IJITEB
Vol. 8
No. 1
2025

International Journal of Information Technology and Business

http://ejournal.uksw.edu/ijiteb

Cost-Sensitive Fraud Detection with Reliability Calibration: A Practical Pipeline with XGBoost and Focal-Proxy Reweighting

Danang[1*], Toni Wijanarko Adi Putra[2]
[1-2]Universitas Sains dan Teknologi Komputer, Indonesia

Abstract: Fraud detection on payment transactions is an extremely imbalanced, high-stakes classification task in which deployment decisions depend not only on ranking quality but also on reliable prob ability estimates. We study credit card fraud detection on a standard real-transaction benchmark (284,807 transactions; 492 frauds) and target two deployment requirements: cost-sensitive thresh olding under asymmetric error costs and reliability calibration so model outputs can be interpreted as stable risk scores. We benchmark Logistic Regression and XGBoost, and propose a focal-proxy reweighting scheme for boosted trees via iterative weight updates inspired by focal loss. Probabili ties are calibrated on validation using Platt scaling, temperature scaling, and isotonic-style monotone calibration; the best calibrator is selected by minimum validation Brier score. For decision making, we choose the operating threshold that minimizes expected cost Cost(t) = 10 · FN(t) + 1 · FP(t) on validation, then evaluate on a held-out test set. On the benchmark split (train 199,364; valida tion 42,721; test 42,722), the calibrated XGBoost baseline achieves AUROC 0.973, AUPRC 0.812, fraud-class F1 0.767, and expected cost 154 with very low calibration error (ECE = 1.1 × 10−4). Overall, calibration reduces ECE and improves or maintains Brier score, while cost-aware thresh olding makes the FN/FP trade-off explicit via decision curves.

Keywords: Class Imbalance, Cost-Sensitive Learning, Fraud Detection, Probability Calibration, XGBoost.

1. Introduction

Fraud detection in card-not-present and point-of-sale systems is fundamentally an operational decision problem, not merely a ranking problem. In production, the system must translate model outputs into actions under hard constraints: limited manual-review capacity, customer friction, and an ever-present trade-off between catching fraud and avoiding unnecessary declines or investigations. The learning problem is also structurally difficult: fraud labels are rare, the class distribution is extremely skewed, and the data-generating process is not static. Labels can be delayed, investigation policies change, and fraud patterns adapt, which means that the same score distribution can correspond to different real-world outcomes over time. In this setting, a model may look strong on aggregate discrimination metrics while still failing as a decision tool because what matters is not only separation, but whether the predicted probabilities are usable for policy and triage and whether the decision threshold is aligned with explicit error costs.

A common failure mode is therefore metric mismatch. AUROC can remain high even when oper ational precision at low false-positive rates is poor, and even AUPRC can be misleading if score cali bration is unstable or if evaluation is disconnected from deployment constraints. Because fraud is rare, the precision–recall trade-off is often the more relevant lens for understanding the behavior of a detector under severe imbalance, and its relationship to ROC clarifies why two models with similar ROC be havior can imply very different positive predictive value at actionable thresholds [1]. Separately, the literature on imbalanced learning emphasizes that data-level and algorithm-level imbalance treatments (including resampling) can materially change both discrimination and probabil ity estimates, which is critical when downstream decisions consume model outputs as calibrated risks rather than as raw ranks [2], [3], [4]. In fraud specifically, prior work highlights that common imbalance strategies such as undersam pling can interact with calibration and should be evaluated with probabilistic objectives rather than with ranking metrics alone [5].

This study considers supervised fraud detection on a widely used benchmark of real credit card trans actions from the ULB dataset family [5], [6]. The standard fea ture set (Time, anonymized PCA components, and Amount) represents a practical tabular setting where gradient-boosted decision trees are frequently competitive and widely deployed due to their strong per formance and favorable engineering properties [7], [8], [9]. At the same time, modern deep learning for tabular data has progressed substantially, with architectures such as TabNet and transformer-based variants proposed to better capture feature interactions [10], [11], [12]. However, evidence across bench marks suggests that deep models are not uniformly superior to strong tree-

* Corresponding authors
e-mail addresses: danang150787@gmail.com

based baselines on tabular data, reinforcing the value of careful baseline design and decision-oriented evaluation rather than as suming architecture alone solves deployment needs [13]. Motivated by this landscape, we focus on boosting-based baselines and study what is typically the real bottleneck in fraud deployment: reliability and decision alignment, not just raw separability.

Asecond failure mode is probability unreliability. In many operational pipelines, predicted proba bilities are treated as risk scores to drive prioritization, audit planning, and policy rules; the organization implicitly assumes that a score of 0.2 corresponds to roughly a 20% fraud rate among flagged transac tions. When this assumption is violated, alert volumes become unstable, capacity planning breaks down, and policy thresholds become brittle. Calibration is therefore not a cosmetic post-processing step, but a requirement for consistent decision-making. Prior work shows that modern models can be substantially miscalibrated even when accuracy or discrimination is high, motivating explicit post-hoc calibration and proper probabilistic evaluation [14]. Moreover, calibration must be evaluated carefully: both the choice of metric and the diagnostic procedure matter, especially under im balance [15]. The problem becomes more severe under dataset shift, where uncertainty estimates and calibration may degrade, and the ability to trust probabilistic out puts under changing conditions becomes a central concern [16], [17].

Research focus**.** We center this work on two practical requirements that determine whether a fraud detector can be used as a decision component. First, we study reliability calibration: can the model's predicted probabilities be trusted as risk scores for prioritization and policy, or do they systematically over/under-estimate true fraud likelihood in ways that would destabilize thresholds and downstream op erations? Second, we study cost-sensitive decision making: given that the cost of missing a fraudulent transaction (false negative) typically dominates the cost of sending a legitimate transaction to manual re view (false positive), how should the decision threshold be chosen to minimize expected operational cost rather than using arbitrary cutoffs such as 0.5 [18]. These two requirements are tightly coupled in practice: even a strong ranker can produce poor decisions if probabilities are miscalibrated, and even a calibrated model can be misused if the threshold ignores the cost matrix and capacity constraints.

