SCIENTIFIC CULTURE
www.sci-cult.com

SCIENTIFIC CULTURE, Vol. 12, No. 2.1, (2026), pp. 291-309
Open Access. Online & Print

DOI: 10.5281/zenodo.122.12625

INTELLIGENT CREDIT SENTINEL: A SYNERGISTIC MULTI-LAYERED MACHINE-LEARNING FRAMEWORK FOR DETECTING FRAUD AND BILLING ANOMALIES

Md Tuhin Rana[1], Shuvashish Roy[2], Ashim Sen Gupta[3], Rokhshana Parveen[4], Nadia Mehjabeen Oyshi[5], Dipankar Das[6], and Abhigyan Bhattacharjee[7]

[1]Student, Department of Statistics, University of Dhaka, Bangladesh
mdtuhin-2016913783@stat.du.ac.bd,
[2]Senior Researcher, Research & Innovation Division, Prime Bank PLC, Dhaka, Bangladesh,
shuvashishroy@gmail.com,
[3]First Assistant Vice President, International Division, Social Islami Bank PLC, Bangladesh,
asgcubd@gmail.com,
[4]Research Scholar, Dhaka, Bangladesh, rokhshana2006@gmail.com,
[5]Student, Department of Statistics, University of Dhaka, Bangladesh
nadiamehjabeen-2017413967@stat.du.ac.bd,
[6]Assistant Professor, Department of Commerce, University of Science & Technology, Meghalaya, Ri Bhoi, Meghalaya, India, dipankardas.dds@gmail.com
[7]Professor, Department of Management, North-Eastern Hill University, Tura Campus, India
abhigyan.nehu@gmail.com

Received: 01/12/2025
Accepted: 02/01/2026

Corresponding Author: Md Tuhin Rana
(mdtuhin-2016913783@stat.du.ac.bd)

ABSTRACT

Traditional monolithic systems struggle against the dynamic nature of financial fraud and other transactional risks. This paper introduces the Intelligent Credit Sentinel, a novel four-layered hierarchical architecture designed for robust and multi-faceted transaction risk assessment. The system employs an unsupervised autoencoder for broad anomaly detection (Layer 1), two specialized supervised XGBoost classifiers for targeted fraud and billing error detection (Layers 2 & 3), and a final logistic regression meta-learner to synthesize these outputs into a single, actionable risk score (Layer 4). Through extensive feature engineering and hyperparameter tuning to manage severe class imbalance, the specialist layers achieved high performance, with the final billing error model attaining 94% precision. The synergistic combination of these layers in the meta-learner resulted in a final system-wide recall of 82.4% for all high-risk events. The findings demonstrate that this modular, tiered approach is a highly effective, interpretable, and operationally efficient paradigm for modern financial security.

KEYWORDS: Fraud Detection, Multi-Layered Architecture, Meta-Learner, Anomaly Detection, XGBoost, Class Imbalance.

1. INTRODUCTION

In the ever-expanding landscape of digital finance, the integrity of electronic transactions is paramount (Goldstein & Uchida, 2016; Nata et al., 2025). Yet, with each technological advance that simplifies commerce, a shadow-self emerges in the form of increasingly sophisticated financial fraud (Mienye& Sun, 2023; . Credit card fraud, in particular, represents a multi-billion dollar challenge, evolving from simple theft to complex schemes that exploit both technical vulnerabilities and human behavior (Mienye& Sun, 2023; (Niu et al., 2019).

Traditional fraud detection systems, often relying on monolithic models, have been shown in the literature to struggle with this dynamic threat (Niu et al., 2019; (Jeyaraj et al., 2024). While these systems can be effective at identifying known fraud patterns, research indicates they often fall short when faced with novel attack vectors or entirely different classes of transactional issues (Niu et al., 2019; (Jeyaraj et al., 2024). Intricate billing errors, for example, represent a separate class of problem that can erode consumer trust and incur significant operational costs, yet are not the primary target of traditional fraud-focused models ("Machine Learning for Identifying Fraud in Credit Card Transactions", 2024).

This paper addresses the critical need for a more resilient and multi-faceted approach to transaction security (Jeyaraj et al., 2024). We posit that to address these documented limitations, a single, all-encompassing model is no longer sufficient (Niu et al., 2019; (Jeyaraj et al., 2024). Instead, we propose a hierarchical system of specialized "experts," each trained for a specific task, to provide a more robust and intelligent defense (Jeyaraj et al., 2024)(Rihan et al., 2023). To this end, we introduce the Intelligent Credit Sentinel, a novel four-layered architecture designed to dissect and analyze transaction risk from multiple perspectives (Rihan et al., 2023).

The proposed system begins with a broad, unsupervised Layer 1, an autoencoder-based anomaly screener that acts as a vigilant gatekeeper, identifying any transaction that deviates from the norm without prior knowledge of specific threats (Goldstein & Uchida, 2016; Nata et al., 2025). This is followed by two parallel, supervised specialist models: a Layer 2 XGBoost classifier, meticulously trained to identify the complex signatures of known fraudulent activities (Jeyaraj et al., 2024), and a highly-tuned Layer 3 XGBoost model, designed to detect the more subtle and ambiguous patterns of billing errors ("Machine Learning for Identifying Fraud in Credit Card Transactions", 2024). The intelligence of these disparate layers is then synthesized by a Layer 4 meta-learner, a final decision engine that weighs the evidence from each expert to produce a single, actionable risk assessment (Rihan et al., 2023).

This paper details the design, implementation, and rigorous evaluation of each layer of the Intelligent Credit Sentinel. The methodology and results for the unsupervised anomaly detector are presented, followed by the development and performance of the supervised classifiers for fraud and billing errors (Jeyaraj et al., 2024). Finally, the paper describes the meta-learner and evaluates the performance of the system as a whole (Rihan et al., 2023). Through this multi-layered approach, we demonstrate a system that is not only highly accurate but also interpretable and operationally efficient (- et al., 2025), offering a more comprehensive paradigm for modern transaction security (Jeyaraj et al., 2024).

2. LITERATURE REVIEW

Traditional fraud detection, which relies on static, manually-defined rules (Malik et al., 2022), struggles with significant limitations. These systems are inherently reactive rather than proactive, lack flexibility, and are time-consuming to maintain (Malik et al., 2022). Rule-based approaches employ binary features that flag transactions based on predetermined thresholds, such as geographic anomalies (Boulieris et al., 2023), yet these static rules are easily circumvented by evolving fraud techniques (Pk, 2023). In contrast, machine learning (ML) models represent a fundamental shift, enabling systems to adaptively learn complex patterns from large datasets (Pk, 2023). ML algorithms can efficiently analyze massive transaction volumes to identify nuanced patterns that conventional techniques overlook (Pk, 2023). Deep learning and advanced ML architectures, in particular, provide superior performance by automatically discovering discriminative features and adapting to emerging fraud schemes (Alarfaj et al., 2022). This transition from static, expert-defined rules to data-driven, adaptive ML models addresses the core weaknesses of traditional systems, offering a proactive and scalable solution for detecting sophisticated, evolving fraudulent activities (Malik et al., 2022).

Unsupervised learning approaches, particularly autoencoders, have emerged as powerful tools for anomaly detection in financial systems without requiring labeled fraud data (Jiang et al., 2023). Autoencoders learn to reconstruct normal transaction patterns by minimizing reconstruction error, enabling them to identify significant deviations from this learned normality as anomalies (Jiang et al., 2023; Wu & Wang, 2021). This unsupervised paradigm is valuable because it circumvents the challenge of obtaining comprehensive labeled datasets and can adapt to unknown attack patterns (Jiang et al., 2023). Hybrid architectures combining autoencoders with generative adversarial networks (GANs) may further enhance these detection capabilities (Wu & Wang, 2021). In the financial domain, unsupervised autoencoders serve as an effective first line of defense, flagging generally abnormal transactions for subsequent analysis (Wu & Wang, 2021). By treating fraudulent transactions as anomalous deviations, these models can identify suspicious activities without prior knowledge of specific fraud schemes (Jiang et al., 2023). This addresses critical limitations of purely supervised methods, offering a scalable, adaptive solution for detecting novel fraudulent patterns (Jiang et al., 2023; Wu & Wang, 2021).

In parallel, supervised learning models like XGBoost and Random Forests have demonstrated effective performance in credit card fraud detection by leveraging labeled data to learn specific fraud patterns (Niu et al., 2019). These ensemble methods excel at capturing the complex, non-linear relationships between transaction features and predefined risk categories (Niu et al., 2019). Both XGBoost and Random Forests achieve high accuracy through their ability to iteratively refine predictions and handle feature interactions that simpler models may overlook (Niu et al., 2019). Comparative studies indicate that supervised models generally outperform unsupervised approaches when sufficient labeled data is available, as they are directly optimized to distinguish between normal and abnormal transactions (Niu et al., 2019). The strength of these methods lies in their capacity to learn patterns specific to known fraud types, enabling precise classification (Niu et al., 2019). However, this strength is also their primary limitation, as they require substantial labeled datasets and may fail to identify novel, previously unseen fraud schemes (Niu et al., 2019). In multi-layered frameworks, these classifiers can thus function as specialized secondary layers, building upon an initial anomaly detection phase to provide targeted, high-confidence identification of specific, predefined risks (Niu et al., 2019).

A critical challenge in this domain is the severe class imbalance inherent in financial datasets, where fraudulent transactions constitute a small fraction of total activity. This imbalance causes standard models to exhibit a strong bias toward the majority (legitimate) class (Youssef, 2025; Salekshahrezaee et al., 2023). This bias degrades model performance, particularly in detecting the minority fraud class, which is the primary objective of the system (Salekshahrezaee et al., 2023). The literature identifies two complementary approaches to this problem: data-level and algorithm-level techniques (Salekshahrezaee et al., 2023). Data-level methods modify the class distribution through random undersampling of the majority class or oversampling techniques such as SMOTE (Synthetic Minority Oversampling Technique) and its variants (Youssef, 2025; Salekshahrezaee et al., 2023). Algorithm-level approaches, by contrast, employ cost-sensitive learning or class weighting strategies that apply a higher penalty for misclassifying the minority fraud class during model training (Salekshahrezaee et al., 2023). Studies demonstrate that combining these techniques with ensemble classifiers like XGBoost significantly improves fraud detection performance (Youssef, 2025; Hájek et al., 2022). By strategically addressing class imbalance, robust systems can be developed that effectively identify rare fraudulent transactions (Youssef, 2025; Salekshahrezaee et al., 2023).

