, risk-aware decisioning.

Article History
Received: 28 October 2025
Accepted: 12 December 2025
Published: 26 December 2025

Copyright @Author

Corresponding Author: *
Samina Rajper

Abstract

Sixth-generation radio access networks are expected to support heterogeneous services whose performance requirements vary across throughput, latency, reliability, mobility, and slice-level demand. Conventional resource allocation mechanisms are often reactive because network resources are adjusted after congestion or service-level agreement (SLA) degradation has already appeared. This paper proposes an SLA-aware predictive resource allocation framework for 6G radio access networks using client-temporal telemetry learning. The proposed framework predicts whether the next 90-second client-window will satisfy the SLA and converts the predicted probability into a risk score for proactive resource prioritization. The study uses the public Wireless 6G Network Dataset of Resource Allocation, containing 30,000 telemetry observations from 200 clients. Each observation includes radio-quality, physical-resource-block usage, retransmission, mobility, traffic-slice, throughput, latency, reward, and current-SLA indicators. A client-aware chronological split is used to preserve deployment realism, producing 21,000 training, 4,400 validation, and 4,600 test observations. Temporal lag, delta, and rolling-window features are generated, resulting in 265 predictive features. Seven machine learning models are evaluated: Logistic Regression, Random Forest, Gradient Boosting, HistGradientBoosting, multilayer perceptron, XGBoost, and LightGBM. Results show that Gradient Boosting achieves the best overall predictive performance with 0.8585 accuracy, 0.8566 macro-F1, 0.9403 ROC-AUC, and 0.9573 PR-AUC. Random Forest achieves the strongest violation recall of 0.9140, reducing missed SLA violations to 167 out of 1,941 violation cases. HistGradientBoosting provides the best probability calibration with an expected calibration error of 0.0176. These findings demonstrate that calibrated telemetry learning can support predictive SLA assurance and risk-aware resource prioritization in 6G networks.

I. Introduction

Future 6G radio access networks (RANs) are expected to support a dense and highly heterogeneous ecosystem of users, connected devices, edge services, immersive applications, industrial automation, and latency-sensitive communication. Unlike traditional mobile broadband traffic, many emerging services will require simultaneous management of throughput, latency, reliability, mobility, and network-slice isolation. A network controller therefore cannot rely only on average traffic load or static provisioning. It must anticipate service degradation before it occurs and prioritize resources for clients that are likely to violate performance requirements in the near future [1], [2].

Resource allocation is a long-standing problem in wireless communications. It typically involves assigning limited radio, computing, spectrum, scheduling opportunities, or physical resource block (PRB) capacity to users and services. In 5G and 6G environments, this allocation problem becomes more difficult because service requirements vary across network slices and applications. Enhanced mobile broadband services may require high throughput, ultra-reliable low-latency services require low delay, while massive machine-type communication requires scalable support for many low-power devices. A single rule-based allocation policy can therefore be too rigid for realistic, mixed-service networks [3]-[6].

Machine learning offers a promising decision-support layer for wireless resource management because it can learn nonlinear relationships among radio conditions, traffic load, mobility, retransmission behavior, latency, throughput, and service outcomes. However, many learning-based resource allocation studies focus on simulation settings, optimization formulations, or random train-test partitions. Random evaluation can overestimate deployment performance when temporally adjacent observations from the same client are distributed across training and testing sets. For telemetry-driven network management, temporal ordering and client grouping are important because the model must predict future service states, not memorize nearby observations [7]-[10].

Recent work on 6G SLA-risk forecasting has further shown that client heterogeneity and unseen operating regimes can substantially affect decision reliability, motivating evaluation protocols that go beyond random splits and include robustness-oriented metrics such as balanced accuracy and worst-regime F1 [21].

This paper formulates 6G resource allocation as a predictive SLA-assurance task. Instead of asking whether the current window satisfies the SLA, the model predicts whether the next client-window will satisfy the SLA. This next-window setting is operationally meaningful because the prediction can be used before degradation occurs. The estimated probability of SLA satisfaction is then converted into a violation-risk score. Clients with high predicted risk can be prioritized for additional resource attention, slice-level scheduling support, or controller-side intervention.

The proposed framework is evaluated on the Wireless 6G Network Dataset of Resource Allocation [11]. The dataset contains 30,000 client-window telemetry records from 200 clients. The target variable, sla_ok_next, indicates next-window SLA satisfaction, where class 1 denotes SLA OK and class 0 denotes violation or not-OK. The dataset includes radio-quality metrics such as CQI and SINR, resource-usage metrics such as PRB mean and PRB p95, reliability metrics such as BLER and HARQ retransmissions, mobility/load metrics such as UE and HO, slice composition variables, and performance metrics such as downlink throughput, uplink throughput, latency, reward, and current SLA status.

The main contributions of this paper are as follows:

1. A predictive SLA-aware formulation is developed for 6G RAN resource allocation using next-window client telemetry.
2. A client-aware chronological evaluation protocol is used to reduce temporal

leakage and better represent deployment conditions.

3. A rich temporal feature representation is constructed using lag, delta, and rolling-window statistics, resulting in 265 predictive features.
4. Seven machine learning models are compared under a unified evaluation pipeline, including linear, ensemble, boosting, and neural baselines.
5. The study reports not only accuracy but also macro-F1, ROC-AUC, PR-AUC, Brier score, expected calibration error, violation recall, false-negative rate, and missed violations.
6. A risk-aware allocation layer is introduced to convert predicted SLA probability into proactive resource-prioritization support.

