International Journal of Engineering Inventions
e-ISSN: 2278-7461, p-ISSN: 2319-6491
Volume 15, Issue 11 [November 2026] PP: 95-101

Commitment-Aware Liquidity-Shortfall Prediction Using Traditional Machine Learning

Sahana A M[1], Nithin S[2], Shivaram TS[2], Soundarya G M[2], Radha[2]
[1] *Assistant Professor, Department of Computer Science and Engineering, APS College of Engineering, Bengaluru, India*
[2] *Department of Computer Science and Engineering, APS College of Engineering, Bengaluru, India*
Corresponding Author: Shivaram TS.

ABSTRACT: *This study evaluates whether explicitly representing recurring financial commitments adds predictive information for next-calendar-month negative net cash flow. The experiment uses a synthetic transaction benchmark containing 11,365 transactions from 249 synthetic users and a derived panel of 3,444 monthly snapshots from 241 eligible users. Two Hist Gradient Boosting Classifier models were compared: Model A uses 27 class-blind features describing income, aggregate outflows, net-cash-flow history, transaction reliability, and forecast proxies; Model B retains those features and adds 14 commitment-specific features derived from pre-labeled Subscription, EMI/Loan, and BNPL transaction classes. On the 422-observation chronological test set, Average Precision was 0.9564 for Model A and 0.9617 for Model B (difference +0.0053); ROC-AUC was 0.9446 and 0.9561, respectively (difference +0.0115). Balanced accuracy increased from 89.20% to 90.07%, and the Brier score decreased from 0.0834 to 0.0745. However, the reported paired user-clustered bootstrap 95% confidence intervals included zero (Average Precision: [−0.0038, +0.0151]; ROC-AUC: [−0.0051, +0.0272]; balanced accuracy: [−1.22, +2.98] percentage points). The point estimates favor Model B on some metrics, but the experiment does not establish a reliable incremental benefit. The target is negative monthly net cash flow, not an observed overdraft, loan default, or insolvency. The results are conditional on synthetic data and pre-labeled commitment classes; validation using appropriately governed real-world transaction data and an independently evaluated commitment detector remains future work.*

Keywords: *liquidity-shortfall prediction, recurring financial commitments, financial transactions, gradient boosting, temporal validation, synthetic data*

NOMENCLATURE

Symbol	Description	Unit / range
I_t	Successful income observed in month t	Currency units
C_t	Successful outflows assigned to commitment classes	Currency units
V_t	Other successful non-income outflows	Currency units
CF_t	Monthly net cash flow: I_t − C_t − V_t	Currency units
Y_{t+1}	Binary indicator for negative net cash flow in the next calendar month	0 or 1
p_i	Predicted probability of Y=1 for test observation i	[0, 1]

Date of Submission: 02-11-2026 Date of acceptance: 12-11-2026

I. INTRODUCTION

Personal financial management systems typically categorize past transactions and summarize spending or budgets. Short-horizon cash-flow prediction also depends on identifying recurring obligations—such as digital subscriptions, equated monthly installments (EMIs), and buy-now-pay-later (BNPL) repayments—that are embedded in transaction histories. Transaction classification alone does not answer whether explicitly representing those obligations improves prediction beyond aggregate income and spending history. This study evaluates that incremental contribution for next-calendar-month negative net cash flow.

Prior studies demonstrate the feasibility of transaction categorization, recurring-pattern extraction, cash-flow forecasting and transaction-based credit-distress prediction [1]–[7]. These studies address related but different prediction targets and datasets. The present study focuses on an incremental-feature question: when a conventional model already uses aggregate income, expenses, net cash flow and payment reliability, does adding explicit commitment-specific information improve prediction of a negative net-cash-flow outcome in the following calendar month?

Research question

Does adding explicitly modeled recurring-financial-commitment features improve a traditional machine-learning model's prediction of next-calendar-month negative net cash flow compared with a baseline using class-blind aggregate financial features?

Contributions

- A controlled comparison of two HistGradientBoostingClassifier models using the same chronological partitions, with 27 class-blind baseline features and 14 additional commitment-specific features.
- An evaluation of ranking, threshold-based classification and probabilistic error, with paired user-clustered bootstrap confidence intervals rather than unsupported p-values.
- An implementation audit that aligns the feature lookback window and baseline definition with the research question, and an explicit account of what the synthetic benchmark can and cannot establish.

