International Journal of Natural Sciences and Engineering Innovations
Vol. 1, No. 2 (2026)
https://globalacadbridge.org/

Advanced Machine Learning Models for Credit Card Fraud Detection

Chang Meng[1], Dewei Chen[2], Linxuan Guo[3], Zhaoyang Yu[4]

[1]School of Management, Zhejiang University of Finance & Economics, China
[2]College of Civil Engineering and Architecture, Zhejiang University, China
[3]School of Management, Zhejiang University of Finance & Economics, China
[4]School of Accounting, Zhejiang University of Finance & Economics, China

Email: Chang Meng, 240120430114@zufe.edu.cn

Abstract

Credit card fraud detection is a cost-sensitive learning problem characterized by extreme class imbalance, non-stationary adversarial behavior, and stringent operational constraints on false alarms. Using the publicly available CREDITCARDFRAUD-ULB benchmark of European cardholder transactions, we develop and evaluate a family of advanced deep learning models specialized for continuous tabular data. Our framework combines three ingredients: (i) a cost-sensitive residual multilayer perceptron (RESMLP) that provides a strong supervised baseline; (ii) a feature-tokenizing transformer (FT-TRANSFORMER) that contextualizes each transaction attribute through self-attention; and (iii) an innovative self-supervised pre-training strategy (FRAUDCL-FTT) that couples masked feature modeling with contrastive representation learning prior to supervised fine-tuning. We formulate fraud detection as a calibrated risk scoring problem and therefore evaluate models not only by ranking metrics such as AUROC and AUPRC, but also by probability calibration and decision-theoretic threshold selection. The resulting manuscript is designed to be fully reproducible: the accompanying code automatically trains the models, computes metrics, and exports publication-ready tables and figures.

Keywords: fraud detection; imbalanced classification; tabular deep learning; transformer models; self-supervised learning; calibration; cost-sensitive decision making.

1 Introduction

Credit card fraud detection remains one of the most practically significant benchmark problems in applied machine learning. From a statistical perspective, it is challenging because fraudulent transactions are extremely rare relative to legitimate ones, which induces severe class imbalance and renders naive accuracy-based evaluation essentially meaningless. From an operational perspective, the problem is equally demanding because false negatives correspond to direct financial loss, while false positives consume analyst time, trigger customer friction, and can degrade trust in payment systems. As a consequence, an effective fraud detector must satisfy three distinct requirements simultaneously: strong ranking performance, stable probability estimates, and controllable behavior under deployment-specific alert budgets.

The CREDITCARDFRAUD-ULB benchmark is widely used for research and methodological comparison: it contains $284\,807$ transactions collected over two days, of which only 492 are labeled as fraud (approximately $0.172\,\%$). The features consist of anonymized principal components $V_1, \ldots, V_{28}$ together with the raw variables `Time` and `Amount`. [10, 9] Although this representation limits interpretability, it creates a clean

setting for evaluating methods that must detect sparse, difficult-to-separate anomalies from mostly continuous inputs.

Recent progress in tabular deep learning suggests that carefully structured neural architectures—especially residual MLPs and transformer-based feature tokenizers—can compete with or exceed classical tree-based methods in many regimes [5, 7]. However, performance in fraud detection cannot be summarized by AUROC alone. Under strong skew, the precision–recall geometry becomes more informative [4]; moreover, in financial decision systems, a model's calibrated posterior probabilities often matter as much as its raw ranking ability [6]. These considerations motivate a more comprehensive formulation of the problem: not merely classification, but calibrated, cost-aware risk estimation under scarcity of positives.

Contributions.

- **A formal cost-sensitive formulation.** We cast fraud detection as minimization of expected asymmetric decision cost under an imbalanced data-generating distribution.
- **A strong supervised deep baseline.** We design RESMLP, a residual MLP with normalization, dropout, and focal/class-weighted objectives for highly skewed binary classification.
- **A transformer for continuous tabular inputs.** We use FT-TRANSFORMER, which maps each scalar feature into a token embedding and models cross-feature interactions through self-attention.
- **A richer pretraining scheme.** We propose FRAUDCL-FTT, a self-supervised pretraining strategy that jointly optimizes masked feature modeling and contrastive alignment across stochastic corruptions of each transaction.
- **Calibration-aware evaluation.** We report discrimination, precision–recall behavior, calibration, and threshold selection under explicit cost or alert-budget constraints.
- **Reproducibility.** The accompanying pipeline automatically generates LaTeX-ready tables and figures, enabling end-to-end replication.