The remainder of this paper is organized as follows. Section II reviews related work. Section III describes the dataset and problem formulation. Section IV presents the proposed methodology. Section V explains the experimental setup. Section VI reports and discusses the results. Section VII compares the study with prior work. Section VIII presents limitations and future directions. Section IX concludes the paper.

IEER
Institute for Excellence in Education & Research

II. Related Work

A. Resource Allocation and Network Slicing in 5G/6G Networks

Resource allocation is central to modern wireless systems because radio spectrum, PRBs, transmit power, scheduling opportunities, and edge-computing resources are limited. In 5G and 6G networks, this problem is further complicated by network slicing. A slice may represent a particular application class, traffic type, or service profile. Because different slices have different quality-of-service requirements, resource allocation should adapt to service-specific performance goals rather than only maximizing aggregate throughput [3], [4], [10].

Recent surveys on resource management and network slicing emphasize the transition from static provisioning toward intelligent and adaptive management [3]-[6]. These studies highlight that future 6G systems will need learning-based support for traffic prediction, resource orchestration, slice placement, scheduling, interference management, and service assurance. However, many existing works discuss resource allocation at a high conceptual or optimization level, while fewer studies provide a reproducible telemetry-based pipeline that predicts SLA degradation and connects it to a practical risk-based decision layer.

B. Machine Learning for Wireless Network Management

Machine learning has been increasingly used in wireless network management because network telemetry contains patterns that are difficult to capture using simple rule-based models [1], [5], [8], [9]. Tree-based ensembles and boosting methods are particularly attractive for tabular telemetry because they handle nonlinear feature interactions, heterogeneous feature scales, and mixed signal patterns [13], [14], [16], [17]. Neural models can also be used, but they often require larger datasets, careful tuning, and hardware compatibility [18]-[20].

For 6G networks, learning-based techniques have been proposed for channel prediction, traffic forecasting, network slicing, dynamic resource allocation, federated optimization, and self-organizing network control [1], [2], [7]-[10]. Still, a key concern is evaluation realism. If samples from the same client and adjacent time windows appear in both training and testing sets, the reported performance may reflect leakage rather than true future prediction. This paper therefore uses a client-aware chronological split as the main protocol. In particular, federated SLA-risk forecasting under leave-one-regime-out evaluation has shown that models with strong ranking performance may still differ in thresholded decision quality under unseen 6G RAN regimes [21].

C. SLA Prediction and Reliability-Aware Decisioning

SLA prediction aims to estimate whether a service will satisfy predefined performance requirements.

In network management, an SLA can be defined using throughput, latency, packet loss, jitter, reliability, or a combination of these indicators [3], [4], [6]. The target in this study is sla_ok_next, which directly supports next-window prediction. This enables proactive decision-making because the model estimates whether a client is likely to experience a violation before the next 90-second interval begins.

Reliability of predicted probabilities is important in this setting. If a model is overconfident or poorly calibrated, its probability scores may mislead the resource controller. Calibration metrics such as expected calibration error (ECE) and reliability diagrams are therefore included [15]. A calibrated model is particularly useful when risk thresholds are used to decide whether a client should receive resource attention.

Reliability of predicted probabilities is important in this setting because an overconfident SLA predictor can mislead a resource controller when probability scores are converted into risk thresholds. Recent operational screening studies in security and healthcare domains similarly show that calibrated risk scores, threshold selection, and uncertainty-aware outputs are useful when model predictions are used for auditable triage rather than only offline accuracy reporting [22], [23]. In 6G RAN settings, client drift and unseen operating regimes further motivate evaluation beyond random splits, including robustness-oriented metrics that reflect thresholded decision quality [21].

D. Research Gap

The literature contains many studies on 5G/6G resource allocation, network slicing, and machine learning-assisted wireless control [1]-[10]. However, three gaps remain. First, many studies emphasize optimization or simulation rather than public telemetry-based reproducibility. Second, many machine learning studies report accuracy but do not include violation-focused metrics such as false-negative rate and missed SLA violations. Third, probability calibration is often overlooked even though resource allocation depends on reliable risk estimates [15]. This paper addresses these gaps by combining public 6G telemetry, client-aware chronological splitting, temporal feature engineering, model calibration, and risk-aware allocation analysis. This gap is consistent with recent operational ML evidence showing that strong discrimination alone is insufficient when predicted scores are used as actionable risks in automated or semi-automated workflows [22].

III. Dataset and Problem Formulation

A. Dataset Description

This study uses the Wireless 6G Network Dataset of Resource Allocation [11]. The main telemetry file used in the experiment is 6g_fl_telemetry_200_clients.csv. The dataset contains 30,000 observations and 29 original columns. Each row represents a client-window observation. There are 200 clients, and the window duration is 90 seconds. The target variable sla_ok_next is already provided in the dataset.

The raw telemetry variables include client identifiers, temporal indicators, radio-quality metrics, resource-usage indicators, reliability metrics, mobility/load indicators, traffic-slice composition, performance indicators, reward, current SLA status, and next-window SLA status. The dataset contains a moderately imbalanced but usable class distribution: 12,940 observations are SLA violation/not-OK cases and 17,060 observations are SLA OK cases.

The complete experimental workflow and decision-support pipeline are summarized in Fig. 1.