To overcome the limitations of any single approach, hybrid and meta-learning architectures have emerged as a powerful paradigm for synthesizing outputs from multiple specialist models (Airlangga, 2024). Stacking ensemble techniques combine diverse base learners—such as Random Forest, XGBoost, and Support Vector Machines (SVM)—through a meta-learner that optimally integrates their predictions (Airlangga, 2024). This "stacking" approach leverages the complementary strengths of different algorithms: unsupervised methods excel at detecting novel anomalies, while supervised classifiers focus on identifying known fraud patterns (Airlangga, 2024). The meta-learner, often a simpler model like logistic regression, learns to weigh the evidence from these base models to produce a unified, superior risk score (Airlangga, 2024). Empirical studies demonstrate that stacked ensembles can substantially improve fraud detection accuracy by effectively mitigating individual model weaknesses (Airlangga, 2024). By integrating unsupervised anomaly detection with supervised classifiers, these hybrid architectures create robust, multi-layered defense systems capable of identifying both known and unknown threats (Airlangga, 2024).

The existing literature, therefore, reveals a critical gap. Research predominantly addresses either general anomaly detection or singular, binary fraud classification tasks (Airlangga, 2024). While many studies employ ensemble methods and advanced algorithms, there is a lack of comprehensive frameworks designed to simultaneously distinguish between multiple distinct risk types—such as targeted fraud versus billing errors—within a single, unified system. Most research focuses on a binary (fraud/legitimate) classification without addressing the practical, operational need to differentiate between specific, actionable risk categories (Hájek et al., 2022). Few studies propose multi-layered hybrid architectures that combine unsupervised screening with specialized supervised experts for distinct risk types, all synthesized by a meta-learner. This paper's proposed four-layer system fills this gap by providing a comprehensive, operationally-aligned framework that advances beyond existing single-objective or general anomaly detection approaches.

3. OBJECTIVE OF THE STUDY

The primary objective of this paper is to design and develop a multifaceted hierarchical model for credit card fraud detection, recognizing that reliance on a single detection method is insufficient in today’s complex financial environment. The proposed approach aims to integrate multiple detection techniques within a layered framework to enhance accuracy, reduce falsepositives and improve adaptability against evolving fraud strategies.

4. METHODOLOGY

4.1. Variables

The multi-layered architecture of the system necessitated a distinct set of variables for each analytical stage, tailored to the specific objective of each model. The features were engineered from core transaction data, historical cardholder and merchant profiles, and the outputs of preceding layers.

The primary input variables can be categorized as follows

The multi-layered architecture of the system necessitated a distinct set of variables for each analytical stage, tailored to the specific objective of each model. The features were engineered from core transaction data, historical cardholder and merchant profiles, and the outputs of preceding layers.

The primary input variables can be categorized as follows

These features provide the fundamental context of each transaction. They include the *Transaction_Amount_Local_Currency*, *Merchant_Category_Code* (MCC), *Point_of_Sale_Entry_Mode*, and boolean flags indicating if the transaction was *Is_Card_Present* or an *Is_Cross_Border_Transaction*. For Card-Not-Present (CNP) transactions, *AVS_Response_Code* and *CVV_Match_Result* were also incorporated.

Cardholder and Merchant Profiles To contextualize the transaction within broader patterns, variables representing the cardholder and merchant were used. Cardholder features included *Credit_Limit*, *Reported_Fraud_History_Count*, *Billing_Dispute_History_Count*, and a pre-assigned *Persona_Type*. Merchant attributes were represented by *Merchant_Risk_Level* and *Historical_Billing_Dispute_Rate_Global*.

Cardholder Historical Behavior Baselines To establish a behavioral baseline for each cardholder, several features were calculated based on their historical activity. These included statistical measures of their typical transaction amount (*CH_Avg_Amount*, *CH_Median_Amount*, *CH_StdDev_Amount*) and a Z-score (*CH_Transaction_Amount_ZScore*) to normalize the current transaction amount against their history. Behavioral frequency was captured by *CH_Count_Transactions_per_Day* and *CH_Frequency_MCC_Usage*.

Engineered Temporal and Velocity Features Time was a critical dimension, addressed through several engineered features. The *Transaction_Hour* was transformed into cyclical features (*hour_sin*, *hour_cos*) to preserve the continuity of the 24-hour cycle. Recency was captured via *Time_Since_CH_Last_Transaction_Overall_Min* and *Time_Since_CH_Last_Transaction_at_Same_Merchant_Min*. To model short-term spending velocity, a series of features were calculated over rolling time windows of 1, 6, 24, and 168 hours, including transaction counts (*CH_Count_Transactions_Last_X_Hours*), transaction sums (*CH_Sum_Amount_Transactions_Last_X_Hours*), and the count of unique merchants (*CH_Count_Unique_Merchants_Last_X_Hours*). The *Transaction_DayOfWeek* was also included as a categorical variable.

Inter-Layer Features The Layer 4 meta-learner utilized the probabilistic outputs from the preceding layers as its primary inputs. These variables were the *Layer1_Reconstruction_Error* from the autoencoder, the *Layer2_Fraud_Probability* from the fraud detection model, and the *Layer3_Billing_Error_Probability* from the billing anomaly detector. The original *Transaction_Amount_Local_Currency* was also passed to this final layer to provide context.

Target Variables The dependent variables for the supervised models were *Is_Fraud* for the Layer 2 fraud estimator and *Is_Billing_Error* for the Layer 3 billing anomaly detector. A secondary model in

Layer 3 also predicted the *Billing_Error_Type*. The Layer 4 meta-learner was trained on a composite binary target, *Meta_Target_High_Risk*, which was positive if a transaction was classified as either fraud or a billing error.

Initial Preprocessing To ensure the model was evaluated in a realistic scenario, the dataset was first sorted chronologically by Timestamp. A time-based 80/20 train-test split was then performed, training the models on the first 80% of the data and evaluating them on the most recent 20%.

For use in each layer's specific ColumnTransformer, standardized preprocessing pipelines were created based on the variable types.

Numerical Features Processed by first imputing missing values with the median, followed by scaling with StandardScaler.

Categorical Features: Processed by imputing missing values with a constant 'Missing' string (treating non-response as a distinct category) and then transforming them using OneHotEncoder.

4.2. Layer 1: Unsupervised Anomaly Screening via Deep Autoencoder

The first layer of the proposed Inteligent Credit Sentinel system is an unsupervised anomaly screener designed to identify transactions that deviate from established patterns of normal behavior. For this task, we employ a deep autoencoder, a neural network trained to reconstruct its own input. The core principle is that a model trained exclusively on legitimate transactions will exhibit a significantly higher reconstruction error when presented with an anomalous or fraudulent transaction it has not seen before. This reconstruction error serves as a valuable anomaly score.

Input Vector Representation Each transaction is formally represented as a feature vector $\mathbf{x} \in \mathbb{R}^D$, where D is the total number of raw features. This vector is a concatenation of numerical and categorical feature subsets, $\mathbf{x} = [\mathbf{x}_{num}, \mathbf{x}_{cat}]$.

Based on the implemented feature selection, the numerical feature vector, $\mathbf{x}_{num}$, includes transactional attributes such as *Transaction_Amount_Local_Currency*, cardholder spending patterns like *CH_Avg_Amount* and *CH_Transaction_Amount_ZScore*, and cyclical time-based features *hour_sin* and *hour_cos*. The categorical feature vector, $\mathbf{x}_{cat}$, includes *Merchant_Category_Code*, *Point_of_Sale_Entry_Mode*, and *Persona_Type*, among others.

Data Pre-processing Prior to model training, a preprocessing function, denoted as $\Phi(\cdot)$, is applied to transform the raw input vector $\mathbf{x}$ into a scaled and encoded vector $\mathbf{x}'$. This function consists of two main operations:

- **Min-Max Scaling** Applied to the numerical features $\mathbf{x}_{num}$, this operation scales each feature to a range of $[0,1]$.
- **One-Hot Encoding** Applied to the categorical features $\mathbf{x}_{cat}$, this operation converts each categorical variable into a binary vector representation.

The resulting preprocessed vector, $\mathbf{x}' = \Phi(\mathbf{x})$, has a dimensionality of d, where d corresponds to the *input_dim* variable in the implementation.

Autoencoder Architecture The autoencoder architecture consists of two primary components: an encoder and a decoder.

The **encoder**, denoted by the function f, maps the preprocessed input vector $\mathbf{x}'$ to a lower-dimensional latent space representation $\mathbf{z} \in \mathbb{R}^k$, where $k < d$. This is formally expressed as:

$$\mathrm{z} = f(\mathrm{x}')$$

The encoder is composed of a series of L dense layers, where the output of each layer i is given by:

$$\mathrm{h}_i = \sigma_{relu}(\mathrm{W}_i \mathrm{h}_{i-1} + \mathrm{b}_i) \quad \text{for } i = 1, \dots, L$$

Here, $\mathrm{h}_0 = \mathrm{x}'$, W_i and b_i are the weight matrix and bias vector for layer i respectively, and $\sigma_{relu}(a) = \max(0, a)$ is the Rectified Linear Unit (ReLU) activation function. The latent vector is the output of the final encoder layer, $\mathrm{z} = \mathrm{h}_L$.

The decoder, denoted by the function g, mirrors the encoder's architecture. It attempts to reconstruct the original input vector from the latent representation z, producing a reconstructed vector $\hat{\mathrm{x}}'$:

$$\hat{\mathrm{x}}' = g(\mathrm{z}) = g\big(f(\mathrm{x}')\big)$$

The decoder is composed of dense layers with ReLU activations, culminating in a final output layer with a sigmoid activation function, $\sigma_{sig}(a) = (1 + e^{-a})^{-1}$, to ensure the output values are constrained to the $[0,1]$ range, matching the Min-Max scaled input.

Objective Function and Anomaly Score The autoencoder is trained by minimizing an objective (loss) function, $\mathcal{L}$, which measures the dissimilarity between the original and reconstructed vectors. We employ the Mean Squared Error (MSE), defined as the squared Euclidean norm (L_2-norm) of the difference vector. For a set of N normal training samples $\{\mathbf{x}'_1, \dots, \mathbf{x}'_N\}$, the loss is:

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \left| \; |\mathbf{x}'_i - \hat{\mathbf{x}}'_i| \; \right|_2^2$$
$$= \frac{1}{N}\sum_{i=1}^{N} \left| \; |\mathbf{x}'_i - g\big(f(\mathbf{x}'_i)\big)| \; \right|_2^2$$

where θ represents all trainable model parameters (weights and biases).

After training on the normal transaction dataset *normal_train_df_l1*, the model's parameters are fixed.