II. RELATED WORK

A. Transaction categorization and personal-finance analytics

García-Méndez et al. developed a two-stage banking-description classifier that combines a short-text similarity filter with a support vector machine, addressing the brief, noisy, and domain-specific nature of merchant descriptions [4]. Lesner et al. described large-scale personalized transaction categorization for business accounting, using information shared across users to classify transactions even when counterparties are new [5]. These studies support transaction categorization as an established component, but neither evaluates whether commitment-specific features improve prediction of next-month negative net cash flow.

Kotios et al. combined rule-based transaction categorization and machine learning with a cash-flow forecasting component for small and medium-sized enterprises [3]. Yang et al. proposed a deep-learning system jointly addressing personal transaction classification and budget prediction [7]. The former concerns business accounts and the latter uses a deep-learning architecture; both differ from the present study in target formulation and experimental focus. Their relevance is that prediction can be coupled to upstream transaction understanding, while the additional value of explicit commitment features remains a separate empirical question.

B. Recurring financial patterns

Ibrain, Hernández, and Peinado proposed unsupervised filtering approaches for discrete financial-event series to remove noise and isolate recurring patterns before forecasting the next transaction date or amount [2]. Lorpunmanee and Kamonsantiroj introduced CP-growth to mine partial-periodic recurring patterns across inter-transaction time series [6]. The latter is a general pattern-mining method rather than a direct bank-payment classifier. These studies motivate temporal recurrence analysis, but neither evaluates a baseline against commitment-specific features for negative monthly net cash flow.

C. Transaction-based financial distress prediction

Monteux et al. studied transaction-only machine learning for short-term credit distress and overdraft-related outcomes under a structural shock using a large European bank dataset [1]. That work is directly relevant to the use of transactions as predictors of future financial distress, but its target is overdraft dynamics rather than the synthetic next-month net-cash-flow label used here. It also does not evaluate the incremental contribution of the 14 commitment-specific features examined in this study.

D. Positioning of the present study

Within the seven supplied papers, we did not identify a directly comparable experiment isolating the incremental value of commitment-specific features for this next-calendar-month negative-net-cash-flow target. This finding is limited to the reviewed corpus and is not a claim that no prior system has studied the problem. The contribution is an empirical feature-ablation study, not a new machine-learning algorithm or a claim of universal novelty.

III. PROBLEM FORMULATION AND DATASET

A. Target definition

Let I_t denote successful income transactions in calendar month t, C_t successful outflows assigned to the dataset classes Subscription, EMI/Loan or BNPL, and V_t all other successful non-income outflows. Monthly net cash flow is defined as:

$$CF_t = I_t - (C_t + V_t)$$

For a feature month t, the binary target is the observed net cash flow in the next calendar month t+1:

$$Y_{t+1} = 1 \text{ if } CF_{t+1} < 0; \text{ otherwise } Y_{t+1} = 0.$$

This operational target measures negative monthly net cash flow. It is not an account-balance threshold, an observed overdraft, or proof that a user cannot meet payments. The source dataset contains no true running-balance or overdraft-limit field; therefore, "liquidity shortfall" refers here to the stated negative-net-cash-flow proxy.

B. Dataset and panel construction

The supplied benchmark contains 11,365 synthetic transactions from 249 synthetic users dated 30 December 2023 through 31 December 2025. The transaction table has 18 columns, no nulls in the audit profile, and no exact duplicate rows. Class counts are Subscription 4,768, EMI 3,716, BNPL 1,281 and Normal 1,600. After monthly aggregation and requiring six prior calendar months plus a next-month target, the modeling panel contains 3,444 snapshots from 241 eligible users, with feature months from May 2024 through November 2025. All modeling findings are conditional on this simulated data-generation process.

Monthly rows without transactions are reindexed to zero in the pipeline. Consequently, a target month with no transaction rows becomes CF=0 and is labelled non-shortfall. This occurred in 224 of the 3,444 snapshots overall and in 30 of the 422 test snapshots. The analysis preserves the supplied simulation convention, but its interpretation depends on treating absent monthly transaction rows as complete zero activity rather than missing statement data.