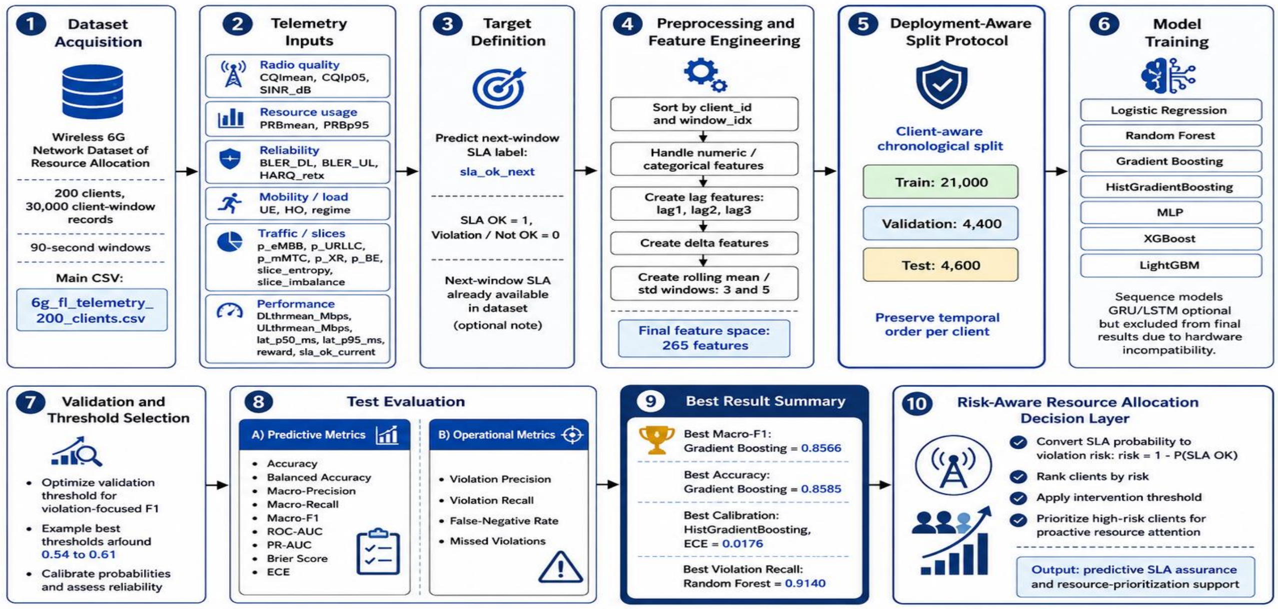

FIGURE 1. Proposed methodology for SLA-aware predictive resource allocation in 6G RANs.

The next-window SLA class distribution is shown in Fig. 2.

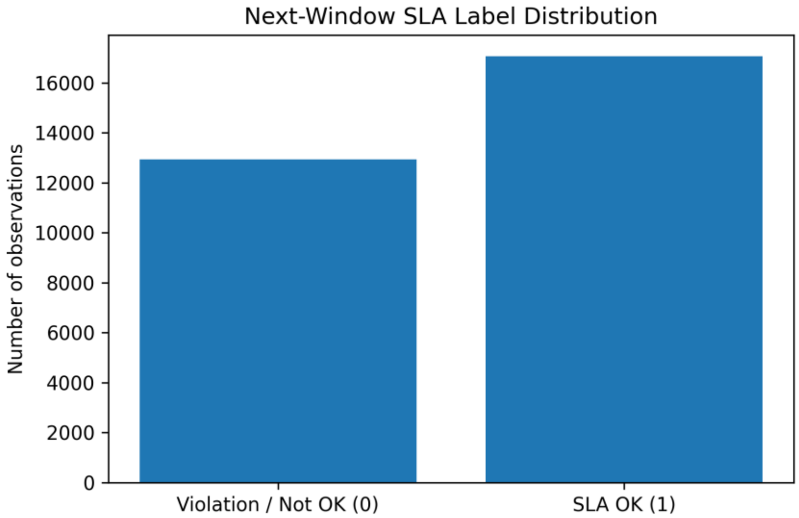

FIGURE 2. Next-window SLA label distribution. The dataset contains 12,940 violation/not-OK observations and 17,060 SLA OK observations.

B. Telemetry Feature Groups

The original telemetry variables are grouped into six categories: radio quality, resource usage, reliability, mobility/load, traffic-slice composition, and performance. Table I summarizes these feature groups

Table I. Telemetry Feature Groups Used for SLA Prediction

Feature Group	Variables	Operational Meaning
Radio quality	CQImean, CQIp05, SINR_dB	Channel condition and signal quality

Feature Group	Variables	Operational Meaning
Resource usage	PRBmean, PRBp95	Average and high-percentile PRB pressure
Reliability	BLER_DL, BLER_UL, HARQ_retx	Error and retransmission behavior
Mobility/load	UE, HO, regime	User load, handover, and traffic regime
Traffic/slices	p_eMBB, p_URLLC, p_mMTC, p_XR, p_BE, slice_entropy, slice_imbalance	Slice traffic composition and imbalance
Performance	DLthrmean_Mbps, ULthrmean_Mbps, lat_p50_ms, lat_p95_ms, reward, sla_ok_current	Throughput, latency, reward, and current SLA evidence

C. Target Definition

The learning task is binary classification. Let (x_{i,t}) denote the telemetry vector for client (i) at window (t). The model predicts whether the next window satisfies the SLA:

y_hat[i,t+1] = f(x[i,t])

where (y_{i,t+1}=1) indicates SLA OK and (y_{i,t+1}=0) indicates SLA violation or not-OK. The dataset already contains the next-window label:

y[i,t+1] = sla_ok_next

For operational interpretation, the violation label is defined as:

v[i,t+1] = 1 - y[i,t+1]

Thus, (v_{i,t+1}=1) denotes a predicted or observed SLA violation.