Contributions. This paper provides an end-to-end, reproducible pipeline that explicitly links model training, probability calibration, and cost-aware decision rules for imbalanced fraud detection. We benchmark transparent and widely used tabular baselines Logistic Regression (LR) and gradient boosted trees via XGBoost (XGB) as strong references for discrimination performance on this dataset family [7]. Building on the intuition that rare and hard examples should have greater influence in heavily imbalanced detection, we introduce a practical focal-proxy objective for boosted trees via iterative sample reweighting inspired by focal loss [19], and we test whether this objective improves downstream decision utility under asymmetric costs. To make outputs usable as risk scores, we apply post-hoc probability calibration on a held-out valida tion set using established methods Platt scaling, isotonic-style monotone calibration, and temperature scaling and we select calibration strategies using proper probabilistic scoring (e.g., Brier score) rather than only discrimination metrics [14], [20], [21]. We also situate these choices in the broader calibration literature that includes more expressive families such as beta calibra tion and extensions beyond temperature scaling [22], [23]. Finally, we select an operational threshold by minimizing expected cost on validation in the spirit of cost-sensitive learning [18], and we report both ranking metrics (AUROC/AUPRC) and decision-oriented diagnostics (reliability behavior and cost curves) on the held-out test set. The outcome is an auditable workflow that emphasizes not only how well a model ranks transactions, but also how reliably it quanti f ies risk and how effectively it supports cost-sensitive decisions in a realistic fraud-screening pipeline.

2. Literature Review

2.1. Fraud detection on tabular data

Credit card fraud detection is commonly formulated as supervised binary classification under extreme class imbalance, delayed feedback, and evolving fraud strategies [5], [6]. In many operational pipelines, the classifier is not a standalone component but part of a broader decision system that includes rules, manual review, customer verification, and post-event investigation. These constraints motivate evaluation protocols that go beyond aggregate discrimination and that reflect how the model will be used in practice, particularly when the positive class is rare and the cost of errors is asymmetric. Reviews of the field consistently emphasize that meaningful progress requires strong baselines, careful metric choice (often precision–recall based), and explicit attention to deployment constraints rather than purely offline ranking improvements [24], [25].

From a modeling standpoint, gradient-boosted decision trees (GBDTs) remain a leading approach for tabular fraud detection because they capture non-linear feature interactions, tolerate heterogeneous feature distributions, and typically require less feature engineering than linear models [7]. Practical implementations such as XGBoost introduced scalable boosting with optimized system design [7], while subsequent frameworks such as LightGBM and CatBoost im proved efficiency and introduced refinements that often yield strong performance on large-scale tabular problems [8], [9]. Beyond raw predictive strength, these models are attractive for production because they train quickly, are relatively stable across hyperparameter settings, and integrate well with debugging and interpretability tooling.

In parallel, deep learning for tabular data has progressed substantially, proposing mechanisms to learn feature representations and interactions directly from heterogeneous inputs. TabNet introduced attentive feature selection with an interpretable sparse masking mechanism [10], and transformer-based approaches such as TabTransformer leverage contextual embeddings to model feature dependencies [11]. Further work, including SAINT, explores attention across rows and contrastive pretraining to improve sample efficiency [12]. However, broader bench mark evidence suggests that deep models are not uniformly superior to carefully tuned tree ensembles on tabular datasets, and performance can be sensitive to regularization, preprocessing decisions, and the nature of the dataset itself [13]. As a result, modern fraud detection studies often treat strong GBDTbaselines as the primary reference and focus innovation on decision reliability, calibration, robustness, and operational alignment rather than on architecture novelty alone [24].

2.2 Imbalance, metrics, and cost sensitivity

A defining challenge in fraud detection is that the prevalence of the positive class can be far below 1%, making naive metrics and default decision rules misleading. Under such imbalance, a classifier that predicts the majority class can achieve high accuracy while providing no operational value. For this reason, evaluation commonly prioritizes precision–recall behavior, which more directly characterizes how many alerts are true frauds at a given recall level. The relationship between ROC and precision recall curves clarifies why a model can show strong ROC performance yet still fail to achieve workable precision at the operating points that matter for rare-event detection [1]. This perspective is central to fraud detection protocols, where investigation resources are finite and false positives create tangible operational burden [6], [24].

Algorithmically, handling imbalance has been studied extensively. Surveys of imbalanced learning highlight families of approaches including data-level methods (over/under-sampling), algorithm-level reweighting, and hybrid strategies that combine resampling with modified loss functions or decision rules [3], [4]. More recent reviews from a data per spective detail how resampling choices affect not only discrimination but also the stability of learned decision boundaries, especially when the minority class is extremely sparse [2]. In fraud detection specifically, undersampling has been used to rebalance training data, but prior work notes that such interventions can distort probability estimates and therefore interact with calibration and downstream thresholding [5]. Consequently, improvements must be assessed with both ranking metrics and decision-relevant outcomes rather than assuming that better training balance automatically yields better operational decisions.

Cost-sensitive learning provides a principled framework for incorporating asymmetric error costs directly into the learning and decision process [18]. In many fraud scenarios, the cost of a false negative (missing a fraud) is substantially larger than the cost of a false positive (sending a legiti mate transaction to review). Cost sensitivity can be introduced during training (e.g., via class-weighted objectives) or at decision time by selecting thresholds that minimize an explicit expected cost. Surveys emphasize that effective solutions in highly imbalanced domains often blend reweighting or resampling with cost-aware objectives and rigorous evaluation protocols that reflect deployment constraints [3], [4]. This motivates fraud pipelines that (i) report discrim ination metrics appropriate for rare events and (ii) explicitly optimize and report decision thresholds derived from a cost matrix rather than relying on arbitrary cutoffs.

2.3 Calibration under noise

In operational fraud screening, predicted probabilities are frequently interpreted as risk scores to priori tize investigations, allocate review capacity, and define policy rules. Calibration aims to align predicted probabilities with empirical outcome frequencies so that, for example, predictions around 0.2 correspond to roughly 20% fraud rates among similarly scored transactions. This requirement is orthogonal to dis crimination: a model can rank examples well yet still be poorly calibrated. Analyses of modern models demonstrate that miscalibration is common even when predictive performance is strong, motivating ex plicit post-hoc calibration and careful probabilistic evaluation [14]. In addition, calibration should be evaluated with appropriate diagnostics and estimators; methodological work highlights that calibration assessment is itself a non-trivial statistical task and can be sensitive to sample size, binning, and the chosen metric [15].

Classical post-hoc calibration methods include Platt scaling, which fits a logistic mapping from model scores to calibrated probabilities [21]. Temperature scaling can be viewed as a con strained variant that rescales logits with a single parameter and has become a common baseline due to its simplicity [14]. More expressive monotone approaches include isotonic-like map pings and Beta calibration, which provides a well-founded extension of logistic calibration for binary classifiers while remaining easy to implement [22]. For settings beyond simple binary calibration, Dirichlet calibration generalizes these ideas to provide well-calibrated multi-class probabilities and illustrates how richer calibration families can outperform simple temperature scaling in more complex output spaces [23]. Although our setting is binary fraud detection, these meth ods provide a useful conceptual toolkit and highlight that the calibration family should be chosen and validated rather than assumed.