2 Related Work

Fraud detection. Fraud detection has been studied from both machine learning and operational perspectives, including adaptive modeling, delayed labels, verification cost, and distribution shift [2, 3]. In practice, the objective is not merely to maximize a scalar metric, but to prioritize suspicious transactions in a way that aligns with limited investigative capacity.

Imbalanced learning. Class imbalance has motivated a broad family of techniques, including reweighting, resampling, synthetic oversampling, and hard-example mining. SMOTE and related oversampling procedures are classical tools [1], while focal loss has become a standard approach for emphasizing difficult minority-class samples [8]. For evaluation, precision–recall analysis is often preferable to ROC analysis when positives are rare [4].

Deep learning for tabular data. Tabular deep learning has evolved from generic MLP baselines to architectures with inductive biases tailored to heterogeneous features. Residual MLP variants remain strong and efficient; transformer-based models such as FT-TRANSFORMER further introduce contextualized feature representations via attention [5]. TabTransformer and related methods demonstrate that feature tokenization can improve robustness and expressive power in structured data settings [7].

Calibration. In cost-sensitive decision systems, calibrated probabilities are crucial because threshold policies operate on posterior scores rather than on rank alone. Temperature scaling provides a simple and effective post-hoc calibration procedure for modern neural networks [6].

3 Dataset and Problem Formulation

3.1 Notation and Data Distribution

Let

$$\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}, \qquad \mathbf{x}_i \in \mathbb{R}^d, \quad y_i \in \{0, 1\},$$

denote the transaction dataset, where $y_i = 1$ indicates fraud and $y_i = 0$ indicates a legitimate transaction. In the present benchmark, $N = 284\,807$ and $d = 30$, corresponding to the variables $\{\texttt{Time}, \texttt{Amount}, V_1, \ldots, V_{28}\}$. The empirical fraud prevalence is

$$\hat{\pi} = \frac{1}{N} \sum_{i=1}^{N} y_i \approx 0.00172,$$

which makes the problem strongly skewed toward the negative class.

The learning objective is to estimate a score function

$$s_\theta : \mathbb{R}^d \to \mathbb{R},$$

parameterized by θ, together with a calibrated posterior probability

$$\hat{p}_\theta(\mathbf{x}) = \sigma(s_\theta(\mathbf{x})) \approx \mathbb{P}(Y = 1 \mid \mathbf{X} = \mathbf{x}),$$

where $\sigma(z) = 1/(1 + e^{-z})$ is the logistic sigmoid.

3.2 Decision-Theoretic Objective

Let $c_{\mathrm{FN}} > 0$ and $c_{\mathrm{FP}} > 0$ denote the costs of a false negative and false positive, respectively. A threshold policy $\delta_\tau(\mathbf{x}) = \mathbb{I}\{\hat{p}_\theta(\mathbf{x}) \geq \tau\}$ induces expected decision risk

$$\mathcal{R}(\tau; \theta) = c_{\mathrm{FN}}\, \mathbb{P}\big(\delta_\tau(\mathbf{X}) = 0, Y = 1\big) + c_{\mathrm{FP}}\, \mathbb{P}\big(\delta_\tau(\mathbf{X}) = 1, Y = 0\big).$$

Under perfectly calibrated probabilities, the Bayes-optimal threshold is

$$\tau^\star = \frac{c_{\mathrm{FP}}}{c_{\mathrm{FP}} + c_{\mathrm{FN}}},$$

which makes calibration directly relevant to downstream operations.