D. Predictive Resource Allocation Interpretation

The classifier estimates the probability that the next window will satisfy the SLA:

p[i,t+1] = P(y[i,t+1] = 1 | x[i,t])

The predicted violation risk is then defined as:

r[i,t+1] = 1 - p[i,t+1]

A simple risk-aware allocation decision can be written as:

a[i,t+1] = 1 if r[i,t+1] >= tau_r, otherwise 0

where a[i,t+1] = 1 means that the client is selected for resource attention, and tau_r is the risk threshold.

IV. Proposed Methodology

A. Overview

The proposed methodology contains ten stages: dataset acquisition, telemetry input organization, target definition, preprocessing, temporal feature engineering, deployment-aware splitting, model training, validation threshold selection, test evaluation, and risk-aware resource allocation. The overall framework is shown in Fig. 1.

Following reliability-first operational screening practice, the predicted probability is treated as a decision-support score only after calibration quality is assessed using Brier score and ECE [22].

B. Preprocessing

The telemetry data are first sorted by client_id and window_idx to preserve temporal structure. Numerical features are imputed using median values and standardized when required. The categorical variable regime is encoded using one-hot encoding. Identifier columns such as client_id and window_idx are used for grouping and splitting but are not directly treated as predictive content in the same way as telemetry measurements. The preprocessing and model-training pipeline was implemented using common machine-learning tooling for tabular prediction [12].

C. Temporal Feature Engineering

To capture short-term evolution in network behavior, lag, delta, and rolling-window features are generated for numerical telemetry variables. Lag features are computed as:

$x[i,t]^{(k)} = x[i,t-k], \quad k \in \{1, 2, 3\}$

where k ∈ {1, 2, 3}. Delta features are computed as:

$\Delta x[i,t]^{(k)} = x[i,t] - x[i,t-k]$

Rolling mean features are computed as:

$\bar{x}[i,t]^{(w)} = (1/w) \sum_{j=1}^{w} x[i,t-j], \quad w \in \{3, 5\}$

where w ∈ {3, 5}. Rolling standard deviation features are similarly computed over previous windows. After feature engineering, the final feature space contains 265 features.

D. Deployment-Aware Split Protocol

A client-aware chronological split is used as the main evaluation protocol. For each client, earlier windows are used for training, middle windows for validation, and later windows for testing. This design avoids future-to-past leakage and better represents deployment, where a model trained on historical telemetry predicts future network states.

Table II. Client-Aware Chronological Split

Split	Number of Observations	Role
Training	21,000	Model fitting
Validation	4,400	Threshold selection and model tuning
Test	4,600	Final evaluation

The test set contains 1,941 violation/not-OK cases and 2,659 SLA OK cases.

E. Learning Models

Seven machine learning models are evaluated:

1. Logistic Regression;
2. Random Forest;
3. Gradient Boosting;
4. HistGradientBoosting;
5. Multilayer Perceptron (MLP);
6. XGBoost;
7. LightGBM.

The model set includes a linear classifier, a bagging-based ensemble, boosting-based ensembles, histogram-based boosting, a neural baseline, and scalable gradient-boosted tree models [12]-[17]. Sequence models such as GRU and LSTM were prepared because recurrent architectures are commonly used for temporal sequence modeling [19], [20], but they were excluded from the final comparison because the Kaggle GPU environment produced a CUDA RNN-kernel incompatibility during execution. The tabular models completed successfully and provide the final reported results.

IEER
Institute for Excellence in Educa

F. Validation Threshold Selection

Instead of using a fixed threshold of 0.5, the threshold is optimized on the validation set with emphasis on violation-focused performance. This is important because the operational cost of missing an SLA violation can be higher than the cost of unnecessarily prioritizing a stable client. The selected thresholds range from 0.54 to 0.61 across models.

G. Calibration Metrics

Calibration is evaluated because risk-aware allocation depends on reliable probabilities [15]. ECE is computed by dividing predicted probabilities into M bins. Let B_m be the set of samples in bin m. ECE is defined as:

$ECE = \sum_{m=1}^{M} (|B_m|/n) \, |acc(B_m) - conf(B_m)|$

where $acc(B_m)$ is the empirical accuracy of bin B_m, $conf(B_m)$ is its average predicted confidence, and n is the total number of samples [15].

The Brier score is also reported:

$Brier = (1/n) \sum_{i=1}^{n} (p_i - y_i)^2$

H. Risk-Aware Allocation Utility

To evaluate decision usefulness, the predicted violation risk can be used to select clients for intervention. A utility score can be defined as:

$U = (1/n)(\alpha TP + \beta FP + \gamma FN + \delta TN)$

where (TP) is a correctly captured violation, (FP) is an unnecessary intervention, (FN) is a missed violation, and (TN) is a correctly ignored stable client. In this study, the predictive results are interpreted mainly through violation recall, false-negative rate, and missed violations because these metrics directly represent the ability to detect clients requiring resource attention.