For any new transaction, $\mathbf{x}_{new}$, its anomaly score, $E(\mathbf{x}_{new})$, is calculated as its reconstruction error:

$$E(\mathbf{x}_{new}) = |\ \ |\Phi(\mathbf{x}_{new}) - g\left(f\left(\Phi(\mathbf{x}_{new})\right)\right)|\ \ |_2^2$$

A transaction is flagged as an anomaly if its score E exceeds a predetermined threshold, τ.

4.3. Layer 2: Supervised Fraud Likelihood Estimation

While the Layer 1 autoencoder serves as a general anomaly screener, Layer 2 employs a supervised learning approach to specifically model the complex, non-linear patterns indicative of known fraudulent activities. For this task, we utilize XGBoost, a highly efficient and scalable implementation of Gradient Boosting Decision Trees (GBDT), to produce a probabilistic fraud likelihood score for each transaction.

Model Formulation The XGBoost model constructs an ensemble of K regression trees. The final prediction for a given input feature vector $\mathbf{x}_i$ is the sum of the predictions from each individual tree, passed through a logistic function to produce a probability. Let f_k represent the k-th tree in the ensemble; the raw prediction score $\hat{y}_i$ is given by:

$$\hat{y}_i = \sum_{k=1}^{K} f_k\,(\mathbf{x'}_i)$$

where $\mathbf{x'}_i$ is the preprocessed feature vector for transaction i, and K is the total number of trees, corresponding to the *n_estimators* parameter. The final fraud probability is then $P(\text{Fraud}|\mathbf{x'}_i) = \sigma(\hat{y}_i)$, where $\sigma(\cdot)$ is the logistic function.

The trees are built in an additive manner. At each iteration t, a new tree f_t is trained to minimize the overall objective function, which includes a loss term and a regularization term:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x'}_i)\right) + \Omega(f_t)$$

Here, $l(y_i, \hat{y}_i)$ is the loss function measuring the error between the true label $y_i \in \{0,1\}$ and the prediction $\hat{y}_i$. The regularization term, $\Omega(f_t)$, penalizes the complexity of the newly added tree to prevent overfitting.

Handling Class Imbalance The dataset exhibits a severe class imbalance, with fraudulent transactions (*Is_Fraud* = 1) representing a small minority of the data. To address this, we introduce a weighting mechanism directly into the loss function. A weight, W_{pos}, is applied to the loss for all positive class instances. This weight corresponds to the *scale_pos_weight* parameter in the XGBoost implementation and is calculated as the ratio of the number of negative class samples to positive class samples:

$$W_{pos} = \frac{\text{count}(y = 0)}{\text{count}(y = 1)}$$

The objective function is thus modified to place a significantly higher penalty on misclassifying a fraudulent transaction compared to a legitimate one.

Feature Space The feature vector $\mathbf{x}$ for Layer 2 is more comprehensive than that of Layer 1. It incorporates a wide range of variables from *layer2_numerical_features* and *layer2_categorical_features*. These include not only transactional data but also features describing the cardholder's history (*Reported_Fraud_History_Count*), merchant risk profiles (*Historical_Fraud_Rate_Global*), real-time velocity checks (*CH_Count_Transactions_Last_1H*), and explicit transaction verification results (*AVS_Response_Code*, *CVV_Match_Result*). Crucially, the anomaly score *Layer1_Reconstruction_Error* from the first layer is also included as a feature, allowing this supervised model to leverage the insights from the unsupervised screener. All features are preprocessed using the *preprocessor_l2* pipeline, which applies *StandardScaler* to numerical features and *OneHotEncoder* to categorical ones.

4.4. Layer 3: Billing Anomaly Detection and Classification

The third layer of the system is a specialized two-stage component designed to identify and categorize specific types of non-fraudulent, high-risk events, namely billing errors. This layer operates independently of the fraud detector to capture distinct patterns associated with operational discrepancies.

Billing Error Detection with XGBoost The first stage employs a supervised binary classifier to determine the probability that a given transaction is a billing error. The model selected for this task is XGBoost.

Model Formulation The XGBoost model constructs an ensemble of K decision trees, where K is the *n_estimators* parameter. The final raw prediction score $\hat{y}_i$ for a preprocessed input vector $\mathbf{x'}_i$ is the sum of the scores from each individual tree, f_k:

$$\hat{y}_i = \sum_{k=1}^{K} f_k\,(\mathbf{x'}_i)$$

The model is trained in an additive fashion, where each new tree f_t is trained to minimize the objective function, which balances a loss term and a regularization term Ω to control model complexity:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x'}_i)\right) + \Omega(f_t)$$

where $l(y_i, \hat{y}_i)$ is the logistic loss function for binary classification.

Handling Class Imbalance To address the severe class imbalance of billing errors, a weighting parameter W_{pos} (*scale_pos_weight*) is incorporated into the loss function. This applies a greater penalty for misclassifying the minority (positive) class. Based on an automated hyperparameter search, the optimal value was determined to be:

$$W_{pos} = 2.82$$

Billing Error Type Classification with Random Forest For transactions flagged as billing errors by the first stage, a second multi-class classifier is used to categorize the specific *Billing_Error_Type*. A Random Forest model was implemented for this purpose.

Model Formulation A Random Forest is an ensemble of K individual decision trees. Each tree is trained on a random bootstrap sample of the data. The final prediction for a transaction $\mathbf{x'}_i$ is determined by a majority vote among all trees in the forest. Let C be the set of possible error types (e.g., *Duplicate_Charge*, *Unwanted_Subscription_Renewal*). The predicted class $\hat{y}_i$ is the one that receives the most votes:

$$\hat{y}_i = \underset{c \in C}{\operatorname{argmax}} \sum_{k=1}^{K} \mathbb{I}\left(f_k(\mathbf{x'}_i) = c\right)$$

where $f_k(\mathbf{x'}_i)$ is the prediction of the k-th tree and $\mathbb{I}(\cdot)$ is the indicator function, which is 1 if the condition is true and 0 otherwise.

Handling Class Imbalance To manage potential imbalances between different error types, the Random Forest is configured with *class_weight*='*balanced*'. This setting automatically assigns a weight w_c to each class c in the training process, which is inversely proportional to its frequency. The weight is calculated as:

$$w_c = \frac{n_{\text{samples}}}{n_{\text{classes}} \times n_{\text{samples},c}}$$

where n_{samples} is the total number of training samples, n_{classes} is the number of distinct error types, and $n_{\text{samples},c}$ is the number of samples belonging to class c. This ensures the model gives equal importance to all error types, regardless of their prevalence.

4.5. Layer 4: Meta-Learner for Final Risk Assessment

The final layer of the *Inteligent Credit Sentinel* system is a decision and action engine, implemented as a meta-learner. The purpose of this layer is to synthesize the specialized outputs from the preceding three layers into a single, unified probability score that represents the overall risk of a transaction. This final score is then used to recommend concrete actions, such as approving, reviewing, or declining a transaction.

Input Features and Unified Target The meta-learner does not operate on the raw transaction data. Instead, its feature space is constructed from the outputs of the previous layers. The input feature vector for the meta-learner, denoted as $\mathbf{x}_{\text{meta}}$, is composed of:

- The anomaly score from Layer 1 ($E(\mathbf{x})$, *Layer1_Reconstruction_Error*).
- The fraud probability from Layer 2 ($P(\text{Fraud}|\mathbf{x}')$, *Layer2_Fraud_Probability*).
- The billing error probability from Layer 3 ($P(\text{Billing Error}|\mathbf{x}')$, *Layer3_Billing_Error_Probability*).
- The original *Transaction_Amount_Local_Currency* is also included to provide context on the financial magnitude of the event.

To train this model, a unified binary target variable, y_{meta} (*Meta_Target_High_Risk*), is created. A transaction is considered a high-risk event ($y_{\text{meta}} = 1$) if it is either a confirmed fraud or a confirmed billing error. Formally:

$$y_{\text{meta}} = (y_{\text{fraud}} = 1) \vee (y_{\text{billing\_error}} = 1)$$

Model Formulation A Logistic Regression model is employed as the meta-learner due to its interpretability and efficiency. The model learns a set of coefficients, $\mathbf{w}$, and a bias term, b, to map the input features to a final risk probability. The probability of a transaction being a high-risk event is modeled using the logistic (sigmoid) function $\sigma(\cdot)$:

$$P(y_{\text{meta}} = 1|\mathbf{x'}_{\text{meta}}) = \sigma(\mathbf{w}^T\mathbf{x'}_{\text{meta}} + b) = \frac{1}{1 + e^{-(\mathbf{w}^T\mathbf{x'}_{\text{meta}} + b)}}$$

where $\mathbf{x'}_{\text{meta}}$ is the scaled meta-feature vector. The learned coefficients in $\mathbf{w}$ directly correspond to the importance the meta-learner places on the output of each preceding layer when making its final decision.

Handling Class Imbalance The unified target variable, y_{meta}, is also imbalanced, as high-risk events are rare. To counteract this, the Logistic Regression model is configured with *class_weight*='*balanced*'. This setting adjusts the loss function by applying a weight, w_c, to each class c that is inversely proportional to its frequency:

$$w_c = \frac{n_{\text{samples}}}{n_{\text{classes}} \times n_{\text{samples},c}}$$

where n_{samples} is the total number of samples, n_{classes} is the number of classes (two in this case), and $n_{\text{samples},c}$ is the number of samples in class c. This ensures that the model does not become biased towards the majority “Not High Risk” class.

5. RESULTS

5.1. Layer 1: Results of Unsupervised Anomaly Screening

The performance of the Layer 1 autoencoder as a broad-based anomaly screener was evaluated against two distinct types of adverse events: actual fraud and billing errors. The results, presented in Figure 1, demonstrate the model's effectiveness in identifying transactions that deviate from normative patterns, particularly those associated with fraudulent activity.

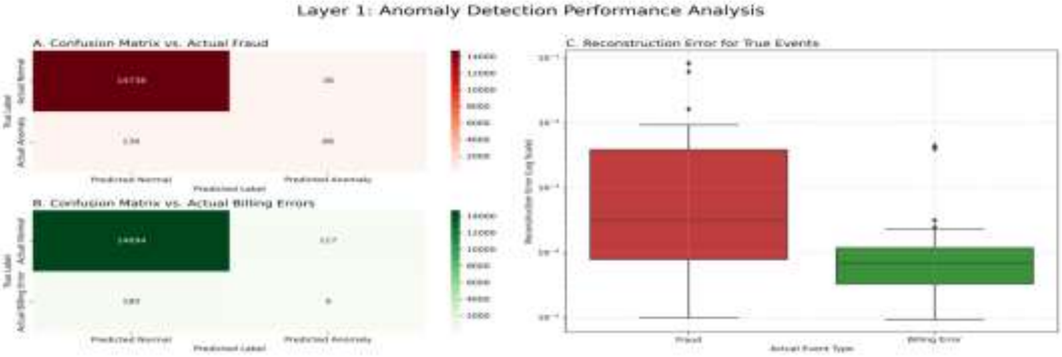

Figure 1: Anomaly Detection Performance Analysis of Layer 1.