Calibration quality is typically quantified using proper scoring rules such as the Brier score, which directly measures the accuracy of probabilistic forecasts [20]. Complementary diagnostics in clude summary measures of calibration error (e.g., ECE-style metrics) and visual reliability diagrams, with

careful evaluation procedures recommended to avoid misleading conclusions, especially under im balance and dataset shift [15]. Finally, the broader uncertainty and robustness litera ture emphasizes that predictive uncertainty and calibration can degrade under distribution shift, raising questions about how much one can trust probabilistic outputs when the data distribution changes [1], [17]. In fraud detection, where adversarial adaptation and temporal drift are common, this motivates calibration-aware evaluation and decision rules that explicitly account for uncertainty and operational cost.

3. Proposed Method

Our pipeline is designed to mirror a realistic deployment workflow for fraud screening: we first train discriminative models on a training split, then use a separate validation split to (i) tune model training via early stopping/hyperparameters, (ii) fit post-hoc probability calibration, and (iii) choose an operating threshold using an explicit cost matrix. Only after all choices are fixed do we report final performance on a held-out test split. This separation is essential because calibration and thresholding are decision-layer components that can easily overfit if they are optimized on the test set.

3.1 Dataset and evaluation protocol

We evaluate on a real-world credit card transaction dataset in the ULB benchmark family [5]. Each record contains a timestamp-like feature (Time), 28 anonymized components (V1 V28) obtained via a privacy-preserving transformation, and the transaction Amount. The binary label Class indicates whether the transaction is fraudulent. The dataset exhibits extreme class imbalance (fraud prevalence far below 1%), which is representative of operational fraud screening where positive examples are scarce and error costs are asymmetric [5], [6]. Because of this imbalance, evaluation must emphasize metrics and decision procedures that remain meaningful in rare-event regimes rather than relying on accuracy or default thresholds.

To better approximate deployment conditions and reduce temporal leakage, we split the data chrono logically into train/validation/test with the following sizes: train n = 199,364 (fraud = 344), validation n = 42,721 (fraud = 74), and test n = 42,722 (fraud = 74). The test set is held out for final report ing only. All model selection decisions that could bias outcomes including early stopping, calibration choice, and threshold selection are performed exclusively on the validation set. This protocol matches the stream-oriented nature of fraud detection benchmarks, where concept drift and evolving behavior make temporal separation particularly relevant [6].

In terms of preprocessing, we apply feature standardization only when required by the model family (i.e., for linear models). Tree-based methods operate directly on raw feature scales and are generally robust to monotone reparameterizations, so they do not require normalization. Throughout, we treat the predicted probability as a risk score consumed by downstream decision rules; consequently, we explicitly evaluate calibration in addition to ranking performance.

3.2 Baselines

Weconsider two widely used baselines for tabular fraud detection: Logistic Regression (LR) as a trans parent linear reference and gradient-boosted decision trees via XGBoost (XGB) as a strong non-linear baseline [7]. These baselines reflect common industrial practice: LR offers sim plicity and interpretability, while boosted trees typically provide high performance on heterogeneous tabular data with minimal feature engineering.

Logistic Regression (LR). LR is trained on standardized inputs to ensure stable optimization and comparable coefficient scaling. For an input feature vector x, LR outputs

$$p(x) = \sigma(w^{\top}x + b), \quad (1)$$

where $\sigma(\cdot)$ is the sigmoid function. Given the extreme imbalance, we employ imbalance-aware training (e.g., class-balanced weighting) so that the learned decision boundary is not dominated by the majority class. LR serves as a calibrated-probability baseline in the sense that its outputs have an interpretable probabilistic form, while still benefiting from post-hoc calibration when the model is misspecified or when regularization shifts probability estimates.

XGBoost (XGB). We train gradient-boosted decision trees with a standard log-loss objective and early stopping on the validation split [7]. Early stopping acts as a practical reg ularization mechanism that selects the number of boosting rounds based on validation performance, reducing overfitting without requiring exhaustive tuning. Because the positive class is rare, we also incorporate imbalance-aware settings (e.g., positive-class weighting) so that the learner pays sufficient attention to fraud examples. XGB is a strong baseline for tabular fraud detection and provides a useful testbed for studying how reweighting, calibration, and thresholding affect decision utility.

3.3 Focal-proxy reweighting for XGBoost

Focal loss was introduced to emphasize hard examples by down-weighting easy ones, improving learn ing in highly imbalanced detection settings [19]. For binary labels y ∈ {0,1} and predicted probability p, focal loss can be written as

$$\mathcal{L}local(y,p) = -\alpha y(1-p)^{\gamma}\log(p) - (1-\alpha)(1-y)p^{\gamma}\log(1-p), \quad (2)$$

where $\gamma \geq 0$ controls how strongly easy examples are down-weighted and α balances classes. While focal loss is naturally defined for differentiable models optimized by gradient descent, boosted trees commonly incorporate per-sample weights rather than directly optimizing focal-style terms in closed

form. We therefore implement a practical focal-proxy for XGB through iterative sample reweighting.

Concretely, we start from a class-balance weight vector wcb (to address base-rate imbalance), train an XGB model on the training split using these weights, and compute predicted probabilities {pi} on the training set. We then update each sample weight using focal-inspired factors that depend on model confidence: for fraud samples (yi = 1), the weight is scaled by a factor proportional to (1 − pi)γ with coefficient α, so positives that the model assigns low probability (hard positives) receive larger emphasis; for non-fraud samples (yi = 0), the weight is scaled by a factor proportional to pγ i with coefficient (1 −α), so negatives that the model assigns high probability (hard negatives) receive larger emphasis. Intuitively, this increases the relative influence of hard positives and hard negatives while reducing the influence of easy examples that are already handled confidently. Because repeated reweighting can produce extreme weights and unstable training dynamics, we clip weights to a fixed range after each update to prevent a small number of samples from dominating the fit. We repeat this train–predict reweight procedure for R rounds, yielding a final boosted model trained under the last updated weights. In our experiments we use R = 3, γ = 2.0, and α = 0.25, which are standard focal-inspired settings that emphasize hard examples while maintaining training stability [19]. This procedure is deliberately simple and compatible with standard XGB tooling, enabling an apples-to-apples comparison against the baseline XGB objective.