V. Experimental Setup

A. Implementation Environment

The experiments were executed in a Kaggle notebook environment. The dataset was automatically detected from the Kaggle input directory, and the telemetry file used for model training was 6g_fl_telemetry_200_clients.csv [11]. The implementation used scikit-learn for conventional tabular models [12], XGBoost for scalable boosted trees [13], and LightGBM for efficient gradient boosting [14]. The environment used CUDA for general availability, but sequence models were not included in the final result table because the recurrent neural network kernel was incompatible with the available GPU runtime. All tabular models completed successfully.

B. Feature Processing

The original dataset contains 29 columns and 30,000 observations. After excluding identifiers and the target column and adding temporal lag, delta, rolling mean, and rolling standard deviation features, the final feature matrix contains 265 features: 264 numerical features and one categorical feature group derived from regime.

C. Evaluation Metrics

The predictive metrics are accuracy, balanced accuracy, macro precision, macro recall, macro-F1, ROC-AUC, PR-AUC, Brier score, ECE, Matthews correlation coefficient (MCC), and Cohen's kappa. The operational metrics are violation precision, violation recall, violation F1, false-negative rate for violation cases, missed violations, actual violations, and predicted violations.

Violation recall is especially important:

$$Violation\ Recall = TP\_v / (TP\_v + FN\_v)$$

where (TP_v) is the number of true violation cases correctly predicted as violation and (FN_v) is the number of true violations incorrectly predicted as SLA OK.

The false-negative rate for violations is:

$$FNR\_v = FN\_v / (TP\_v + FN\_v)$$

VI. Results and Discussion

A. Dataset Distribution

Fig. 2 shows the next-window SLA label distribution. Out of 30,000 observations, 12,940 are violation/not-OK cases and 17,060 are SLA OK cases. This distribution is moderately imbalanced but not extreme. It justifies reporting macro-F1, balanced accuracy, and violation-focused metrics instead of relying only on accuracy.

Table III. Dataset and Split Summary

Item	Value
Dataset file	6g_fl_telemetry_200_clients.csv
Total observations	30,000
Original columns	29
Clients	200
Window duration	90 seconds
Final engineered features	265
Training observations	21,000
Validation observations	4,400
Test observations	4,600
Test violation cases	1,941
Test SLA OK cases	2,659

B. Overall Predictive Performance

Table IV reports the main results under the client-aware chronological test split. Gradient Boosting achieves the best overall performance with 0.8585 accuracy and 0.8566 macro-F1, consistent with the strength of boosted decision-tree models on structured tabular problems [17]. Logistic Regression achieves the highest ROC-AUC and PR-AUC, indicating strong ranking performance even with a simple linear model [12]. HistGradientBoosting achieves the lowest ECE, showing the best calibration among the tested models [15]. Random Forest achieves the best violation recall, which is operationally important for proactive resource allocation [16].

Table IV. Model Comparison Under Client-Aware Chronological Split

Model	Thr.	Acc.	Bal. Acc.	Macro-F1	ROC-AUC	PR-AUC	Brier	ECE
LogReg	0.54	0.8528	0.8592	0.8515	**0.9412**	**0.9578**	0.0996	0.0403
RF	0.60	0.8500	0.8586	0.8491	0.9397	0.9571	0.0989	0.0258
GB	0.55	**0.8585**	0.8616	**0.8566**	0.9403	0.9573	0.0979	0.0234
HistGB	0.61	0.8554	**0.8622**	0.8542	0.9406	0.9569	**0.0972**	**0.0176**
MLP	0.60	0.8463	0.8544	0.8453	0.9381	0.9558	0.1007	0.0277
XGB	0.58	0.8543	0.8591	0.8528	0.9397	0.9567	0.0982	0.0214
LightGBM	0.54	0.8567	0.8615	0.8552	0.9396	0.9565	0.0999	0.0341

C. SLA-Violation Detection Performance

For resource allocation, correctly identifying future SLA violations is more important than only predicting the majority class. Table V reports violation-focused metrics. Random Forest achieves the highest violation recall of 0.9140 and misses only 167 out of 1,941 violation cases. HistGradientBoosting and MLP also provide high violation recall, while Gradient Boosting provides the best balance between overall macro-F1 and violation detection.

IEER
Institute for Excellence in Education & Research

Table V. SLA-Violation Detection Metrics

Model	Viol. Prec.	Viol. Rec.	Viol. F1	FNR	Missed	Actual Viol.	Pred. Viol.
LogReg	0.7834	0.9001	0.8377	0.0999	194	1,941	2,230
RF	0.7723	**0.9140**	0.8372	**0.0860**	**167**	1,941	2,297
GB	**0.8025**	0.8815	0.8402	0.1185	230	1,941	2,132
HistGB	0.7848	0.9057	**0.8409**	0.0943	183	1,941	2,240
MLP	0.7701	0.9062	0.8327	0.0938	182	1,941	2,284
XGB	0.7914	0.8892	0.8375	0.1108	215	1,941	2,181
LightGBM	0.7938	0.8923	0.8402	0.1077	209	1,941	2,182

D. Diagnostic Visualization Analysis

The confusion matrices and reliability diagrams in Figs. 3-6 provide a model-level interpretation of both classification behavior and probability reliability. The figures are cited in sequential order to maintain IEEE-style figure numbering.