C. Chronological data partitions

Partition (feature month)	Period	Snapshots	Users	Positive labels	Prevalence
Training	May 2024–May 2025	2,463	238	1,569	63.70%
Validation	June–August 2025	559	197	298	53.31%
Test	September–November 2025	422	161	200	47.39%
Full panel	May 2024–November 2025	3,444	241	2,067	60.02%

Table 1. Chronological modeling partitions. User counts overlap across time partitions; the test evaluates future periods for known users.

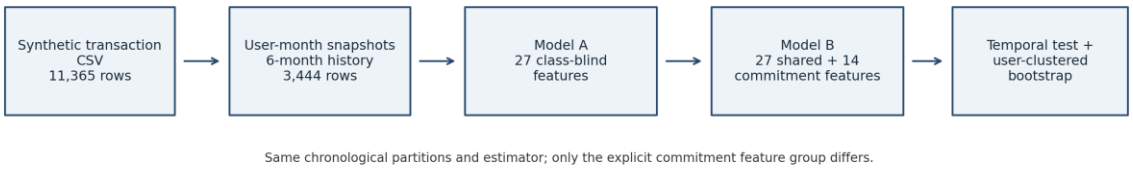


Fig. 1. Audited experimental workflow. Only the commitment-specific feature group differs between models.

IV. METHODS

A. Feature-set comparison

Model A is a 27-feature conventional baseline. Its features are computed without grouping transactions by commitment class or merchant category: six income-history features; seven aggregate-outflow and generic transaction features; seven aggregate net-cash-flow and buffer features; four recent transaction-status reliability features; and three baseline forecast proxies. The baseline includes historical total expenditure and net cash flow, so it does not discard the effect of recurring outflows from aggregate cash-flow history; rather, it does not explicitly disaggregate their amounts and types.

Model B retains the 27 baseline features and adds 14 commitment-specific features computed from pre-labeled transaction_class values over the prior six months. These cover previous, rolling and expected amounts; commitment-to-income and class-specific/total burden; active months, growth and class diversity; and two compound forecast proxies. No explicit next-due-date feature is included, and upstream detector accuracy is not evaluated.

Feature group	Model A	Model B	Description
Income history	6	6	Rolling means, slope, volatility and stability
Aggregate expenses / generic transaction statistics	7	7	Total outflows; no commitment-category grouping in shared baseline features
Net cash flow and buffers	7	7	Historical aggregate net cash flow
Transaction reliability	4	4	Recent success, failure, pending and resolved-payment rates
Baseline forecast proxies	3	3	Income, total expenses and net cash flow

Feature group	Model A	Model B	Description
Commitment levels and burden	0	9	Historical commitment amounts and class-specific burdens
Commitment dynamics and diversity	0	3	Active months, recent growth, class diversity
Commitment-aware compound forecasts	0	2	Total-expense and net-cash forecast proxies
Total	27	41	Model B adds 14 features

Table 2. Feature architecture after the audit of the original feature construction.

B. Feature calculations and leakage controls

The monthly transaction history was restricted to information available at the feature month t. Historical income and expenditure summaries use the preceding six calendar months; three-month aggregates use the latest three months. Commitment burdens are average monthly successful positive outflows within the same six-month lookback. In the audited run, baseline features were rebuilt from total outflows and generic transaction statistics so they do not depend on transaction_class-based commitment grouping. Model B uses the dataset's pre-labeled transaction_class values only for the added commitment features. The experiment therefore tests the value of commitment-aware feature enrichment conditional on those labels; it does not test whether the system can infer the labels reliably from raw bank descriptions.

The target is calculated from transactions in month t+1, while feature construction is restricted to months at or before t. The train/validation/test split is made by feature month, so outcome months do not overlap between partitions: training feature months end in May 2025, validation uses June–August 2025, and testing uses September–November 2025. This is a temporal forecast for known users, not a test on unseen users. Median imputation and scaling are fitted only on the training partition. The estimator is unweighted; the original script's sample-weight expression evaluated to one for both classes because the positive class was the training majority.

C. Classifier and hyperparameters

Both feature sets use scikit-learn HistGradientBoostingClassifier [8], [9] with learning_rate=0.06, max_iter=180, max_leaf_nodes=15, min_samples_leaf=25, l2_regularization=1.0 and random_state=42. The probability threshold for confusion-matrix metrics is 0.50. The model is evaluated with identical train/validation/test snapshots, and both model variants receive the same training protocol.