Figure 1A and 1B present the confusion matrices for the model's anomaly predictions against the ground truth for fraud and billing errors, respectively. When evaluated against actual fraud (Figure 1A), the model successfully identified 88 fraudulent transactions (True Positives) while failing to flag 136 (False Negatives). This corresponds to a recall of approximately 39% for fraudulent events. The model generated 38 False Positives by flagging legitimate transactions as anomalous.

Conversely, the model was significantly less effective at identifying billing errors (Figure 1B). It correctly flagged only 9 such events while missing 180, indicating that the patterns characterizing billing errors are less distinct from normal transactional behavior and are not well-captured by the autoencoder's learned representation of "normality."

Figure 1C provides a comparative visualization of the reconstruction error distributions for true fraudulent events versus true billing errors. A distinct difference is observable: the median and interquartile range of reconstruction errors for fraudulent transactions are substantially higher than those for billing errors. This result strongly indicates that the autoencoder's anomaly signal, quantified by the reconstruction error E(x), is a more potent indicator for fraud than for billing discrepancies. The model correctly perceives fraudulent activities as more significant deviations from the learned norm, thereby assigning them higher anomaly scores.

Layer 1: Visualizing the Reconstruction Error Distribution To visualize the separability of normal and anomalous transactions, the distribution of the autoencoder's reconstruction error, E(x), was plotted on both a linear and a logarithmic scale (Figure 2). This comparison is critical due to the severe class imbalance inherent in the dataset.

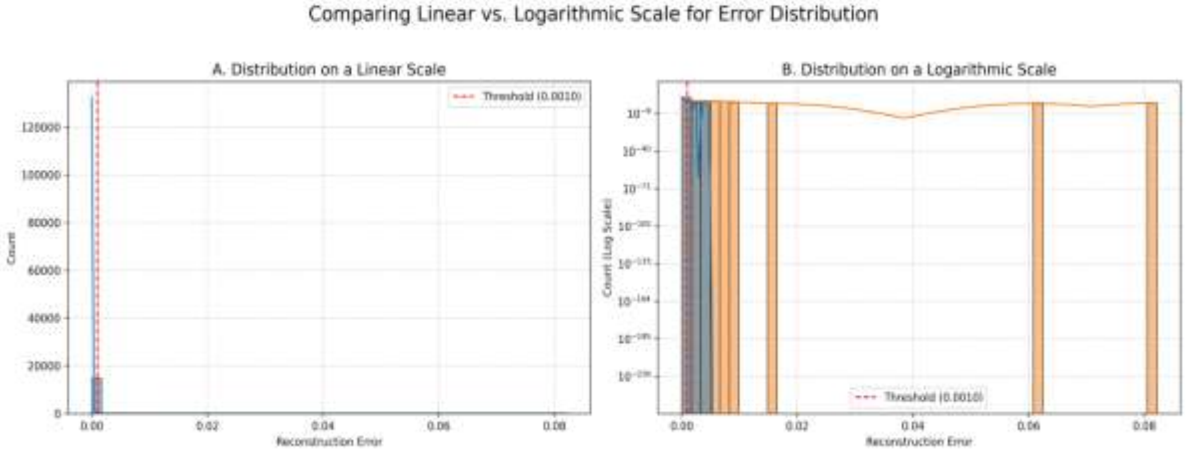

Figure 2: Comparing Linear vs Logarithmic Scale for Error Distribution.

Figure 2A, which uses a standard linear scale for the y-axis (Count), effectively illustrates that the vast majority of transactions are normal, exhibiting a reconstruction error very close to zero. However, this representation completely obscures the distribution of the anomalous class, as their count is orders of magnitude smaller than that of the normal class, rendering them invisible on this scale.

In contrast, Figure 2B utilizes a logarithmic scale for the y-axis. This transformation compresses the high counts of the normal transactions, thereby making the distribution of the far less frequent anomalous transactions clearly visible. This view is essential as it confirms that transactions with higher reconstruction errors are predominantly fraudulent events. The logarithmic plot demonstrates a discernible, albeit overlapping, separation between the error distributions of the two classes, reinforcing the utility of the reconstruction error as a valid anomaly signal. The optimal threshold, derived from the Precision-Recall curve, is shown to effectively partition these two distributions.

Layer 1: Quantitative Performance Evaluation To quantitatively assess the performance of the autoencoder as a fraud detector, the Precision-Recall (PR) and Receiver Operating Characteristic (ROC) curves were generated, as shown in Figure 3. These metrics are essential for evaluating classifier performance on imbalanced datasets.

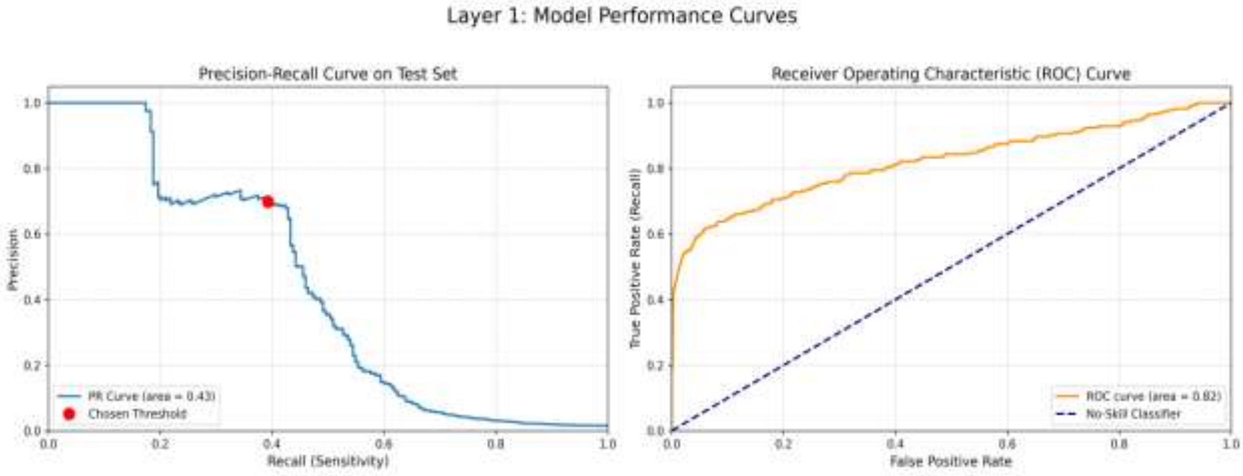

Figure 3: Model Performance Curves.

The Precision-Recall curve (Figure 3, Left) illustrates the trade-off between the model's precision (the fraction of flagged anomalies that are actual frauds) and its recall (the fraction of actual frauds that are correctly flagged). The Area Under the Curve (PR AUC) is 0.43. A random or no-skill classifier would achieve a PR AUC equivalent to the prevalence of the positive class in the dataset (a value significantly less than 0.43), indicating that the model's performance is substantially better than baseline. The "Chosen Threshold," marked in red, represents the operating point selected to balance precision and recall, achieving a recall of approximately 0.39 while maintaining a precision of around 0.70.

The ROC curve (Figure 3, Right) plots the True Positive Rate (Recall) against the False Positive Rate across all possible thresholds. The Area Under the ROC Curve (ROC AUC) is 0.82. A score of 0.5 represents a no-skill classifier, while a score of 1.0 represents a perfect classifier. The achieved AUC of 0.82 signifies a strong discriminative ability, indicating that there is an 82% probability that the model will rank a randomly chosen fraudulent transaction with a higher reconstruction error than a randomly chosen legitimate transaction.

Layer 1: Visualizing the Reconstruction Mechanism To provide a qualitative and intuitive understanding of the autoencoder's behavior, the original and reconstructed feature vectors were visualized in a two-dimensional feature space. Figure 4 illustrates this comparison for two representative scaled features: *Transaction_Amount_Local_Currency* and *CH_Avg_Amount*.

The left panel of Figure 4 displays the results for normal transactions. The green points, representing the reconstructed data (x ˆ), form a dense cloud that almost perfectly overlaps with the blue points, representing the original preprocessed data (x'). This tight correspondence visually confirms that the model has effectively learned the underlying manifold of normal data, resulting in a low reconstruction error for legitimate transactions.

In contrast, the right panel displays the results for fraudulent transactions. A significant divergence is evident between the original fraudulent data points (blue) and their reconstructed counterparts (red). The reconstructed points are often displaced from their original locations, indicating the model's inability to accurately reproduce these anomalous inputs. This displacement visually represents a high reconstruction error, E(x), which is the fundamental signal used by this layer to flag transactions as potential anomalies. This visualization provides a clear, mechanistic validation of the autoencoder's utility for this task.

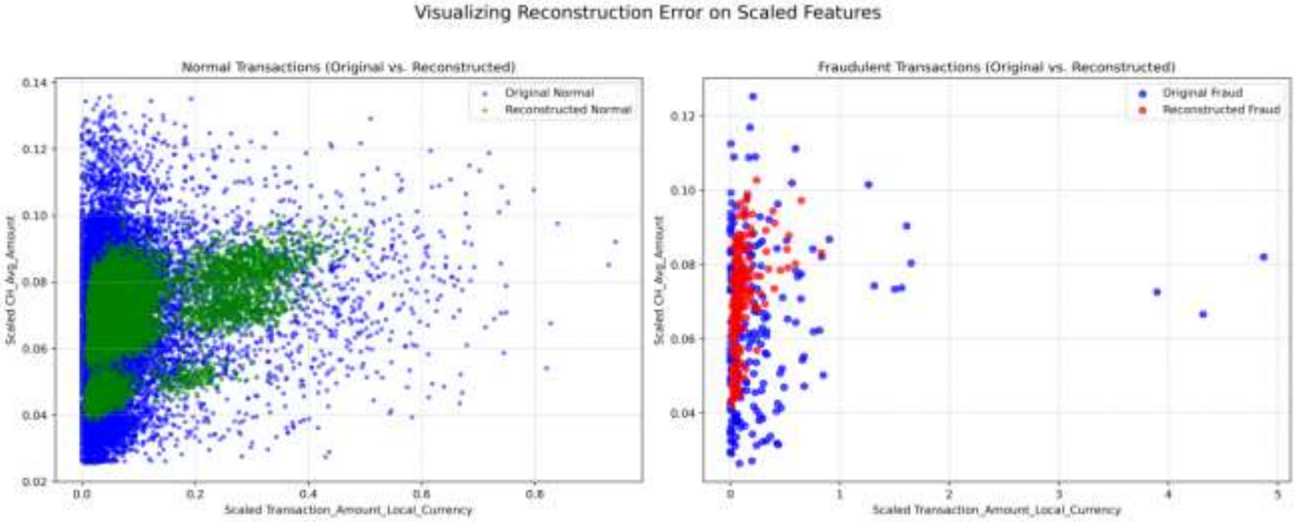

Figure 4: Visualizing Reconstruction Error on Scaled Features.