3.3 Preprocessing and Splitting

We transform the heavy-tailed monetary attribute by

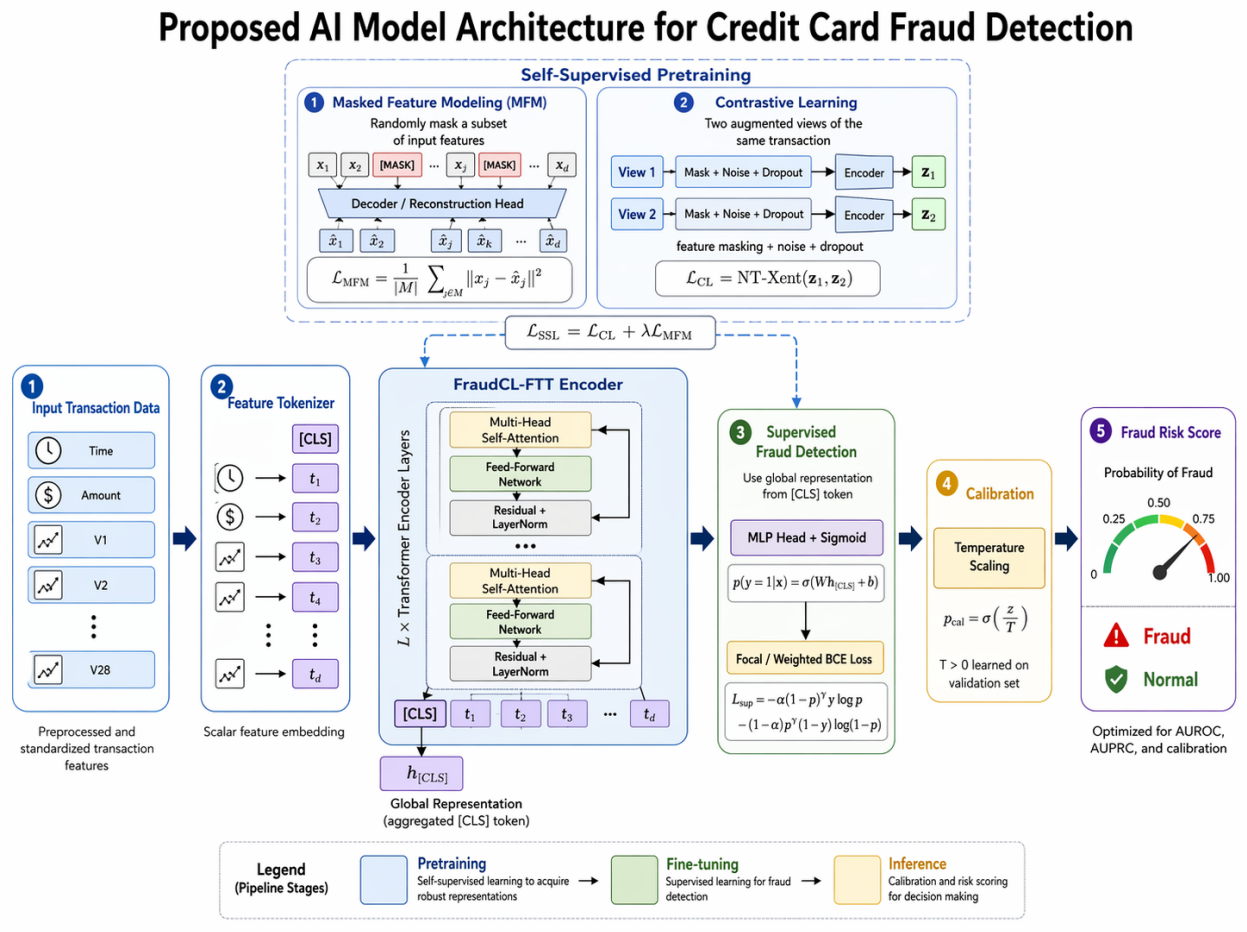
$$x_{\mathrm{Amount}} \leftarrow \log\big(1 + x_{\mathrm{Amount}}\big),$$

and standardize each feature using training-only statistics:

$$\tilde{x}_{ij} = \frac{x_{ij} - \mu_j^{\text{train}}}{\sigma_j^{\text{train}} + \varepsilon}.$$

To reduce temporal leakage, our preferred evaluation protocol is a time-ordered split in which earlier transactions are used for training and later ones for validation and test. A stratified random split may additionally be reported for comparability with prior studies.