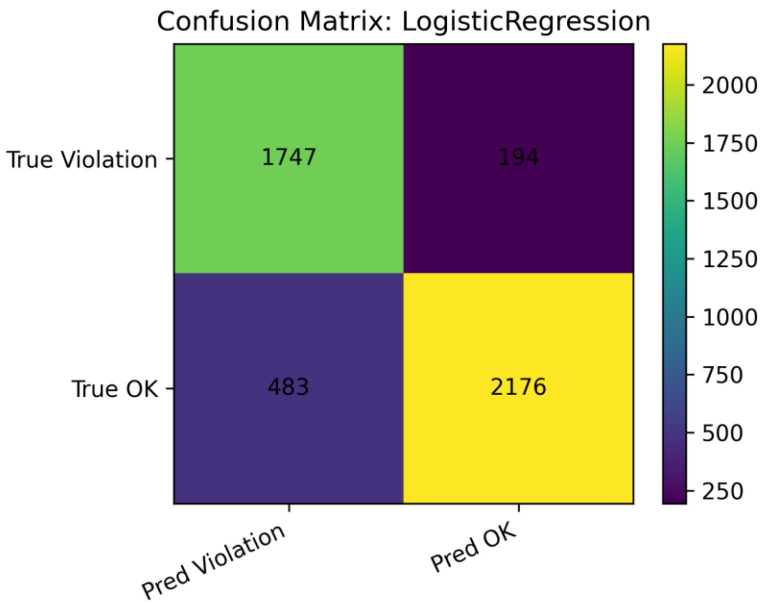

FIGURE 3. Confusion matrix for Logistic Regression under the client-aware chronological split.

As shown in Fig. 3, Logistic Regression correctly identifies 1,747 violation cases and 2,176 SLA OK cases. It misses 194 violation cases and incorrectly flags 483 SLA OK cases as violations. This behavior produces high violation recall while maintaining strong overall ranking performance.

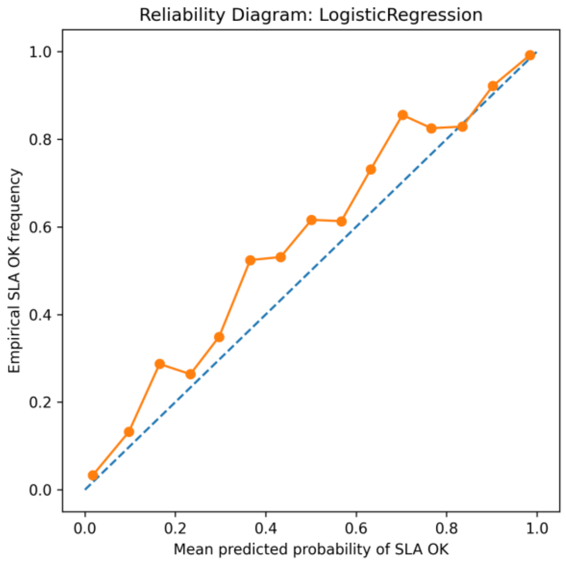

FIGURE 4. Reliability diagram for Logistic Regression.

Fig. 4 shows that the Logistic Regression reliability curve generally follows the diagonal trend, although it deviates in some middle probability ranges. Its expected calibration error is 0.0403, indicating acceptable but not best-in-class calibration.

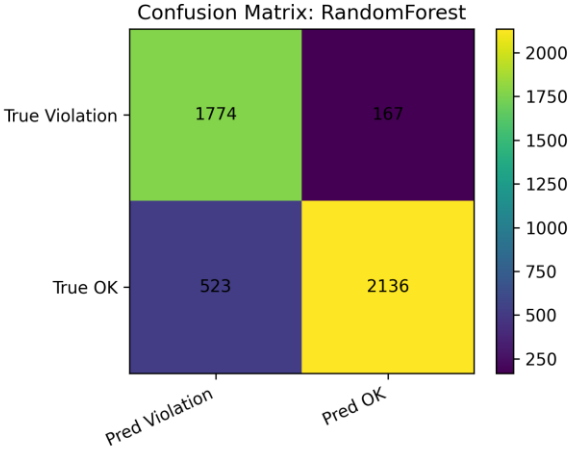

FIGURE 5. Confusion matrix for Random Forest under the client-aware chronological split.

As shown in Fig. 5, Random Forest correctly identifies 1,774 violation cases and 2,136 SLA OK cases. It misses only 167 violation cases, the lowest among all evaluated models, but incorrectly flags 523 SLA OK cases as violations. This reflects an operationally useful trade-off when missed violations are more costly than temporary resource over-prioritization.

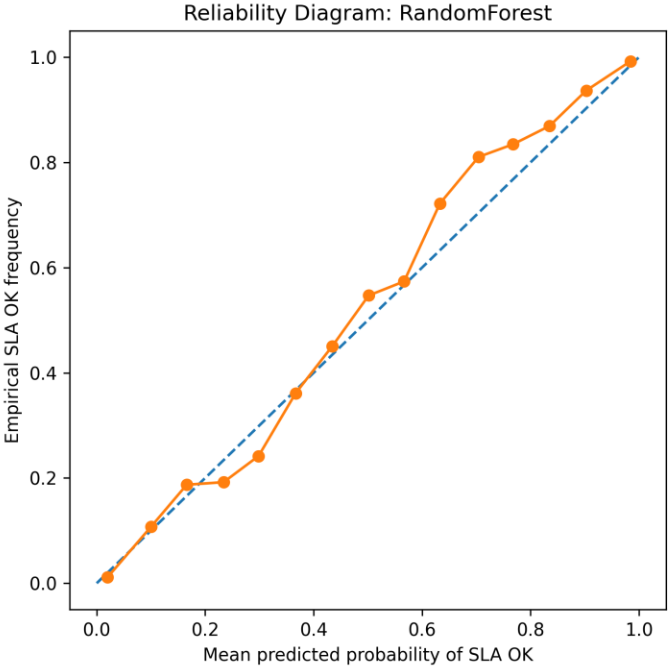

FIGURE 6. Reliability diagram for Random Forest.