5.2. Layer 2: Results of Supervised Fraud Classification

The performance of the Layer 2 XGBoost classifier was evaluated using Precision-Recall (PR) and Receiver Operating Characteristic (ROC) curves, presented in Figure 5. These metrics assess the model's ability to effectively distinguish between legitimate and fraudulent transactions.

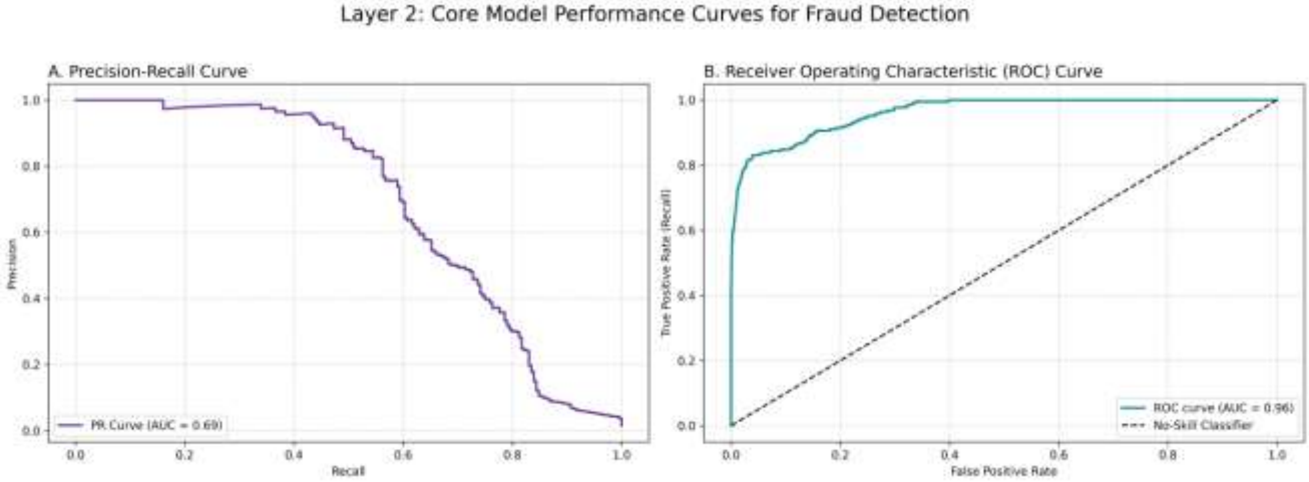

Figure 5: Core Model Performance curves for Fraud Detection.

The PR curve (Figure 5A) is particularly informative for imbalanced classification tasks. The model achieves a PR Area Under the Curve (AUC) of 0.69. This score represents a significant improvement over a random baseline and demonstrates the model's capacity to maintain a high level of precision across a substantial range of recall values. The curve's shape indicates that the model can successfully identify a large fraction of fraudulent transactions while minimizing the rate of false positive alarms.

The ROC curve (Figure 5B) further underscores the model's exceptional discriminative power. The model attains a ROC AUC of 0.96, a value approaching a perfect score of 1.0. This indicates a very high probability that the model will correctly assign a higher fraud likelihood score to a randomly selected fraudulent transaction than to a randomly selected legitimate one. The steepness of the curve towards the top-left corner signifies that the model achieves a high True Positive Rate (Recall) while incurring a very low False Positive Rate, confirming its robustness as a fraud likelihood estimator.

Layer 2: Classification Performance and Probability Analysis The detailed classification performance of the Layer 2 XGBoost model is presented in Figure 6. This includes both the final classification decisions and the underlying probability distributions that inform them.

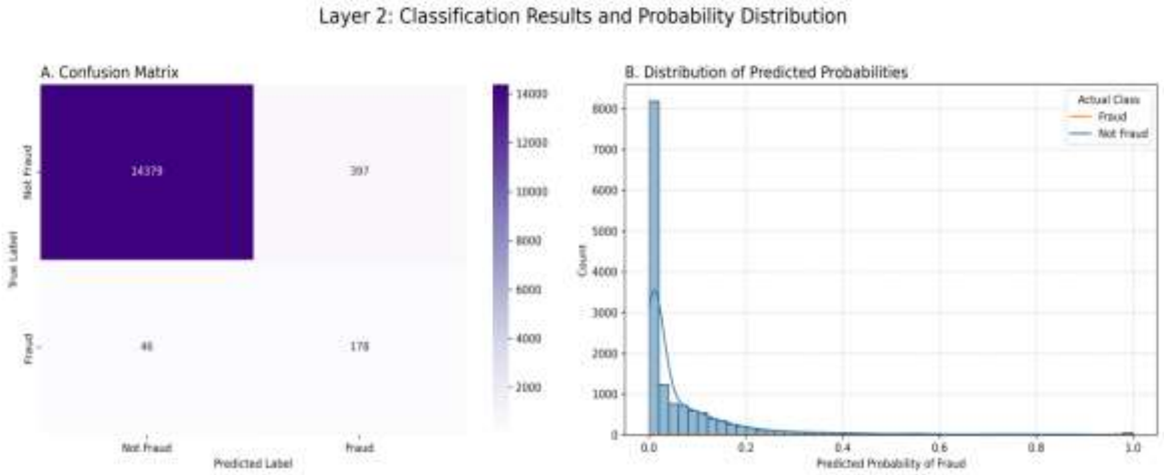

Figure 6: Classification Results and Probability Distribution.

The confusion matrix (Figure 6A) quantifies the model's performance at the default 0.5 probability threshold. The model correctly identified 178 fraudulent transactions (True Positives) while failing to detect 46 (False Negatives), corresponding to a high recall of approximately 79.5%. The model incorrectly flagged 397 legitimate transactions (False Positives), which reflects the trade-off made by the scale_pos_weight parameter to prioritize the capture of fraudulent events.

The histogram of predicted probabilities (Figure 6B) provides a more granular view of the model's behavior. It illustrates a clear and effective separation between the two classes. The legitimate transactions (blue distribution) are overwhelmingly assigned a fraud probability near zero. In contrast, the fraudulent transactions (orange distribution) are assigned a much wider range of scores, with a significant concentration at higher probabilities. This distinct separation between the probability distributions for the two classes is the underlying reason for the model's strong discriminative power, as evidenced by its high ROC AUC score.

Layer 2: Model Interpretability and Calibration To ensure the model's decisions are transparent and its probabilistic outputs are reliable, an analysis of feature importance and model calibration was performed (Figure 7).

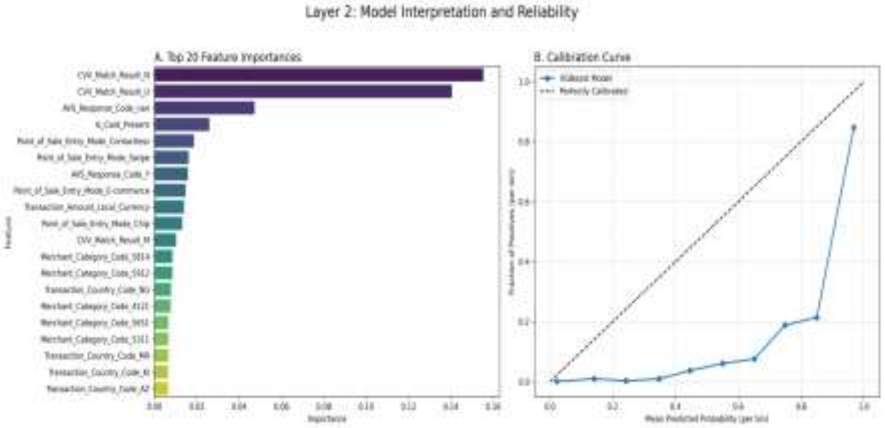

Figure 7: Model Interpretation and Reliability.

The feature importance plot (Figure 7A) reveals the key drivers of the model's predictions. The results align with established domain knowledge. The most influential features are *CVV_Match_Result_N* (CVV does not match) and *CVV_Match_Result_U* (CVV check was not performed or issuer is not certified), followed by *AVS_Response_Code_nan* (missing Address Verification System response). These features are direct indicators of transaction risk. Other significant features include *Is_Card_Present* and various *Point_of_Sale_Entry_Mode* categories, highlighting the importance of the transaction's physical context in assessing its legitimacy.

The calibration curve (Figure 7B) evaluates the reliability of the model's predicted probabilities. A perfectly calibrated model would follow the diagonal dashed line. The curve for the XGBoost model lies below this diagonal, indicating that the model is somewhat under-confident; for instance, when the model predicts a mean probability of 0.8, the actual fraction of fraudulent transactions in that bin is closer to 0.9. However, the curve is monotonic, which is a crucial positive attribute. This demonstrates that an increase in the model's predicted probability consistently corresponds to a true increase in the likelihood of fraud, confirming that the probability scores are a reliable ranking metric for risk.

5.3. Layer 3: Classification Performance and Reliability

The performance of the final, optimized XGBoost model for billing error detection is summarized in Figure 8. The analysis includes the model's classification accuracy via a confusion matrix and the reliability of its probabilistic outputs via a calibration curve.

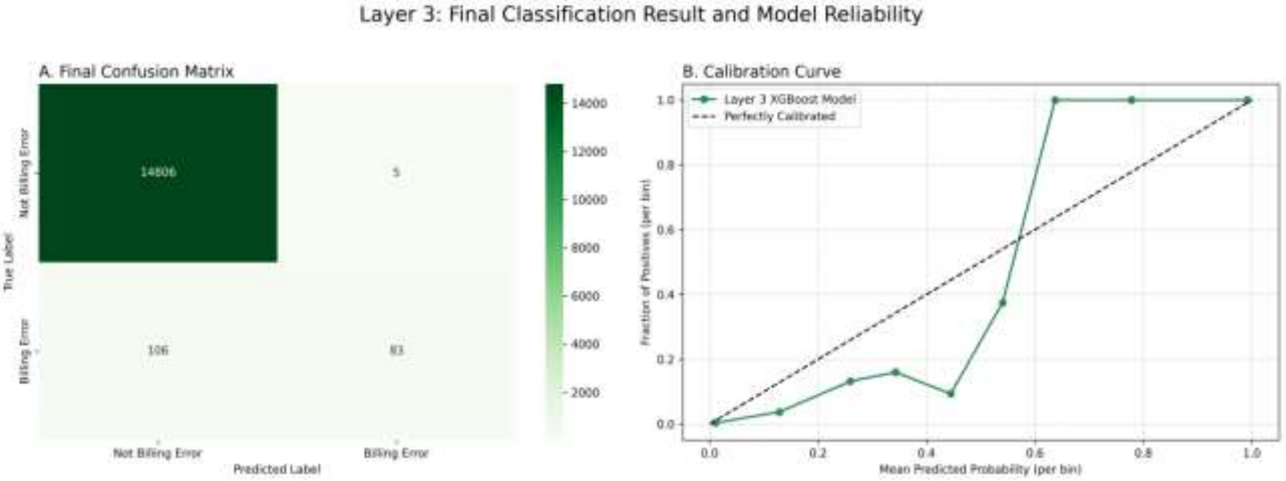

Figure 8: Classification Result and Model Reliability.