4 Methods



4.1 Supervised Learning Objective

Given model logits $z_i = s_\theta(\mathbf{x}_i)$ and probabilities $\hat{p}_i = \sigma(z_i)$, standard empirical risk minimization uses the binary cross-entropy

$$\mathcal{L}_{\text{BCE}}(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\Big[y_i \log \hat{p}_i + (1-y_i)\log(1-\hat{p}_i)\Big].$$

Because fraud is rare, we employ cost-sensitive variants of this objective.

Class-weighted log loss. Let $w_1 > w_0 > 0$ denote positive and negative class weights. The weighted binary loss is

$$\mathcal{L}_{\text{wBCE}}(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\Big[w_1 y_i \log \hat{p}_i + w_0(1-y_i)\log(1-\hat{p}_i)\Big].$$

Focal loss. To emphasize hard or misclassified examples, we also consider focal loss [8]:

$$\mathcal{L}_{\text{focal}}(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\left[\alpha\, y_i(1-\hat{p}_i)^\gamma \log \hat{p}_i + (1-\alpha)(1-y_i)\hat{p}_i^\gamma \log(1-\hat{p}_i)\right],$$

where $\alpha \in (0,1)$ balances the positive class and $\gamma \geq 0$ controls the focusing strength.

4.2 Cost-Sensitive Residual MLP (RESMLP)

Let $\mathbf{x} \in \mathbb{R}^d$ denote a standardized transaction vector. The input is first projected into a hidden representation:

$$\mathbf{h}^{(0)} = W_{\text{in}}\mathbf{x} + \mathbf{b}_{\text{in}}, \qquad \mathbf{h}^{(0)} \in \mathbb{R}^m.$$

Each residual block applies pre-normalization, a two-layer nonlinear map, and a skip connection:

$$\tilde{\mathbf{h}}^{(\ell)} = \text{LN}\Big(\mathbf{h}^{(\ell)}\Big), \tag{1}$$

$$\mathbf{u}^{(\ell)} = W_2^{(\ell)}\,\phi\Big(W_1^{(\ell)}\tilde{\mathbf{h}}^{(\ell)} + \mathbf{b}_1^{(\ell)}\Big) + \mathbf{b}_2^{(\ell)}, \tag{2}$$

$$\mathbf{h}^{(\ell+1)} = \mathbf{h}^{(\ell)} + \text{Dropout}\Big(\mathbf{u}^{(\ell)}\Big), \tag{3}$$

for $\ell = 0, \ldots, L-1$, where $\phi(\cdot)$ is GELU and LN denotes layer normalization. The final fraud logit is

$$z = \mathbf{w}_{\text{out}}^\top \text{LN}\Big(\mathbf{h}^{(L)}\Big) + b_{\text{out}}.$$

This architecture preserves the efficiency and stability of MLPs while increasing expressive power through depth and residual refinement.

4.3 Feature Tokenizer Transformer (FT-TRANSFORMER)

For continuous tabular data, each scalar feature is mapped into an embedding token. Let x_j be the j-th feature of $\mathbf{x}$. The tokenizer computes

$$\mathbf{t}_j = x_j\mathbf{w}_j + \mathbf{b}_j, \qquad \mathbf{w}_j, \mathbf{b}_j \in \mathbb{R}^m,$$

for $j = 1, \ldots, d$. We prepend a learnable classification token $\mathbf{t}_{\text{cls}} \in \mathbb{R}^m$ and define the input token matrix

$$T^{(0)} = \begin{bmatrix} \mathbf{t}_{\text{cls}} \\ \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_d \end{bmatrix} \in \mathbb{R}^{(d+1)\times m}.$$

Each transformer layer applies multi-head self-attention and a feed-forward network with residual connections:

$$Q_h = T^{(\ell)}W_h^Q, \qquad K_h = T^{(\ell)}W_h^K, \qquad V_h = T^{(\ell)}W_h^V, \tag{4}$$

$$\text{Attn}_h\Big(T^{(\ell)}\Big) = \text{softmax}\Big(\frac{Q_hK_h^\top}{\sqrt{m_h}}\Big)V_h, \tag{5}$$

$$\text{MSA}\Big(T^{(\ell)}\Big) = \text{Concat}\big(\text{Attn}_1, \ldots, \text{Attn}_H\big)W^O, \tag{6}$$

$$\bar{T}^{(\ell)} = T^{(\ell)} + \text{MSA}\Big(\text{LN}(T^{(\ell)})\Big), \tag{7}$$

$$T^{(\ell+1)} = \bar{T}^{(\ell)} + \text{FFN}\Big(\text{LN}(\bar{T}^{(\ell)})\Big). \tag{8}$$