Fig. 6 shows that the Random Forest reliability curve remains close to the diagonal across several probability regions. Its ECE is 0.0258, while HistGradientBoosting provides the lowest numerical ECE of 0.0176 in the final comparison.

These results support the use of calibrated probabilities for risk-aware resource allocation.

Calibration matters because the proposed resource-allocation layer uses predicted probabilities to rank clients by violation risk. A poorly calibrated model may overestimate or underestimate risk, leading to either unnecessary resource interventions or missed SLA degradation.

F. Best Model Interpretation

The results show that no single model is best across every criterion. Gradient Boosting is the best overall classifier by macro-F1 and accuracy [17]. Random Forest is the best violation-recall model and is therefore attractive for resource-protection use cases where missing an SLA violation is costly [16]. HistGradientBoosting is the best calibration model, which makes it useful when the controller depends heavily on risk thresholds [15]. Logistic Regression provides strong AUC and PR-AUC with low training cost, making it a valuable interpretable baseline [12].

From a deployment perspective, the final choice depends on the operator's objective. If the objective is balanced prediction, Gradient Boosting is the best choice. If the objective is to minimize missed SLA violations, Random Forest is preferable. If the objective is calibrated probability estimation for risk ranking, HistGradientBoosting is most suitable.

G. Risk-Aware Resource Allocation Implication

The predicted probability P(SLA OK) is transformed into violation risk using r = 1 − P(SLA OK). A network controller can then rank clients by risk and prioritize high-risk clients for resource attention. This does not replace the physical-layer scheduler; rather, it acts as a predictive decision-support layer that informs scheduling, slicing, or orchestration components.

The Random Forest model illustrates this trade-off clearly. It captures 1,774 out of 1,941 violation cases, missing only 167. However, it also flags 523 stable clients as violations. In safety- or SLA-critical settings, this may be acceptable because the cost of a missed violation may be greater than the cost of extra monitoring or temporary prioritization. In resource-constrained settings, a stricter risk threshold may be selected to reduce unnecessary interventions.

VII. Comparison With Prior Studies

Table VI compares the proposed study with representative directions in the literature. Existing works commonly focus on network slicing, intelligent resource allocation, federated learning, or general 6G network optimization [1]-[10]. The proposed work differs by using a public 6G telemetry dataset [11], predicting the next-window SLA label, preserving client-temporal order, reporting calibration [15], and connecting the prediction output to risk-aware resource prioritization.

Table VI. Comparison With Related Research Directions

Study / Direction	Main Focus	Data Type	Evaluation Style	Calibration Reported	Risk-Aware SLA Decision Layer	Difference From This Study
Machine learning for 6G resource allocation	Intelligent resource assignment and automation	Often simulation or conceptual	Varies	Usually not central	Often optimization-focused	This study uses public client-window telemetry and next-window SLA prediction
Network slicing resource management surveys	Slice orchestration, service isolation, multi-domain management	Survey / conceptual	Not experimental	No	Conceptual	This study provides reproducible model results and violation-focused metrics

Study / Direction	Main Focus	Data Type	Evaluation Style	Calibration Reported	Risk-Aware SLA Decision Layer	Difference From This Study
Federated/split learning for 6G networks	Distributed learning, privacy, edge intelligence	Framework-oriented	Broad review or scenario testing	Not primary	Not primary	This study focuses on centralized SLA-risk prediction and calibrated resource prioritization
AI-enabled 6G optimization surveys	AI for self-configuration and resource management	Survey / high-level	Not dataset-specific	Not primary	Conceptual	This study gives concrete telemetry-based evaluation with real tables and figures
Calibration studies in machine learning	Reliability of predicted probabilities	General ML benchmarks	Calibration-focused	Yes	No network allocation layer	This study applies calibration to 6G SLA-risk prediction
Proposed study	Next-window SLA prediction and proactive resource prioritization	Public 6G RAN telemetry, 30,000 client-window records	Client-aware chronological split	Yes, ECE and reliability diagrams	Yes	Combines telemetry learning, calibration, violation metrics, and resource-risk decisioning

The comparison shows that the proposed contribution is not merely another classifier benchmark. Its novelty lies in aligning the learning objective with a network-management action: predicting next-window SLA risk and using that risk for proactive resource attention. The inclusion of calibration and violation-focused metrics strengthens the operational relevance. This design is also aligned with recent audit-ready risk-screening work, where model output is not treated as a final autonomous decision but as a prioritized, explainable, and threshold-controlled support layer for limited operational capacity [23].

VIII. Limitations and Future Work

This study has several limitations. First, the proposed allocation policy is a decision-support layer rather than a complete physical-layer scheduler. It identifies clients that are likely to need additional resource attention but does not directly optimize PRB assignment, power control, or slice reconfiguration. Second, the dataset is telemetry-based [11], and the results depend on the representativeness of the telemetry generation or collection process. Third, sequence models were not included in the final comparison because of GPU-kernel incompatibility in the execution environment, although recurrent architectures remain relevant for temporal learning [19], [20]. Fourth, the current results use one main split protocol; future work can add repeated cross-client splits, cross-regime tests, and online adaptation. Therefore, the proposed framework should be interpreted as a risk-triage layer, similar to operational screening systems in which calibrated model scores support prioritization rather than replacing domain experts or downstream controllers [23].