3.4 Reliability calibration

In fraud screening, predicted probabilities are often consumed directly as risk scores; thus calibration is essential when probabilities are used for prioritization and policy. Let z denote the model score before the sigmoid mapping (a logit for LR or a margin/score for XGB), and let p be the resulting uncalibrated probability estimate. We apply post-hoc calibration using the validation set only, preserving the test set for unbiased reporting. We compare three calibration families that are widely used in practice:

a. Platt scaling: a logistic mapping ˆp = σ(az + fit on validation data (Platt, 1999).
b. Temperature scaling: a single-parameter rescaling ˆp = σ(z/T) with T > 0, which is a strong and simple baseline for post-hoc calibration (Guo et al., 2017).
c. Monotone calibration: an isotonic-style monotone mapping and beta-calibration variants that provide a principled extension of logistic calibration while remaining easy to implement [15], [22].

Calibration is selected based on probabilistic correctness rather than discrimination: we choose the best calibration variant by the minimum validation Brier score (Eq. 5), a proper scoring rule for prob ability forecasts [20]. We additionally report ECE-style summaries and reliability behavior to provide complementary diagnostics, while noting that calibration evaluation depends on estimation choices (e.g., binning) and should be interpreted carefully [15]. Importantly, calibra tion is a post-hoc layer: it does not change the ranking induced by scores, but it can substantially change decision quality when thresholds and policies operate on probabilities.

3.5 Cost-sensitive thresholding and decision curve

Given an explicit cost matrix, deployment should choose an operating threshold that minimizes expected operational cost rather than defaulting to a fixed cutoff. We adopt an asymmetric cost setting with CFN = 10and CFP = 1, reflecting the common assumption that missing a fraud is substantially more expensive than reviewing a legitimate transaction [18]. For any threshold t, define FN(t) and FP(t) as the number of false negatives and false positives produced by classifying a transaction as fraud when ˆp≥ t. The expected cost on a split is

$$\text{Cost}(t) = C_{\text{FN}} \cdot \text{FN}(t) + C_{\text{FP}} \cdot \text{FP}(t) \quad (3)$$

We choose the validation-optimal threshold

$$t^* = \arg\min_t \text{Cost}(t), \quad (4)$$

then apply (ˆp,t) to the test set. This protocol isolates threshold selection from test reporting and explic itly connects model outputs to downstream decision costs. In addition to reporting the single operating point t, we plot a decision curve Cost(t) over a grid of thresholds. This curve makes the FN/FP trade off transparent and reveals whether cost is robust to small threshold perturbations (stable operation) or highly sensitive (brittle operation), which is important in rare-event regimes where small changes in threshold can significantly change alert volume.

3.6 Metrics

We report both ranking-oriented and decision-oriented metrics, along with calibration diagnostics, to capture the full pipeline behavior. For ranking performance, we report AUROC and AUPRC. In highly imbalanced problems, AUPRC is especially informative because it reflects precision at relevant recall levels and directly characterizes the usefulness of top-ranked alerts; the relationship between ROC and precision–recall clarifies why ROC summaries can obscure performance differences in rare-event set tings [1][1]. For operational performance at the selected operating point, we report fraud-class precision, recall, and F1 computed on the test set using the validation-chosen threshold t. This reflects how the system would behave when deployed under a fixed cost policy.

For calibration, we report Brier score and ECE-style summaries. Brier score is

$$\text{Brier} = \frac{1}{N}\sum_{i=1}^{N}(\hat{p}_i - y_i)^2, \quad (5)$$

which measures the mean squared error of probabilistic forecasts and is a proper scoring rule [20]. ECE is computed via probability binning and averaging the absolute gap between per-bin accu racy and confidence; it is widely used as a descriptive diagnostic but depends on design choices such as bin count and estimator, motivating careful interpretation

and consistent reporting [15]. Together, these metrics allow us to distinguish (i) models that rank well but produce unreliable probabilities from (ii) models that produce stable, decision-ready probabilities and therefore support cost-aware thresholding more effectively.

4 Result And Discussion

Experimental environment and software versions are summarized in Table 1. Reporting the environment is not a formality: for highly imbalanced detection tasks, small differences in library versions, solvers, or boosting implementations can alter convergence behavior, early-stopping decisions, and consequently the operating threshold and cost curves. In addition, calibration routines and metric computations (e.g., binning strategies for ECE) can differ subtly across implementations, which can lead to inconsistent conclusions if the experimental stack is not recorded. To support reproducibility and to make the re ported results auditable, we provide the exact software stack and key runtime information alongside the exported prediction, calibration, and metric artifacts. This practice is aligned with the broader emphasis in deployed risk modeling that the full decision pipeline (not just a single metric) should be repeatable and inspectable.

Item	Value
Python	3.12.12
Runtime	Linux (Colab), CPU (no GPU/TPU detected)
NumPy/SciPy/scikit-learn	2.0.2 / 1.17.0 / 1.5.2
XGBoost	3.1.3
Matplotlib	3.10.0

4.1 Initial data analysis

The dataset contains 284,807 transactions with 492 fraud cases (≈ 0.17%), making the task extremely imbalanced. This base rate immediately implies that accuracy is not meaningful and that the domi nant practical question is how to operate at a threshold that yields a manageable alert volume while capturing as much fraud as possible. From an operational perspective, extreme imbalance also means that most transactions are legitimate, so a small absolute increase in false positives can translate into a large increase in analyst workload. Conversely, because the positive class is rare, each missed fraud can be materially costly. These properties motivate cost-sensitive evaluation and thresholding rather than default probability cutoffs [18].

Asecond implication of the low base rate is that headline ranking metrics must be interpreted with care. While AUROC is commonly reported, it can obscure practical behavior in rare-event regimes; precision–recall analysis more directly reflects the quality of top-ranked alerts and the burden implied by false positives [1]. For example, two models with similar ROC curves may have very different precision at the recall levels that an organization can afford, and these differences become decisive when manual review is the bottleneck. This is also where probability calibration matters: if probabilities are miscalibrated, the same nominal threshold can imply very different alert volumes and true fraud rates across time, making operational control brittle [14], [15].

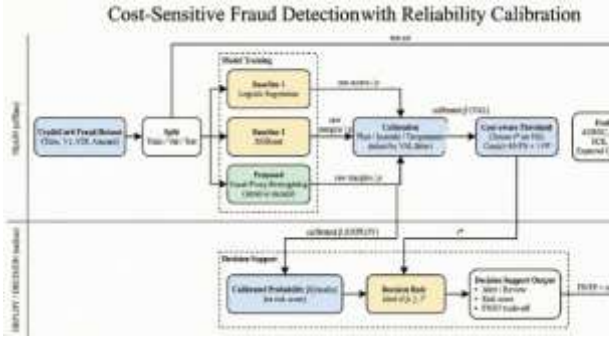
We use a chronological split into train/validation/test to reduce the risk of temporal leakage and to better approximate deployment, where models are trained on historical data and applied to future transactions. This split also supports a clean separation between (i) model fitting, (ii) calibration and threshold selection, and (iii) final reporting. In particular, calibration and threshold selection are per formed on the held-out validation split, because tuning these decision-layer components on the test set would inflate performance and obscure how the pipeline behaves under realistic selection pressure. This protocol is consistent with stream-oriented fraud detection settings where evolving patterns and delayed labels are common[5], [6] [6]. Finally, given the rarity of positives, we interpret ranking curves (PR/ROC) and decision curves as complementary views: ranking curves explain whether the model can prioritize fraud, while decision curves explain whether that prioritization translates into lower operational cost at a workable operating point.