The confusion matrix (Figure 8A) for the final Layer 3 model demonstrates a highly practical and well-balanced performance. The model correctly identified 83 billing errors (True Positives) while missing 106 (False Negatives), resulting in a recall of approximately 44%. Critically, the model generated only 5 False Positives, leading to an exceptionally high precision of 94%. This indicates that when the model flags a transaction as a billing error, the alert is highly reliable, minimizing the operational cost of reviewing false alarms.

The calibration curve (Figure 8B) assesses the trustworthiness of the model's probability scores. The curve for the Layer 3 model exhibits a sigmoidal shape relative to the "Perfectly Calibrated" diagonal. This indicates the model is slightly under-confident for low-probability predictions and slightly over-confident for high-probability predictions. However, the curve is strongly monotonic, showing a clear positive correlation between the predicted probability and the actual fraction of positive cases. This confirms that the model's probability scores are effective for ranking transactions by their likelihood of being a billing error, even if they are not perfectly calibrated.

Layer 3: Quantitative Performance Metrics The quantitative performance of the optimized Layer 3 model is further detailed by the Precision-Recall (PR) and Receiver Operating Characteristic (ROC) curves, presented in Figure 9.

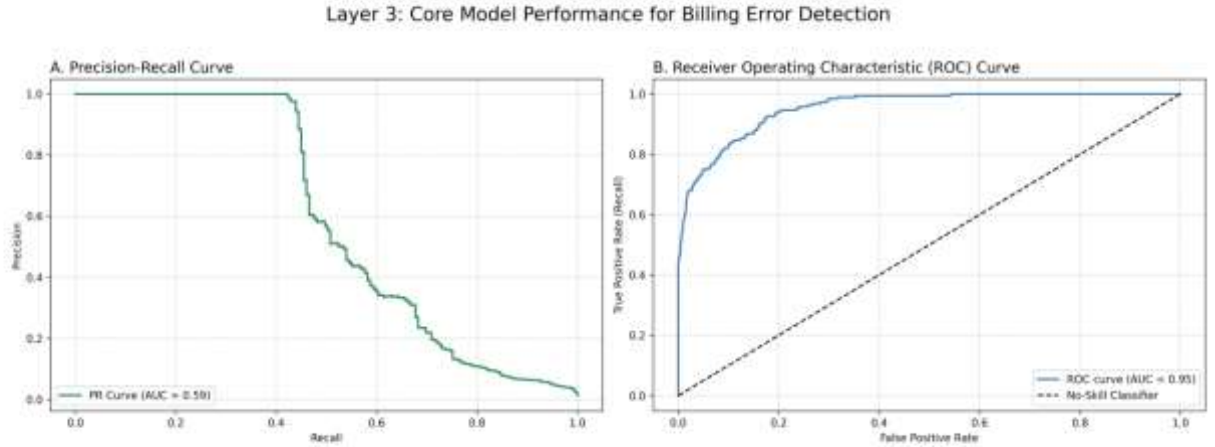

Figure 9: Core Model Performance for Billing Error Detection.

The PR curve (Figure 9A) for the billing error detector achieves an Area Under the Curve (AUC) of 0.59. This result is substantially better than a random baseline and demonstrates the model's effectiveness. The curve illustrates the inherent trade-off in the detection task; the model can maintain very high precision at recall levels below approximately 0.4, after which precision begins to decline more rapidly as the model attempts to identify a larger fraction of the positive class. This performance is indicative of a practical classifier that can be tuned to prioritize either high alert reliability or comprehensive detection depending on the operational requirements.

The ROC curve (Figure 9B) confirms the model's strong discriminative capabilities with a ROC AUC of 0.94. This high value indicates that the model is very effective at ranking transactions, with a 94% probability of assigning a higher risk score to a random billing error than to a random legitimate transaction. The sharp ascent of the curve toward the top-left corner demonstrates that the model achieves a high True Positive Rate (Recall) while maintaining a low False Positive Rate, solidifying its utility as a reliable detector for billing anomalies.

Layer 3: Probability Distribution and Feature Analysis Further insight into the behavior of the Layer 3 model is provided by an analysis of its predicted probability distribution and feature importances, as shown in Figure 10.

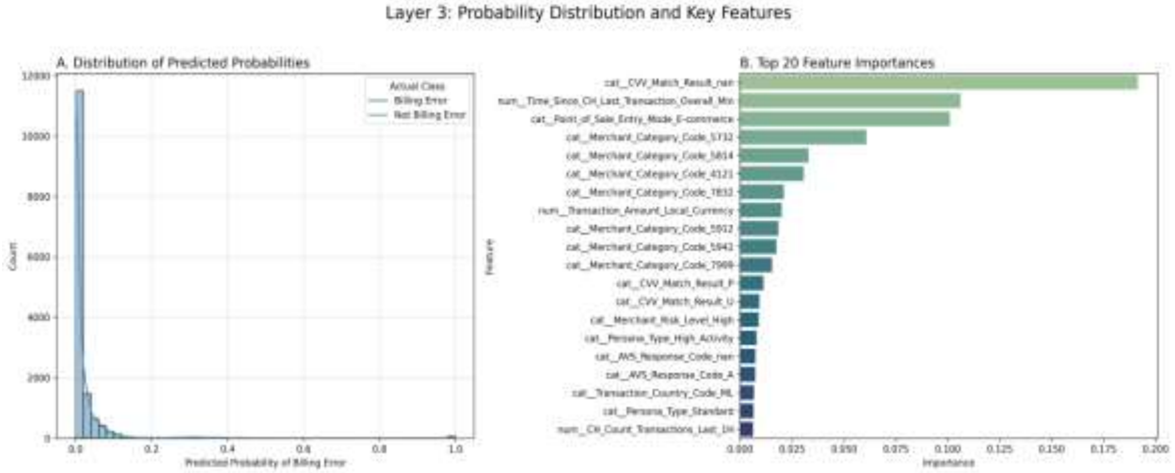

Figure 10: Probability Distribution and Key Features.

The distribution of predicted probabilities (Figure 10A) confirms the model's conservative nature, which is appropriate for a high-precision task. The vast majority of transactions are assigned a billing error probability very close to zero, aligning with the low prevalence of billing errors in the dataset. The small number of transactions that receive a higher probability score are the ones targeted for review, and as established by the confusion matrix, these high-probability predictions are highly reliable.

The feature importance plot (Figure 10B) identifies the primary drivers of the model's decisions. The most significant feature is *CVV_Match_Result_nan*, indicating that transactions where the CVV check was not performed or was unavailable are strong indicators of potential billing discrepancies. Other key features include *Time_Since_CH_Last_Transaction_Overall_Min* and *Point_of_Sale_Entry_Mode_E-commerce*, suggesting that the timing of transactions and the context of online purchases are highly relevant for distinguishing billing errors. The importance of various merchant category codes further underscores the model's ability to learn patterns associated with specific types of merchants.

Layer 3: Billing Error Type Classification Results

Table 1: Classification Report of Random Forest Model to Determine Billing Error Type.

Class	Precision	Recall	F1-Score	Support
Duplicate_Charge	1.00	0.96	0.98	23
Unwanted_Subscription_Renewal	0.99	1.00	1.00	166
Accuracy			**0.99**	**189**
Macro Avg	1.00	0.98	0.99	189
Weighted Avg	0.99	0.99	0.99	189

Furthermore, the second-stage Random Forest classifier, tasked with categorizing the specific Billing_Error_Type for transactions already flagged by the binary detector, demonstrated exceptional performance. On the test set of confirmed billing errors, this multi-class model achieved an overall accuracy of 99%. As detailed in Table 1, it was highly effective at distinguishing between the two primary error types, achieving 1.00 precision and 0.96 recall for Duplicate_Charge, and 0.99 precision and 1.00 recall for Unwanted_Subscription_Renewal. This high-performing classification stage, with a weighted F1-score of 0.99, confirms the system's capability to not only detect billing errors but also to accurately classify their nature for effective operational handling.

5.4. Layer 4: Meta-Learner Performance Evaluation

The overall performance of the Layer 4 meta-learner, which provides the final system-wide risk assessment, is evaluated using the PR and ROC curves shown in Figure 11. These curves are generated based on the model's ability to predict the unified Meta_Target_High_Risk target.

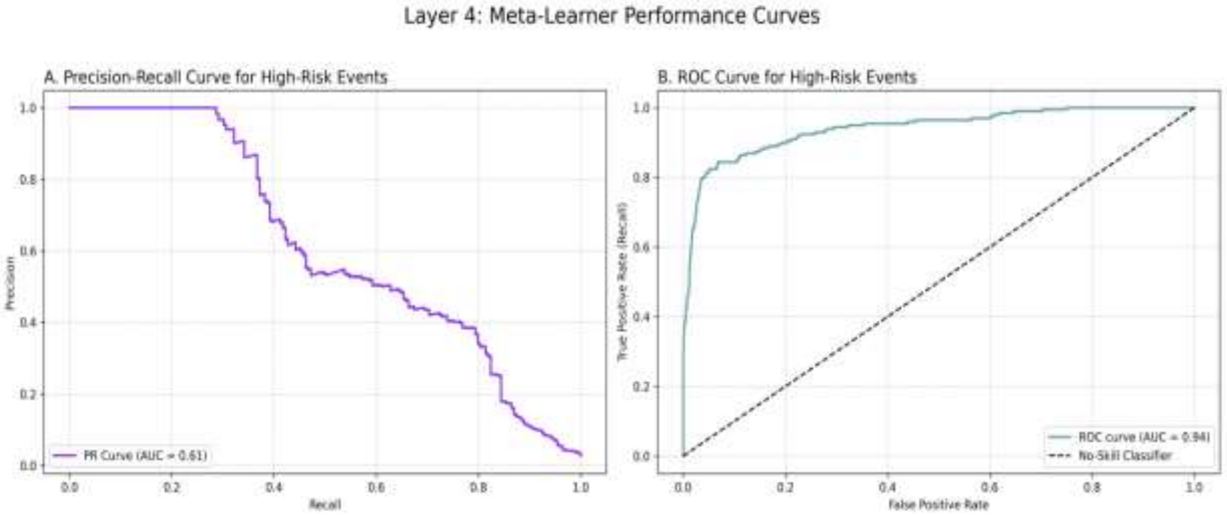

Figure 11: Meta-Learner Performance Curves.