After L layers, the contextualized classification token $\mathbf{c} = T_{0,:}^{(L)}$ is fed to an MLP head:

$$z = g(\mathbf{c}),$$

which produces the fraud logit. The key modeling advantage is that attention learns pairwise and higher-order cross-feature interactions without requiring explicit manual feature engineering.

4.4 Self-Supervised Pretraining with FRAUDCL-FTT

To strengthen representations under extreme label sparsity, we introduce a two-view self-supervised pretraining stage before supervised fine-tuning. Let $\mathcal{A}(\mathbf{x};\omega)$ denote a stochastic corruption operator parameterized by random seed ω, combining feature masking, feature dropout, and small Gaussian noise:

$$\tilde{\mathbf{x}}^{(1)} = \mathcal{A}(\mathbf{x};\omega_1), \qquad \tilde{\mathbf{x}}^{(2)} = \mathcal{A}(\mathbf{x};\omega_2).$$

The encoder $E_\theta(\cdot)$ is instantiated by the transformer backbone.

Masked feature modeling (MFM). Let $M \subseteq \{1,\dots,d\}$ denote the masked coordinates. From the contextual token embeddings, a reconstruction head predicts the original feature values on the masked subset:

$$\hat{x}_j = r_\psi(E_\theta(\tilde{\mathbf{x}})_j)\,, \qquad j \in M.$$

The reconstruction loss is

$$\mathcal{L}_{\mathrm{MFM}}(\theta,\psi) = \frac{1}{|M|}\sum_{j\in M}(\hat{x}_j - x_j)^2\,.$$

Contrastive representation learning. Let $\mathbf{z}^{(1)} = g_\varphi(E_\theta(\tilde{\mathbf{x}}^{(1)}))$ and $\mathbf{z}^{(2)} = g_\varphi(E_\theta(\tilde{\mathbf{x}}^{(2)}))$ be projected representations of two stochastic views of the same transaction, normalized onto the unit sphere. For a mini-batch of size B, the NT-Xent objective is

$$\mathcal{L}_{\mathrm{CL}} = -\frac{1}{2B}\sum_{i=1}^{B}\left[\log\frac{\exp(\mathrm{sim}(\mathbf{z}_i^{(1)},\mathbf{z}_i^{(2)})/\tau)}{\sum_{k\neq i}\exp(\mathrm{sim}(\mathbf{z}_i^{(1)},\mathbf{z}_k^{(2)})/\tau)} + \log\frac{\exp(\mathrm{sim}(\mathbf{z}_i^{(2)},\mathbf{z}_i^{(1)})/\tau)}{\sum_{k\neq i}\exp(\mathrm{sim}(\mathbf{z}_i^{(2)},\mathbf{z}_k^{(1)})/\tau)}\right],$$

where $\mathrm{sim}(\mathbf{a},\mathbf{b}) = \mathbf{a}^\top\mathbf{b}/(\|\mathbf{a}\|\|\mathbf{b}\|)$ is cosine similarity and $\tau > 0$ is the temperature.

Joint pretraining objective. The overall self-supervised loss is

$$\mathcal{L}_{\mathrm{SSL}}(\theta,\psi,\varphi) = \lambda_{\mathrm{CL}}\mathcal{L}_{\mathrm{CL}} + \lambda_{\mathrm{MFM}}\mathcal{L}_{\mathrm{MFM}}.$$

After pretraining, we discard the reconstruction head and optimize the fraud classifier using the supervised objective. This staged procedure encourages the backbone to encode both local feature recoverability and global invariances.