4.2 Qualitative overview of the decision-support pipeline

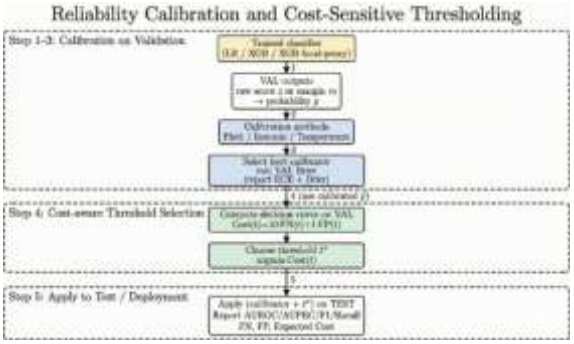
Figure 1 provides a qualitative overview of the decision-support workflow. Figure 1a illustrates the full end-to-end pipeline from offline training to deployment, emphasizing that the model output is used as a risk score rather than merely a binary prediction. Figure 1b zooms into the two decision-support steps that are often under-specified in fraud papers but are critical in practice: reliability calibration and cost-aware threshold selection.

The key design choice is that calibration and threshold selection are treated as model selection steps, not as post-hoc visualization. Platt scaling and related calibration families translate raw scores into probabilities that better match empirical fraud frequencies [14], [21]. This matters because operational policies frequently interpret the score as a probability-like quantity and use it to set thresholds, allocate review capacity, or trigger additional verification. Conceptually, calibration converts a model from “a ranker” into “a risk estimator” and allows stakeholders to interpret outputs in terms of expected fraud rates rather than arbitrary score units. After calibration, the operating threshold is selected by minimizing an explicit expected cost. This shifts the evaluation from a purely ranking-based story (“is fraud higher than non-fraud?”) to a decision-based story (“what is the cheapest way to operate given asymmetric error costs?”) [18]. The combined workflow therefore makes the decision logic auditable: if alert volume spikes or false negatives

increase, the system owner can trace whether the issue comes from ranking quality, calibration drift, or an ill-chosen threshold under the current cost regime. Framed this way, the fraud detector operates as an adaptive decision-support layer rather than a standalone classifier, which is consistent with broader security-framework thinking that separates sensing, analysis, and response/control components [26].



a. End-to-end workflow: train models offline, calibrate on validation, select a cost-aware threshold, and deploy calibrated risk scores.



b. Step-wise procedure for reliability calibration and cost-sensitive threshold selection.

Figure 1. Decision-support pipeline with reliability calibration and cost-aware thresholding.

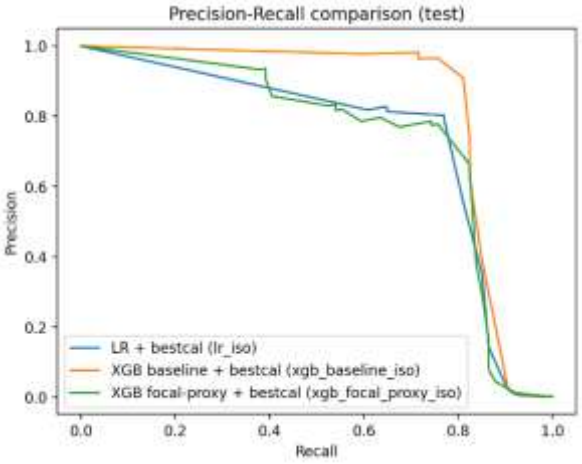
4.3 Mainquantitative Results

Table 2 reports the main ablation on the test set. Overall, the tree-ensemble baseline (XGBoost) outper forms Logistic Regression on both ranking and operational metrics, which is consistent with the broader evidence that boosted trees are strong baselines for heterogeneous tabular data [7]. This outcome is not surprising: non-linear ensembles can model higher-order feature interactions that a linear decision boundary cannot capture, and they typically deliver superior separability in complex tabular settings. However, in an imbalanced fraud setting, the most consequential differences arise in the region of the curve that corresponds to deployable thresholds: it is not enough to perform well on average; the model must be able to concentrate fraud cases near the top of the ranked list so that a limited review budget yields meaningful fraud capture. This is precisely the regime where precision–recall be havior is more informative than ROC summaries [1]. The reported AUROC and AUPRC therefore serve as high-level indicators, while the downstream decision metrics at the chosen operating threshold capture how the pipeline would behave when deployed under a cost policy.

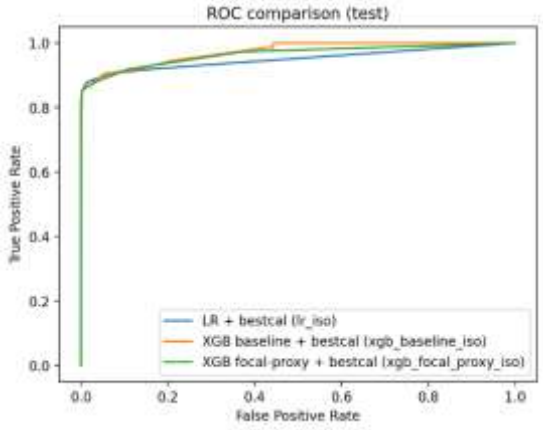
Calibration has a distinct but complementary role. Post-hoc calibration substantially reduces ECE and typically improves (or at least stabilizes) Brier score, indicating more reliable probability estimates [14], [15], [20]. Importantly, calibration does not change the ranking of examples (in the monotone case), so its primary benefit is in decision consistency: a cali brated score is more interpretable as risk, supports more stable alert thresholds, and reduces the chance that a fixed threshold corresponds to wildly different empirical fraud rates over time. This is a practical consideration in fraud, where drift and policy changes can cause score distributions to shift [6]. In our results, the minimum expected cost is primarily governed by two factors: the un derlying ranking quality (how well the model places fraud above non-fraud) and the threshold selection rule under the chosen cost ratio [18]. Calibration improves the reliability of the score-to-risk mapping, which is essential for operational use, but it cannot compensate for weak ranking at the top of the list.