The PR curve (Figure 11A) for the meta-learner achieves an AUC of 0.62. This strong performance indicates that the synthesized risk score is highly effective. The shape of the curve demonstrates that the model can identify over 30% of all high-risk events (recall) while maintaining nearly perfect precision. This highlights the meta-learner's ability to successfully combine the signals from the preceding layers to produce a highly reliable final risk score.

The ROC curve (Figure 11B) further validates the final model's effectiveness, with a ROC AUC of 0.94. This near-perfect score signifies an excellent capacity to discriminate between high-risk and benign transactions. The steep initial ascent of the curve demonstrates that the meta-learner can achieve a very high True Positive Rate while maintaining a minimal False Positive Rate, confirming that the multi-layered architecture successfully culminates in a robust and accurate final decision engine.

Layer 4: Final Classification Performance and Interpretability The final performance and interpretability of the meta-learner are detailed in Figure 12, which presents the confusion matrix for the unified high-risk target and the feature importances that drive the final decision.

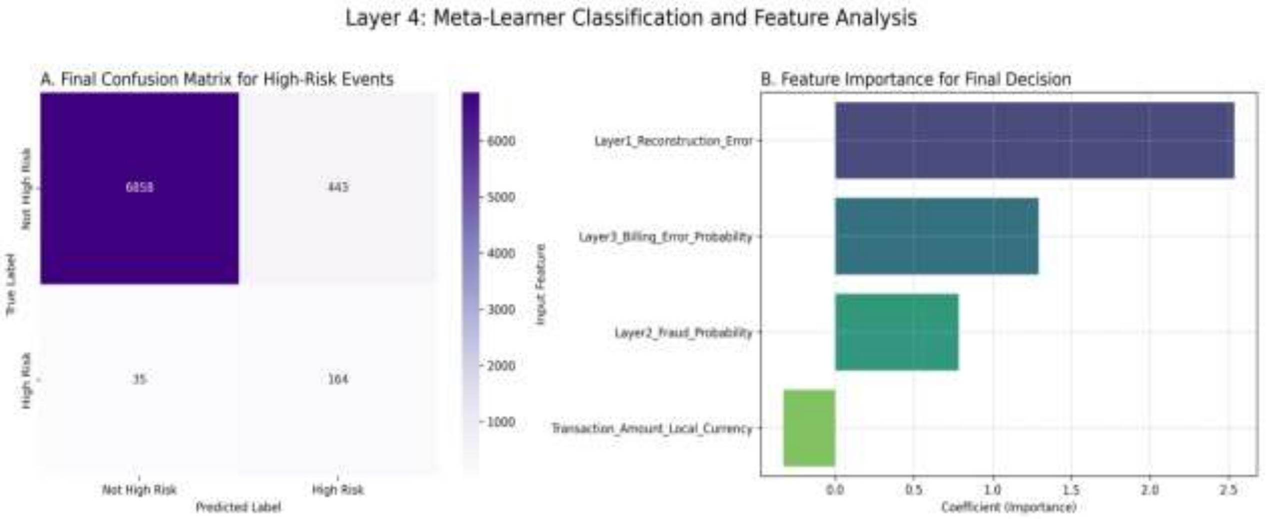

Figure 12: Meta-Learner Classification and Feature Analysis.

The confusion matrix (Figure 12A) for the Meta_Target_High_Risk demonstrates the system's overall effectiveness. The model correctly identifies 164 high-risk events (True Positives) while missing only 35 (False Negatives). This corresponds to a final system-wide recall of approximately 82.4%, indicating that the multi-layered architecture is highly successful at its primary goal of capturing the vast majority of adverse events, including both fraud and billing errors.

The feature importance plot (Figure 12B) provides crucial insight into how the meta-learner synthesizes the signals from the preceding layers. The most influential feature is the *Layer1_Reconstruction_Error*, confirming the significant value of the initial unsupervised anomaly detection layer in identifying transactions that deviate from the norm. The *Layer3_Billing_Error_Probability* and *Layer2_Fraud_Probability* also hold substantial positive coefficients, indicating that the meta-learner correctly associates higher probabilities from these specialist models with increased overall risk. This analysis confirms that the meta-learner has successfully learned a logical and effective strategy for combining the outputs of the specialist models into a reliable, final risk assessment.

Layer 4: Actionable Insights and Risk Distribution The final output of the Inteligent Credit Sentinel system is a set of recommended actions derived from the meta-learner's risk probability scores. Figure 13 illustrates the distribution of these actions and the underlying risk scores that determined them.

The bar chart of suggested actions (Figure 13A) demonstrates the system's operational output. Based on predefined risk thresholds (0.4 for review, 0.7 for decline), the vast majority of transactions are categorized as '*Approve*', which is expected in a real-world scenario. A smaller, manageable number of transactions are escalated for '*Flag_For_Review*' or immediate '*Decline_Or_StepUp*', showcasing the system's ability to translate probabilistic outputs into a practical and efficient workflow.

The histogram of the final risk scores (Figure 13B) provides a compelling visualization of the model's confidence. The distribution is distinctly bimodal, with a large peak of low-risk scores concentrated near zero and a smaller, but clearly defined, peak of high-risk scores concentrated near 1.0. This separation indicates that the meta-learner is highly decisive, with very few transactions falling into an ambiguous intermediate-risk category. The plot confirms that the action thresholds are well-positioned to effectively segment the low-risk and high-risk populations, providing a strong justification for the final decisioning logic.

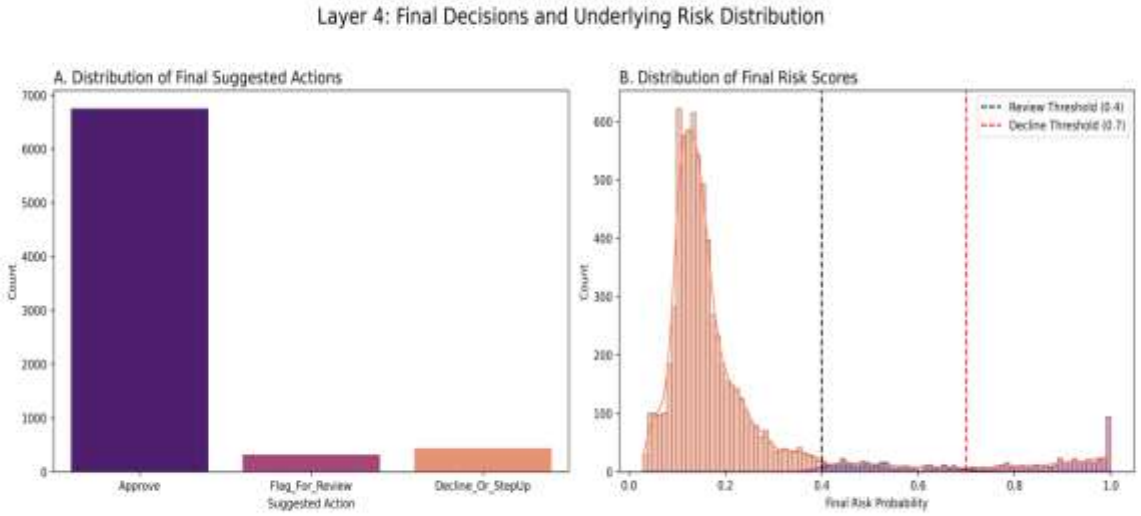

Figure 13: Final Decisions and Underlaying Risk Distribution.

Layer 4: Case Study of the Decision Engine To demonstrate the end-to-end functionality of the multi-layered system, Table 1 presents a selection of outputs from the Layer 4 meta-learner. The table showcases how the final risk probability and suggested action are derived from the inputs of the preceding layers.

The examples illustrate the system's behavior on typical, low-risk transactions, where low probabilities from all layers result in a low final risk score and an '*Approve*' action. The most illustrative case is that of transaction 67513, a confirmed fraudulent event. For this transaction, the Layer 1 anomaly score and Layer 3 billing probability are both low. However, the Layer 2 fraud detector assigned a very high fraud probability of 0.99243.

The meta-learner correctly synthesized these inputs, assigning the highest importance to the strong signal from the fraud detection layer. Consequently, it produced a final risk probability of 0.95195, leading to the correct suggested action of '*Decline_Or_StepUp*'. This case study effectively demonstrates the core strength of the multi-layered architecture: the ability of the final decision engine to intelligently weigh the evidence from different specialist models to arrive at an accurate and actionable conclusion.

6. DISCUSSION

The development and evaluation of the *Inteligent Credit Sentinel* system reveal several key insights into the effectiveness of a multi-layered architecture for detecting diverse financial transaction risks. This discussion synthesizes the findings from the unsupervised anomaly screener, the specialized supervised classifiers, and the final meta-learner, focusing on their comparative performance and the practical implications of the hierarchical design. The results confirm that a modular, multi-faceted approach not only enhances detection accuracy but also creates a more interpretable and operationally efficient framework for managing risk.

6.1. The Synergy of Unsupervised and Supervised Learning

A primary finding of this study is the distinct but complementary roles of unsupervised and supervised learning methodologies (Zimbe et al., 2025). The Layer 1 autoencoder established its utility as a broad-based anomaly screener, successfully learning a representation of normative behavior to flag transactions that deviated from the norm (Nettey & Ansong, 2025). It demonstrated a foundational ability to identify fraudulent events without any prior labels (Nettey & Ansong, 2025). However, its performance was significantly surpassed by the Layer 2 supervised XGBoost model, which was trained specifically on fraud labels and achieved a much higher recall (Zimbe et al., 2025). This contrast underscores a fundamental principle: while unsupervised methods are invaluable for detecting novel anomalies (Zimbe et al., 2025), supervised models achieve superior performance on well-defined targets (Zimbe et al., 2025). The true synergy, however, was revealed in the final Layer 4 meta-learner (Zimbe et al., 2025), where the Layer 1 reconstruction error emerged as the single most important feature for the final risk decision. This demonstrates that the unsupervised layer provided a powerful, overarching context of "abnormality" that, when combined with the specific insights of the supervised layers (Zimbe et al., 2025), created a more robust and effective final model than any single component could achieve on its own (Nettey & Ansong, 2025).