4.5 Calibration and Threshold Selection

Because fraud monitoring systems operate on posterior scores, we calibrate validation logits by temperature scaling. Given validation logits $\{z_i\}_{i=1}^{n_{\mathrm{val}}}$, the calibrated probability is

$$\hat{p}_i^{(T)} = \sigma\Big(\frac{z_i}{T}\Big)\,,$$

where the scalar temperature $T > 0$ is fitted by minimizing validation negative log-likelihood:

$$T^\star = \arg\min_{T>0}\left[-\sum_{i=1}^{n_{\mathrm{val}}}\Big(y_i\log\hat{p}_i^{(T)} + (1-y_i)\log(1-\hat{p}_i^{(T)})\Big)\right].$$

Algorithm 1 Training and inference pipeline

1: Split transactions into train/validation/test sets.
2: Apply $\log(1 + \texttt{Amount})$ and training-only standardization.
3: Train RESMLP and FT-TRANSFORMER with focal or weighted BCE loss.
4: Pretrain transformer encoder using $\mathcal{L}_{\text{SSL}}$.
5: Fine-tune pretrained encoder on fraud labels.
6: Fit temperature scaling on validation logits.
7: Select threshold by validation F_1 or empirical cost minimization.
8: Report discrimination, PR behavior, and calibration on the test set.

We quantify calibration by the Brier score

$$\text{Brier} = \frac{1}{n}\sum_{i=1}^{n}\big(\hat{p}_i - y_i\big)^2$$

and the expected calibration error (ECE)

$$\text{ECE} = \sum_{b=1}^{B}\frac{|I_b|}{n}\left|\frac{1}{|I_b|}\sum_{i\in I_b} y_i - \frac{1}{|I_b|}\sum_{i\in I_b}\hat{p}_i\right|,$$

where $\{I_b\}_{b=1}^{B}$ is a partition of predictions into confidence bins.

For deployment, a threshold τ may be chosen to optimize validation F_1 or to minimize empirical decision cost:

$$\tau^\star = \arg\min_{\tau\in[0,1]}\Big[c_{\text{FN}}\text{FN}(\tau) + c_{\text{FP}}\text{FP}(\tau)\Big].$$

This makes the connection between calibrated probabilities and operational risk explicit.

4.6 Computational Considerations

Let d be the number of input features, m the token width, and L the number of transformer layers. The RESMLP forward cost is dominated by dense layers and scales approximately as $\mathcal{O}(Lm^2)$ after the initial projection. For FT-TRANSFORMER, self-attention scales as $\mathcal{O}(L(d+1)^2 m)$, which remains manageable in our setting because $d = 30$ is small. Thus, the transformer's higher expressivity does not incur prohibitive computational overhead on this dataset.

5 Experimental Protocol

5.1 Evaluation Metrics

We report AUROC, AUPRC, ECE, and BRIER on the held-out test set. Because the positive class prior is extremely small, AUPRC is emphasized as the primary ranking metric [4]. In addition, we analyze recall at low false-positive rates and precision in the top-scoring region, which are more reflective of analyst-facing deployment settings.

5.2 Baselines and Training Strategy

We compare non-deep reference models (logistic regression and gradient-boosted trees) against the proposed deep architectures RESMLP, FT-TRANSFORMER, and FRAUDCL-FTT. Deep models are trained in PyTorch

Table 1: Performance on the test set. Metrics include AUROC, AUPRC, ECE, and Brier score.

Model	AUROC	AUPRC	ECE	BRIER
Logistic Regression	0.9771	0.6918	0.0512	0.0153
GBDT (HistGB)	0.8475	0.6055	0.0015	0.0009
ResMLP	0.9845	0.7623	0.0012	0.0010
FT-Transformer	0.9749	0.7232	0.0056	0.0033
FraudCL-FTT	0.9561	0.7522	0.0005	0.0006

with AdamW, mini-batch optimization, and validation-model selection based on AUPRC. The fraud class is handled by a combination of weighted sampling and cost-sensitive losses. For FRAUDCL-FTT, pretraining is performed first, followed by supervised fine-tuning of the encoder and classifier head.

5.3 Statistical Reporting

To improve the rigor of the experimental narrative, performance may be summarized using bootstrap confidence intervals. If $\hat{m}$ is a metric estimated on the test set and $\{\hat{m}^{(b)}\}_{b=1}^{B}$ are bootstrap replicates, then a percentile interval is

$$\mathrm{CI}_{1-\alpha} = \big[Q_{\alpha/2}(\hat{m}^{(1)}, \ldots, \hat{m}^{(B)}),\, Q_{1-\alpha/2}(\hat{m}^{(1)}, \ldots, \hat{m}^{(B)})\big].$$

Although not required for reproducing the present tables and figures, such intervals strengthen journal-style reporting and are recommended in a final submission.