Model	t	AUROC	AUPRC	$Prec_{fraud}$	Rec_{fraud}	F1
XGBbaseline (uncal.)	0.201	0.9706	0.8256	0.8108	0.8108	0.8108
XGBbaseline + bestcal (iso.)	0.059	0.9734	0.8116	0.7176	0.8243	0.7673
XGBfocal-proxy (uncal.)	0.982	0.9722	0.7337	0.6489	0.8243	0.7262
XGBfocal-proxy + bestcal (iso.)	0.033	0.9650	0.7169	0.6489	0.8243	0.7262
LR+bestcal (iso.)	0.059	0.9502	0.6697	0.5556	0.8108	0.6593

We summarize ranking behavior using Precision–Recall and ROC curves in Figure 2. In this rare event regime, PR curves directly show how precision decays as recall increases, and they highlight whether a model can maintain useful precision at the recall levels that an organization might target given finite review capacity [1]. ROC curves are included for completeness and comparability with prior work, but we interpret them cautiously in light of imbalance. Calibration and operational decision behavior are visualized with reliability diagrams and validation decision curves in Figure 3. Reliability diagrams reveal whether predicted probabilities are systematically overconfident or underconfident; decision curves reveal how expected cost changes as the threshold varies, providing a direct view of sensitivity to threshold choice under asymmetric costs [14], [15], [18]. Together, these plots diagnose whether a model is merely a good ranker or a robust decision component.

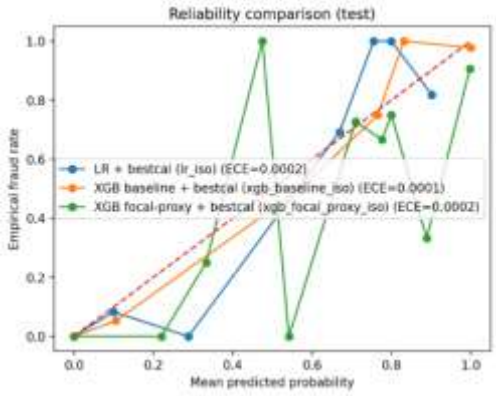


a. Precision–Recall curves.

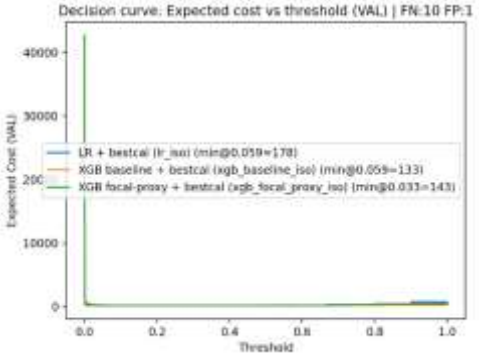


b. ROC curves.

Gambar 2. *Ranking performance for selected models.*



a. Reliability diagrams (calibration).



b. Decision curve on validation: expected cost vs. threshold.

Gambar 3. *Calibration and operational decision behavior.*

4.4 Discussion: what improves decisions and what does not

The results highlight a practical separation of concerns. First, discrimination (ranking) determines how much fraud can be concentrated in the top-scored region; this largely governs how much value any downstream policy can extract from the model. Second, calibration governs whether the score has a stable probabilistic interpretation, which affects threshold stability, comparability across time, and policy robustness [14]. Third, cost-aware thresholding defines the operating point that trades false negatives against false positives according to the business policy [18]. In other words, ranking enables effective prioritization, calibration enables reliable risk interpretation, and cost aware thresholding enables policy-aligned decisions. Any improvement proposal should be evaluated by asking which of these three components it targets and whether the claimed gains translate into lower expected cost at deployable operating points.

These findings also contextualize why focal-inspired reweighting does not automatically guarantee lower operational cost in boosted-tree settings. While focal loss is designed to emphasize hard exam ples [19], iterative reweighting can also increase variance, amplify noise, or shift the score distribution in ways that reduce precision in the high-confidence region precisely where fraud screen ing is most sensitive. This is consistent with broader imbalance literature showing that reweighting and resampling can improve recall while simultaneously degrading precision, and the net effect depends on the operating regime and cost policy [2], [3], [5]. In practice, focal-proxy reweighting may need additional safeguards (e.g., stronger clipping, alternative schedules for γ, or drift-aware validation) to ensure that harder-example emphasis does not overfit rare patterns or noise. This underscores why decision curves are essential: a method can improve one portion of the curve while degrading the cost-minimizing region that determines deployment utility.

Finally, it is important to interpret calibration improvements correctly. Lower ECE and Brier indicate better probability reliability [15], [20], but reliability alone does not guarantee better decisions if the ranking quality is insufficient. Conversely, a strong ranker with poor calibration may still yield acceptable decisions if the threshold is tuned on recent data, but such a system will be brittle and hard to maintain as conditions change. The goal of the proposed pipeline is to reduce this brittleness by explicitly separating training, calibration, and cost-aware thresholding and by providing diagnostics that reveal where failures originate.

4.5 Error analysis

To better understand failure modes, we examine false negatives (missed frauds) and false positives (le gitimate transactions flagged as fraud) for the best-calibrated LR and the best-calibrated XGB baseline. Because false negatives incur a higher cost under our policy, these cases are especially important: they indicate where the model assigns low risk to transactions that are actually fraudulent. Such errors can arise from overlap in feature space, limited representation of fraud patterns in the training set, or drift between training and test periods [6]. Conversely, false positives quantify analyst load and customer friction; reducing false positives at a fixed recall has direct operational value because it decreases manual workload and improves the precision of escalations.

Qualitatively, the calibrated XGB baseline produces substantially fewer false positives than LR at a similar fraud recall, implying lower analyst workload for a fixed cost ratio. This aligns with the expec tation that non-linear ensembles can better separate the majority class near the decision boundary and reduce spurious high-risk assignments [7]. To support transparency and follow-up analysis, the full top-50 false-negative and false-positive lists for the selected models are included in the accompanying CSV artifacts (exported run outputs). These artifacts enable practitioners to inspect whether errors concentrate in specific transaction ranges (e.g., high Amount or particular anonymized component patterns) and to formulate targeted mitigation strategies such as rule-based overrides, tar geted feature augmentation, or policy adjustments that change review criteria or cost ratios.