Future research can extend the framework in several directions. Reinforcement learning can be used to convert predicted risk into continuous resource-allocation actions [8], [9]. Federated and regime-aware extensions are a natural next step, since prior 6G SLA-risk forecasting work shows that client drift, non-IID behavior, and unseen operating regimes can change the practical value

IEER

of SLA-risk predictors [21]. Uncertainty estimation and calibration methods can improve risk confidence [15]. Online learning can allow the model to adapt to traffic drift. Finally, the framework can be integrated with a network digital twin or RAN intelligent controller to evaluate closed-loop resource management.

IX. Conclusion

This paper proposed an SLA-aware predictive resource allocation framework for 6G radio access networks using client-temporal telemetry learning. The task was formulated as next-window SLA prediction using the public Wireless 6G Network Dataset of Resource Allocation [11]. A client-aware chronological split was used to preserve temporal realism, and lag, delta, and rolling-window features were created to capture short-term telemetry dynamics. Seven machine learning models were evaluated using predictive, calibration, and operational metrics [12]-[17].

The results show that Gradient Boosting achieved the best overall performance with 0.8585 accuracy and 0.8566 macro-F1. Random Forest achieved the strongest violation recall of 0.9140 and missed only 167 out of 1,941 actual violation cases. HistGradientBoosting achieved the best calibration with an ECE of 0.0176. These results demonstrate that telemetry-driven machine learning can support proactive SLA assurance in 6G RANs. By converting predicted SLA probability into violation risk, the proposed framework provides a practical resource-prioritization layer that can assist schedulers, slice managers, and network controllers.

REFERENCES

[1] S. Ali et al., “6G white paper on machine learning in wireless communication networks,” arXiv:2004.13875, 2020.

[2] Z. Yang, M. Chen, K.-K. Wong, H. V. Poor, and S. Cui, “Federated learning for 6G: Applications, challenges, and opportunities,” Engineering, vol. 8, pp. 33–41, 2022.

[3] S. Ebrahimi et al., “Resource management from single-domain 5G to end-to-end 6G network slicing: A survey,” IEEE Communications Surveys & Tutorials, 2024.

[4] A. A. Abba Ari et al., “IoT-5G and B5G/6G resource allocation and network slicing: A survey,” IET Networks, 2025.

[5] H. F. Alhashimi et al., “Survey on AI-enabled resource management for 6G networks,” Computer Networks, 2025.

[6] M. M. Islam et al., “Performance evaluation and optimization for 6G networks: A comprehensive review,” ICT Express, 2025.

[7] H. Hafi et al., “Split federated learning for 6G enabled-networks,” IEEE Access, 2024.

[8] A. Nouruzi et al., “Smart resource allocation model via artificial intelligence in software defined 6G networks,” arXiv:2302.04655, 2023.

[9] L. Ma, N. Cheng, X. Wang, R. Sun, and N. Lu, “On-demand resource management for 6G wireless networks using knowledge-assisted dynamic neural networks,” arXiv:2208.01785, 2022.

[10] Y. K. Tun, N. H. Tran, D. T. Ngo, S. R. Pandey, Z. Han, and C. S. Hong, “Wireless network slicing: Generalized Kelly mechanism based resource allocation,” arXiv:1907.02182, 2019.

[11] N. Maitlo, “Wireless 6G Network Dataset of Resource Allocation,” Kaggle Dataset, 2026.

[12] F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[13] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., 2016, pp. 785–794.

[14] G. Ke et al., “LightGBM: A highly efficient gradient boosting decision tree,” in Proc. Advances in Neural Information Processing Systems, 2017, pp. 3146–3154.

[15] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in Proc. International Conference on Machine Learning, 2017, pp. 1321–1330.

[16] L. Breiman, "Random forests," Machine Learning, vol. 45, pp. 5–32, 2001.

[17] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," Annals of Statistics, vol. 29, no. 5, pp. 1189–1232, 2001.

[18] C. M. Bishop, Pattern Recognition and Machine Learning. New York, NY, USA: Springer, 2006.

[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[20] K. Cho et al., "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in Proc. EMNLP, 2014, pp. 1724–1734.

[21] P. Mangi, S. Bibi, A. Nawaz, and S. Bibi, "When Clients Drift: Federated SLA-Risk Forecasting Across Unseen 6G RAN Regimes," Spectrum of Engineering Sciences, vol. 4, no. 4, pp. 1015–1023, 2026, doi: 10.5281/zenodo.19723844.

[22] B. Raza, A. Maitlo, Z. H. Shar, and I. Hyder, "Operational Android Malware Filtering: Calibrated Probabilities and Distribution-Free Guarantees," Kashf Journal of Multidisciplinary Research, vol. 2, no. 12, pp. 58–73, 2025, doi: 10.71146/kjmr778.

[23] I. Hyder, R. A. Shaikh, R. H. Arain, Z. Hussain, and B. Raza, "Audit-Ready Healthcare Fraud Screening: Split-Safe Provider Aggregation and Explainable Boosted Risk Triage," Southern Journal of Computer Science, vol. 2, no. 1, pp. 18–28, 2026.

IEER
Institute for Excellence in Education & Research