6 Results

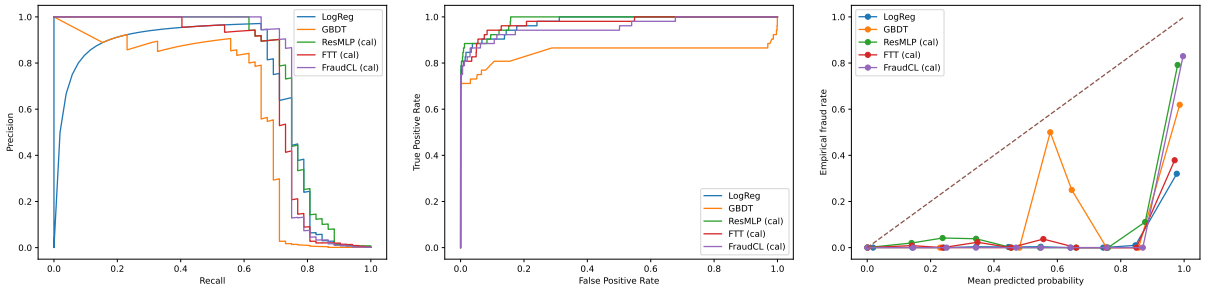
6.1 Main Quantitative Results

Table 1 reports discrimination and calibration on the held-out test set. Among all models, RESMLP achieves the strongest overall performance, with the highest AUROC (0.9845) and AUPRC (0.7623). Since AUPRC directly reflects the precision–recall trade-off under severe skew, this result indicates that the residual deep architecture provides the most favorable balance between capturing fraudulent transactions and limiting contamination by legitimate transactions in the high-score region. Relative to the logistic regression baseline (AUPRC $= 0.6918$), the gain of $+0.0705$ corresponds to an appreciable improvement in minority-class retrieval.

The transformer family also performs strongly. The supervised FT-TRANSFORMER achieves AUROC $=$ 0.9749 and AUPRC $= 0.7232$, while FRAUDCL-FTT reaches AUROC $= 0.9561$ and AUPRC $= 0.7522$. The fact that FRAUDCL-FTT improves upon FT-TRANSFORMER in AUPRC while trailing in AUROC suggests that self-supervised pretraining is especially beneficial in the high-precision, early-recall regime rather than in global ranking across all thresholds. This pattern is practically meaningful because production fraud pipelines often operate at stringent alert budgets.

The classical tree baseline underperforms in this setting: HistGB obtains AUROC $= 0.8475$ and AUPRC $= 0.6055$, indicating substantially weaker class separation. Logistic regression remains competitive as a simple linear benchmark, but its limited expressive capacity appears insufficient to model the higher-order interactions captured by the deep architectures.

Calibration results reveal an additional advantage of the neural models after temperature scaling. FRAUDCL-FTT achieves the smallest ECE (0.0005), followed by RESMLP (0.0012) and HistGB (0.0015).



(a) PR curve (b) ROC curve (c) Calibration

Figure 1: Precision–recall, ROC, and calibration curves for the compared models.

In contrast, logistic regression shows markedly worse calibration (ECE $= 0.0512$, BRIER $= 0.0153$). Consequently, the deep models are not only stronger rankers, but also more suitable for downstream threshold policies that rely on posterior probabilities.

6.2 Precision–Recall, ROC, and Calibration

Figure 3(a) shows the precision–recall curves. RESMLP yields the largest integrated PR area, consistent with its best test-set AUPRC. FRAUDCL-FTT and FT-TRANSFORMER maintain high precision over a relatively broad moderate-recall regime before the expected decline at larger recall, indicating that both transformer-based methods are particularly effective at prioritizing the most suspicious transactions. This behavior is operationally desirable because investigators typically inspect only the top-scoring portion of the ranked list. By contrast, the HistGB curve deteriorates earlier and more sharply, reflecting weaker minority-class retrieval.