6.2. Deconstructing Risk: The Divergent Signatures of Fraud and Billing Errors

The investigation revealed a clear disparity in the "detectability" of fraudulent transactions versus billing errors, highlighting that not all risks present equally strong signals. The Layer 2 fraud model attained an exceptional ROC AUC of 0.96, driven by features with clear, unambiguous links to fraudulent activity, such as failed CVV and AVS checks. These "smoking gun" indicators allowed the model to easily and confidently separate fraud from legitimate transactions. In contrast, the Layer 3 billing error detector required extensive feature expansion and tuning to achieve its strong final performance. Its key features were more subtle and circumstantial, relying on transaction timing and merchant-specific history. This suggests that billing errors have a more ambiguous signature that can closely resemble normal customer behavior, making them an inherently more challenging detection problem. This finding validates the architectural decision to dedicate a separate, highly-tuned layer to this specific challenge rather than grouping it with the more distinct patterns of fraud. (Zimbe et al., 2025)

6.3. From Probabilities to Practicality: The Role of Model Tuning and Calibration

This research underscores the critical importance of iterative model tuning in transforming a theoretically powerful model into a practically useful one. The initial supervised models, particularly for the rare billing error class, were not immediately effective despite strong underlying metrics like a high ROC AUC. Achieving a balance between capturing rare events (recall) and minimizing false alarms (precision) required a rigorous, data-driven tuning process. The automated hyperparameter search for the Layer 3 model, which identified an optimal *scale_pos_weight* of 2.82, was instrumental. This single tweak transformed the model from a high-recall but impractical detector into a well-balanced classifier with exceptional 94% precision. This journey from a raw model to a refined one highlights that for imbalanced classification problems, the process of navigating the precision-recall trade-off through careful hyperparameter optimization is as important as the initial choice of algorithm itself. (Zimbe et al., 2025)

6.4. The Meta-Learner as an Intelligent Arbiter

The success of the Layer 4 meta-learner demonstrates the core strength of the multi-layered architecture: the ability to synthesize diverse, specialized signals into a single, superior decision metric. The final model achieved a remarkable system-wide recall of approximately 82.4% for the unified "High-Risk Event" target, confirming its effectiveness. The feature importance analysis revealed that the meta-learner did not simply average the inputs but learned to weigh them intelligently. Its reliance on the Layer 1 anomaly score as the most critical feature suggests it learned to prioritize the general signal of "weirdness" as a primary indicator of risk, which it then refined using the more specific fraud and billing error probabilities. Furthermore, the bimodal distribution of its final risk scores—with clear peaks at very low and very high risk—proves that the final engine is highly decisive, avoiding the ambiguity that can plague monolithic systems and providing a clear basis for action.

6.5. A Hierarchical Architecture as a Blueprint for Operational Efficiency

Ultimately, the *Inteligent Credit Sentinel* system serves as a compelling blueprint for a practical and efficient risk management workflow. The tiered structure mirrors a sophisticated human-led operational process: a low-cost, automated initial screening (Layer 1) filters the vast majority of transactions, followed by analysis from dedicated experts (Layers 2 and 3), with a final "manager" (Layer 4) making an evidence-based decision. This hierarchical approach is inherently efficient, ensuring that the most intensive scrutiny is reserved for the small subset of transactions that truly warrant it. By translating the final, confident risk scores into concrete, triaged actions—*Approve*, *Flag_For_Review*, and *Decline_Or_StepUp*—the system provides a clear, automated, and interpretable pathway from data to decision, representing a robust and scalable paradigm for modern financial security.

7. CONCLUSION

The challenge of securing digital transactions in an era of ever-evolving threats demands a paradigm shift away from singular, monolithic defenses toward more dynamic, intelligent, and multi-faceted systems. This research confronted this challenge by proposing and validating the Inteligent Credit Sentinel, a novel four-layered architecture designed to deconstruct and analyze transaction risk with a depth and specificity that a single model cannot achieve. By treating different risk types not as a singular problem but as distinct challenges requiring specialized expertise, we have demonstrated a path to a more robust and efficient security posture.

The journey through the layers of the Sentinel has yielded significant insights. We confirmed that an unsupervised autoencoder can serve as an effective first-line screener, identifying general abnormality and providing a crucial, high-level context that proved indispensable to the final decision engine. We further demonstrated that specialized, supervised XGBoost models, when meticulously tuned to navigate the critical trade-off between precision and recall, can achieve exceptional performance in detecting the distinct signatures of both overt fraud and subtle billing errors. The true success of the architecture, however, was realized in the final meta-learner, which acted not as a simple aggregator but as an intelligent arbiter, learning to weigh the evidence from each preceding layer to forge a single, confident, and highly accurate risk assessment.

Ultimately, the Inteligent Credit Sentinel system serves as a compelling blueprint for a new generation of financial security systems. It successfully translates a complex hierarchy of probabilistic outputs into a clear, triaged, and actionable workflow, proving that a modular, multi-layered approach can deliver a solution that is at once powerful in its accuracy, elegant in its design, and practical in its application. The principles demonstrated herein offer a promising and scalable framework for building the resilient financial ecosystems of the future.

REFERENCES

Airlangga, G. (2024). A hybrid ensemble approach for enhanced fraud detection: Leveraging stacking classifiers to improve accuracy in financial transaction. *Journal of Computer System and Informatics (JOSYC), 5*(4), 1118–1127. https://doi.org/10.47065/josyc.v5i4.5840

Airlangga, G. (2024). Evaluating the efficacy of machine learning models in credit card fraud detection. *Journal of Computer Networks Architecture and High Performance Computing, 6*(2), 829–837. https://doi.org/10.47709/cnahpc.v6i2.3814

Alarfaj, F., Malik, I., Khan, H., Almusallam, N., Ramzan, M., & Ahmed, M. (2022). Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. *IEEE Access, 10*, 39700–39715. https://doi.org/10.1109/access.2022.3166891

Boulieris, P., Pavlopoulos, J., Xenos, A., & Vassalos, V. (2023). Fraud detection with natural language processing. *Machine Learning, 113*(8), 5087–5108. https://doi.org/10.1007/s10994-023-06354-5

Hájek, P., Abedin, M., & Sivarajah, U. (2022). Fraud detection in mobile payment systems using an XGBoost-based framework. *Information Systems Frontiers, 25*(5), 1985–2003. https://doi.org/10.1007/s10796-022-10346-6

Hájek, P., Abedin, M., & Sivarajah, U. (2022). Fraud detection in mobile payment systems using an XGBoost-based framework. *Information Systems Frontiers, 25*(5), 1985–2003. https://doi.org/10.1007/s10796-022-10346-6

Jiang, S., Dong, R., Wang, J., & Xia, M. (2023). Credit card fraud detection based on unsupervised attentional anomaly detection network. *Systems, 11*(6), 305. https://doi.org/10.3390/systems11060305

Malik, E., Khaw, K., Belaton, B., Wong, W., & Chew, X. (2022). Credit card fraud detection using a new hybrid machine learning architecture. *Mathematics, 10*(9), 1480. https://doi.org/10.3390/math10091480

Niu, X., Wang, L., & Yang, X. (2019). A comparison study of credit card fraud detection: Supervised versus unsupervised. *arXiv Preprint*. https://doi.org/10.48550/arxiv.1904.10604

Pk, R. (2023). Enhanced credit card fraud detection: A novel approach integrating Bayesian-optimized random forest classifier with advanced feature analysis and real-time data adaptation. *International Journal for Innovative Engineering and Management Research*, 537–561. https://doi.org/10.48047/ijiemr/v12/issue05/52

Salekshahrezaee, Z., Leevy, J., & Khoshgoftaar, T. (2023). The effect of feature extraction and data sampling on credit card fraud detection. *Journal of Big Data, 10*(1). https://doi.org/10.1186/s40537-023-00684-w

Wu, T., & Wang, Y. (2021). Locally interpretable one-class anomaly detection for credit card fraud detection. *arXiv Preprint*. https://doi.org/10.48550/arxiv.2108.02501

Youssef, N. (2025). Comparative study of data-level imbalance handling techniques with ensemble models for credit card fraud detection. *Research Square*. https://doi.org/10.21203/rs.3.rs-7004067/v1

(2024). Machine learning for identifying fraud in credit card transactions. *Iraqi Journal of Computer Communication Control and System Engineering*, 71–83. https://doi.org/10.33103/uot.ijccce.24.1.6

-, S., -, D., Ahmed, M., & Meghana, P. (2025). Cardsheild: A credit card fraud detection system. *IJSAT, 16*(2). https://doi.org/10.71097/ijsat.v16.i2.3921

Goldstein, M., & Uchida, S. (2016). A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. *PLOS ONE, 11*(4), e0152173. https://doi.org/10.1371/journal.pone.0152173

Jeyaraj, R., Natraj, N., Rajasekaran, M., & Gangadharan, K. (2024). Machine learning techniques for fraud detection in financial services. *IJAITRD, 01*(01). https://doi.org/10.69942/1920184/20240101/04

Mienye, I., & Sun, Y. (2023). A deep learning ensemble with data resampling for credit card fraud detection. *IEEE Access, 11*, 30628–30638. https://doi.org/10.1109/access.2023.3262020

Nata, A., Manalu, D., Hardinata, J., & Sitorus, P. (2025). Anomaly-based financial fraud detection using autoencoder: A case study on the Kaggle credit card dataset. *Journal of ICT Applications and Systems, 4*(1), 26–35. https://doi.org/10.56313/jictas.v4i1.431

Niu, X., Wang, L., & Yang, X. (2019). A comparison study of credit card fraud detection: Supervised versus unsupervised. *arXiv Preprint*. https://doi.org/10.48550/arxiv.1904.10604

Nettey, D., & Ansong, E. (2025). Anomaly detection with variational autoencoders. *Research Square*. https://doi.org/10.21203/rs.3.rs-5809487/v1

Rihan, S., Anbar, M., & Alabsi, B. (2023). Meta-learner-based approach for detecting attacks on Internet of Things networks. *Sensors, 23*(19), 8191. https://doi.org/10.3390/s23198191

Zimbe, I., Zeyeum, J., Ayano, K., & Olufemi, O. (2025). Designing and evaluating AI-powered predictive models for detecting unemployment insurance fraud: A data-driven approach to enhancing the integrity of U.S. public benefit systems. *International Journal of Science and Research Archive, 16*(1), 2276–2336. https://doi.org/10.30574/ijsra.2025.16.1.2134