5. Comparison

5.1 Comparison dimensions

Ameaningful "state-of-the-art" comparison in fraud detection should consider more than AUROC, be cause fraud screening is ultimately evaluated by how well it supports actions under operational con straints. In particular, an offline model that looks strong on an aggregate ranking metric can still be unusable in deployment if it produces unstable alert volumes, unreliable probability estimates, or thresh olds that do not align with the organization's cost policy. For this reason, we frame comparison along multiple dimensions that map directly to deployment.

First, we consider ranking quality under extreme imbalance. When fraud prevalence is far be low 1%, the relevant operating regime is the top of the ranked list, where analysts will actually review transactions. Precision–recall behavior (and thus AUPRC) is typically more informative than ROC sum maries because it reflects how precision changes as recall increases in rare-event settings [1]. Second, we consider probability reliability. Many operational systems treat model outputs as risk scores for prioritization and policy rules; if these probabilities are miscalibrated, the same nominal threshold can yield different empirical fraud rates over time and cause unstable review volume. Calibration and reliable risk estimation are therefore central components of deployment, not optional diagnostics [14], [15]. We quantify reliability using both Brier score, a proper scoring rule for probabilistic forecasts [20], and ECE-style summaries, which provide an interpretable diagnostic of confidence–accuracy gaps [15].

Third, we consider cost-aware operating points. In practice, error costs are asymmetric: missing fraud (false negatives) is typically more expensive than reviewing legitimate transactions (false posi tives). Cost-sensitive learning formalizes this setting and motivates selecting operating thresholds by minimizing expected cost rather than defaulting to a fixed probability cutoff [18]. Decision curves (expected cost vs. threshold) make the FN/FP trade-off explicit and reveal whether the optimum is stable under small threshold perturbations, which is important when alert volume must be controlled. Fourth, we consider operational burden. Even when two models achieve similar recall, the one that produces fewer false positives implies lower analyst load, reduced customer friction, and lower down stream costs. Reviews of fraud detection emphasize that such deployment-relevant factors are often missing from purely metric-driven comparisons [24]. While offline benchmarks cannot fully capture "false positives per day" without explicit throughput assumptions, re porting false positive counts at the chosen threshold and the sensitivity of alert volume to threshold shifts provides a practical proxy for operational burden.

Overall, many papers report only ranking metrics; fewer report probability calibration and cost aware thresholding, even though fraud decisions are explicitly cost-sensitive and probability outputs are frequently used as risk scores [14], [24]. This study there fore emphasizes probability reliability and cost-aware operation in addition to standard discrimination metrics, aligning the comparison with how fraud models are actually deployed.

5.2 Positioning vs. common baselines

Ontabular datasets such as credit card transactions, gradient-boosted decision trees are widely regarded as strong baselines and are frequently competitive with more complex tabular deep learning models [7], [8], [9], [13]. This competitiveness stems from their ability to model non-linear feature

interactions efficiently and their robustness to heterogeneous feature scales. Our results are consistent with this broader evidence: XG Boost variants outperform Logistic Regression on AUROC/AUPRC and achieve lower expected cost under the chosen cost matrix (Table 2). Importantly, this advantage is not merely an "average metric" effect; it manifests in the deployable region where precision must remain workable at meaningful re call levels. As discussed earlier, this region is best reflected by precision–recall analysis in rare-event regimes [1].

In positioning our work relative to the broader tabular-learning landscape, it is also important to clarify what we are not claiming. We do not claim architectural novelty relative to the rapidly growing literature on deep tabular models. Rather, we treat strong GBDT baselines as the appropriate reference and focus on decision readiness: calibration, cost-aware thresholding, and reproducible evaluation. This emphasis complements the deep tabular literature by addressing the layers that often determine deploy ment success even when the underlying classifier family is competitive. That said, future work should expand the baseline set to include representative deep tabular methods such as TabNet, TabTransformer, and attention-based tabular transformers, and then evaluate whether calibration behavior and cost-aware thresholding remain consistent across model classes [10], [11], [12], [13]. Such a study would help disentangle performance gains due to rep resentation learning from gains due to decision-layer improvements.

Finally, stronger "classical" baselines beyond XGBoost can also be considered under identical pro tocols. LightGBM and CatBoost offer alternative boosting implementations with different tree-growth strategies and regularization behaviors, and they often provide competitive performance and computa tional advantages in tabular settings [8], [9]. Incorporating them would strengthen generality claims about decision-layer improvements (calibration and threshold selec tion) across boosting families.

5.3 Effect of focal-proxy reweighting

Within this benchmark and the tested hyperparameter setting, focal-proxy reweighting does not improve expected cost compared to the best-calibrated XGBoost baseline and reduces AUPRC. This outcome is informative because it highlights that emphasizing "hard" examples is not automatically beneficial in rare-event fraud screening. Focal loss was originally proposed to down-weight easy negatives and focus learning on difficult cases in dense detection problems [19]. However, in extremely imbalanced tabular fraud data, oversuppressing easy negatives can distort the score distribution in the region that matters most (the highest-risk tail), reducing precision among top-ranked alerts and thereby harming decision utility. Put differently, reweighting can trade recall for precision in a way that increases operational burden, even if certain aggregate metrics remain stable.

This suggests that focal-style reweighting in boosted-tree settings may require more careful control of weight dynamics. Targeted extensions include tuning (R,γ,α), experimenting with more conser vative schedules (e.g., gradually increasing γ), and tightening weight clipping to avoid a small set of samples dominating the fit. Additionally, since imbalance treatments interact with probability estima tion, it is valuable to benchmark focal-proxy reweighting against modern resampling protocols under identical chronological splits and identical calibration/thresholding procedures. A key principle for such comparisons is that any change to training distribution (reweighting or resampling) should be evaluated not only by discrim ination but also by calibration and expected cost, because the downstream decision layer is explicitly cost-sensitive [14], [18]. This multi-dimensional evaluation avoids concluding that a method "improves fraud detection" when it merely shifts errors in a way that increases operational workload.

6. Conclusions

This paper presented an end-to-end fraud detection study on an extremely imbalanced credit card bench mark from the ULB dataset family, with the explicit goal of aligning offline evaluation with deployment reality. Rather than treating fraud detection as a pure ranking task, we framed it as a cost-sensitive deci sion problem in which predicted probabilities are consumed as risk scores for prioritization and policy. Under this framing, two aspects become deployment-critical: (i) probability reliability, because miscal ibration makes thresholds unstable and undermines risk-based operations, and (ii) cost-aware thresh olding, because operational decisions must reflect asymmetric error costs rather than arbitrary probability cutoffs [14], [18]. Accordingly, we implemented a reproducible pipeline that cleanly separates training, calibration, and threshold selection: Logistic Regression and gradient boosted trees (XGBoost) were trained as strong tabular baselines [7]; calibration was fit exclusively on a held-out validation split using Platt scaling, temperature scaling, and monotone calibration families [14], [15], [21], [23] and the final operating threshold was selected by minimizing expected cost on validation under a 10:1 false-negative to-false-positive cost matrix [18]. Final reporting was performed on a held-out test split to avoid optimistic bias.