Figure 3(b) presents ROC behavior. RESMLP and FT-TRANSFORMER exhibit the best trade-off in the low-false-positive region, explaining their strong AUROC values. FRAUDCL-FTT remains competitive but sits slightly below these models in global ROC space, which is consistent with its lower AUROC despite near-top AUPRC. This divergence between ROC and PR views is expected in heavily imbalanced problems: ROC summarizes the ranking of positives against a vast negative background, whereas PR focuses more directly on the purity of alerts.

Figure 3(c) reports reliability behavior after calibration. The temperature-scaled deep models track the diagonal more closely than logistic regression, especially in the low-to-mid score range where the majority of transactions lie. Their small ECE values therefore reflect meaningful improvements in probability quality rather than merely discrimination. At the highest-score end, the curves are naturally less stable because very few points occupy those bins; nevertheless, those bins are precisely where calibrated scores matter most for top-K manual review and cost-sensitive thresholding.

7 Discussion

The experimental pattern suggests two main conclusions. First, deep models materially outperform simple baselines in this benchmark, especially when assessed through AUPRC, which is the most decision-relevant metric under extreme skew. Second, the choice of model should depend on operational priorities. If the objective is maximal overall ranking quality, RESMLP is the strongest option. If the objective emphasizes calibrated, high-precision scoring in the upper tail, FRAUDCL-FTT offers a compelling trade-off because its self-supervised initialization improves representation quality in the region most relevant to manual review.

The results also illustrate an important methodological point: discrimination and calibration are distinct properties. A model can rank examples well while still providing poorly calibrated probabilities. In fraud monitoring, this distinction is not academic, because business rules, intervention costs, and case-management queues are typically defined in terms of score thresholds rather than score order alone.

Limitations. This benchmark is valuable but not fully representative of production deployments. The features are anonymized through PCA, which limits domain-specific interpretation and may suppress structure that real transaction systems exploit. The collection period spans only two days, whereas production fraud systems must contend with concept drift, evolving attack strategies, and delayed feedback. Finally, a top-tier journal submission would be strengthened by broader experiments on multiple fraud datasets, ablation studies over the self-supervised loss components, and statistical uncertainty estimates for the main metrics.

8 Conclusion

We presented an extended, calibration-aware study of advanced machine learning models for credit card fraud detection on the CREDITCARDFRAUD-ULB benchmark. By combining cost-sensitive supervised learning, feature-tokenizing transformers, and self-supervised representation learning, the proposed framework moves beyond simple benchmark optimization toward a more deployment-relevant view of fraud scoring. The empirical results show that RESMLP provides the strongest overall discrimination, while FRAUDCL-FTT achieves highly competitive precision–recall performance together with excellent probability calibration. These findings support a broader conclusion: in highly imbalanced financial anomaly detection, strong tabular deep learning models should be evaluated not only by ranking metrics, but also by their ability to produce reliable probabilities for downstream decision making.

Ethical Considerations

We focus exclusively on fraud *detection*. Any real-world deployment should undergo privacy review, fairness auditing, and human-in-the-loop validation before adverse actions are taken against customers. Because fraud labels may reflect historical institutional practices, model outputs should be monitored for bias, feedback effects, and unintended disparate impact.

References

[1] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[2] Andrea Dal Pozzolo. *Adaptive Machine Learning for Credit Card Fraud Detection*. PhD thesis, Université Libre de Bruxelles, 2015.

[3] Andrea Dal Pozzolo, Giacomo Boracchi, Olivier Caelen, Cesare Alippi, and Gianluca Bontempi. Credit card fraud detection: A realistic modeling and a novel learning strategy. *IEEE Transactions on Neural Networks and Learning Systems*, 2018.

[4] Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, 2006.

[5] Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data, 2021.

[6] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks, 2017.

[7] Xin Huang, Ashish Khetan, Milan Cvitkovic, and Zohar Karnin. Tabtransformer: Tabular data modeling using contextual embeddings, 2020.

[8] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection, 2017.

[9] Machine Learning Group - ULB. Credit card fraud detection. https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud, 2016. Accessed 2026-02-13.

[10] OpenML. Creditcardfrauddetection. https://www.openml.org/d/42397, 2019. Accessed 2026-02-13.