Empirically, the best-calibrated XGBoost baseline achieved the lowest expected cost while also maintaining strong ranking performance in the rare-event regime. This result is consistent with the broader evidence that boosted tree ensembles are strong and practical baselines for heterogeneous tabular problems [7], [8], [9], [13]. Post-hoc calibration substantially reduced calibration error (as reflected by ECE-style diagnostics) and improved or

maintained Brier score, supporting the view that calibration is not merely a reporting add-on but a necessary step when probabilities are used as stable risk scores for decision support [14], [20]. In contrast, the proposed focal-proxy reweighting scheme for boosted trees did not yield gains under the evaluated configuration and reduced AUPRC, suggesting that focal-style emphasis on hard examples can distort the score distribution in the high-confidence region that matters most for fraud screening unless the reweighting dynamics are carefully controlled [1], [19]. This negative result is practically useful: it cautions against assuming that reweighting strategies that succeed in other detection domains will automatically improve decision utility under extreme imbalance, especially when precision at the top of the ranked list governs operational workload.

Several extensions would strengthen generality and move the pipeline closer to production robust ness. First, the workflow should be validated on additional fraud datasets and stream settings to assess sensitivity to drift and to confirm that calibration and cost-aware thresholding behave consistently across domains [6], [24]. Second, the cost policy itself should be treated as context-dependent: adaptive or segment-specific cost ratios (e.g., by merchant category, transaction amount, or customer segment) can better reflect real operational trade-offs than a single global matrix, and thresholding should be evaluated under such policies [18]. Third, a more comprehen sive "state-of-the-art" comparison should include additional strong baselines beyond XGBoost, such as LightGBM and CatBoost for boosting families [8], [9], as well as representative deep tabular models (e.g., TabNet, TabTransformer, SAINT, and related transformer based baselines) to understand whether decision-layer benefits transfer across model classes [10], [11], [12], [13]. Finally, because re liability under dataset shift remains a central concern for risk scoring, future work should explicitly stress-test calibration and decision stability under distribution shift scenarios, drawing on broader uncer tainty and calibration perspectives [16], [17]. Overall, the main takeaway is that fraud detection pipelines should be evaluated as decision systems: strong ranking is necessary but insufficient, reliable probabilities and cost-aligned operating points are what ultimately determine deployment value.

References

[1] J. Davis and M. Goadrich, "The Relationship Between Precision-Recall and ROC Curves," in *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, 2006.

[2] M. Carvalho, A. J. Pinho, and S. Brás, "Resampling Approaches to Handle Class Imbalance: A Review from a Data Perspective," *J. Big Data*, vol. 12, no. 1, p. 71, 2025.

[3] G. Haixiang, L. Yijing, S. Jin, G. Mingyun, H. Yuanyue, and G. Bing, "Learning from Class-Imbalanced Data: Review of Methods and Applications," *Expert Syst. Appl.*, vol. 73, pp. 220–239, 2017.

[4] J. M. Johnson and T. M. Khoshgoftaar, "Survey on Deep Learning with Class Imbalance," *J. Big Data*, vol. 6, no. 1, p. 27, 2019.

[5] A. D. Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Calibrating Probability with Undersampling for Unbalanced Classification," in *2015 IEEE Symposium Series on Computational Intelligence*, 2015.

[6] F. Carcillo, A. D. Pozzolo, Y.-A. Le Borgne, O. Caelen, Y. Mazzer, and G. Bontempi, "Stream Based Active Learning for Real-Life Credit Card Fraud Detection: A Comparative Study," *Data Min. Knowl. Discov.*, vol. 35, pp. 1–29, 2021.

[7] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, 2016, pp. 785–794.

[8] G. Ke *et al.*, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[9] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased Boosting with Categorical Features," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[10] S. O. Arik and T. Pfister, "TabNet: Attentive Interpretable Tabular Learning," *Proc. AAAI Conf. Artif. Intell.*, 2021.

[11] X. Huang, A. Khetan, M. Cvitkovic, and Z. Karnin, "TabTransformer: Tabular Data Modeling Using Contextual Embeddings,"

arXiv Prepr., 2020.

[12] G. Somepalli, M. Goldblum, A. Schwarzschild, C. Bruss, and T. Goldstein, "SAINT: Improved Neural Networks for Tabular Data via Row Attention and Contrastive Pre-Training," *arXiv Prepr.*, 2021.

[13] Y. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko, "Revisiting Deep Learning Models for Tabular Data," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[14] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On Calibration of Modern Neural Networks," in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.

[15] D. Widmann, F. Lindsten, J. Roll, and T. B. Schön, "Evaluating Classifier Calibration," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[16] A. Kumar, P. Liang, and T. Ma, "Verified Uncertainty Calibration," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[17] Y. Ovadia *et al.*, "Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[18] C. Elkan, "The Foundations of Cost-Sensitive Learning," in *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI)*, 2001, pp. 973–978.

[19] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988.

[20] G. W. Brier, "Verification of Forecasts Expressed in Terms of Probability," *Mon. Weather Rev.*, vol. 78, no. 1, pp. 1–3, 1950.

[21] J. Platt, "Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods," *Adv. Large Margin Classif.*, pp. 61–74, 1999.

[22] M. Kull, T. M. S. Filho, and P. Flach, "Beta Calibration: A Well-Founded and Easily Implemented Improvement on Logistic Calibration for Binary Classifiers," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, PMLR, 2017, pp. 623–631.

[23] M. Kull, T. S. Filho, and P. Flach, "Beyond Temperature Scaling: Obtaining Well-Calibrated Multi-Class Probabilities with Dirichlet Calibration," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[24] W. Cherif, Y. Madani, and L. Khoukhi, "Credit Card Fraud Detection: Methods, Datasets, and Research Challenges," *IEEE Access*, 2023.

[25] V. N. Dornadula and S. Geetha, "Credit Card Fraud Detection Using Machine Learning Algorithms," *Procedia Comput. Sci.*, vol. 165, pp. 631–641, 2019.

[26] D. Danang, T. Wahyono, I. Sembiring, T. Wellem, and N. H. Dzulkefly, "An Adaptive Framework Integrating ML Blockchain and TEE for Cloud Security," in *2025 4th International Conference on Creative Communication and Innovative Technology (ICCIT)*, IEEE, 2025, pp. 1–7.