D. Evaluation metrics and uncertainty

Average Precision (AP) is the primary ranking metric because it emphasizes precision–recall performance at the observed class prevalence [10]. Secondary measures are ROC-AUC, accuracy, balanced accuracy, precision, recall, F1-score, Brier score and expected calibration error (ECE). The Brier score is the mean squared error between predicted probabilities and binary outcomes [11].

Paired 95% confidence intervals were computed from 1,000 user-clustered bootstrap resamples. At each iteration, test users are sampled with replacement and all their observed test-month rows are retained together, preserving within-user dependence. Confidence intervals are percentile intervals for the Model B minus Model A difference [12]. The bootstrap fraction of resamples favouring a model is descriptive and is not treated as a conventional p-value. These intervals condition on the fitted models and saved test predictions; they do not quantify model-fitting or training-sample variability. No p-values are reported in this manuscript.

V. RESULTS

A. Test-set comparison

Table 3 reports the controlled comparison using the audited feature construction. Model B's Average Precision is 0.0053 higher than Model A's (0.9617 versus 0.9564), but the paired user-cluster bootstrap 95% confidence interval includes zero [−0.0038, +0.0151]. ROC-AUC is 0.0115 higher (0.9561 versus 0.9446), with an interval that also includes zero [−0.0051, +0.0272]. At the fixed 0.50 threshold, balanced accuracy is 90.07% for Model B and 89.20% for Model A (difference +0.87 percentage points; 95% CI [−1.22, +2.98] percentage points). Precision is 92.97% versus 90.58%, while recall is 86.00% versus 86.50%. Model B's Brier score is lower by 0.0089 (95% CI [−0.0190, +0.0012]), and its empirical expected calibration error (ECE) is 0.0284 versus 0.0647. The point estimates are mixed in their practical implications, and the reported intervals do not establish a reliable incremental benefit.

Metric	Model A	Model B	Δ (B−A)	95% user-cluster bootstrap CI for Δ
Average Precision (primary)	0.9564	0.9617	+0.0053	[−0.0038, +0.0151]
ROC-AUC	0.9446	0.9561	+0.0115	[−0.0051, +0.0272]
Accuracy	0.8934	0.9028	+0.0095	—
Balanced accuracy	0.8920	0.9007	+0.0087	[−0.0122, +0.0298]
Precision	0.9058	0.9297	+0.0239	—
Recall	0.8650	0.8600	−0.0050	—
F1-score	0.8849	0.8935	+0.0086	[−0.0141, +0.0307]
Brier score (lower is better)	0.0834	0.0745	−0.0089	[−0.0190, +0.0012]
ECE (lower is better)	0.0647	0.0284	−0.0363	—

Table 3. Test metrics from the audited feature construction (N = 422; 200 positive and 222 negative outcomes).

Metric differences are calculated from the unrounded estimates. The accuracy difference (+0.0095) is based on unrounded rates and therefore differs by 0.0001 from subtraction of the displayed rounded accuracies. The paired user-cluster bootstrap intervals were computed from unrounded test probabilities using 1,000 resamples [12]; the intervals reported for Average Precision, ROC-AUC, balanced accuracy, F1-score, and Brier score all include zero. The proportions of bootstrap resamples favoring Model B (88.8% for Average Precision, 93.4% for ROC-AUC, 78.5% for balanced accuracy, 76.6% for F1-score, and 95.8% for Brier-score improvement) are descriptive proportions, not p-values, and do not establish statistical significance.

B. Threshold-based errors

Model	TN	FP	FN	TP
Model A: 27 features	204	18	27	173
Model B: 41 features	209	13	28	172

Table 4. Confusion matrices at the fixed probability threshold of 0.50.

At the fixed decision threshold of 0.50, Model B produces five fewer false positives (13 versus 18) and has higher precision (92.97% versus 90.58%). It also produces one additional false negative (28 versus 27), with recall decreasing slightly from 86.50% to 86.00%. Specificity increases from 91.89% to 94.14%. These threshold-specific differences describe a modestly more conservative positive-classification profile; they do not by themselves establish that Model B is superior.

C. Exploratory feature ablation

Variant	Features	AP	ROC-AUC	Balanced acc.	Brier
A: baseline	27	0.9564	0.9446	0.8920	0.0834
A + commitment projections	30	0.9598	0.9516	0.8937	0.0770
A + burdens and diversity	32	0.9545	0.9430	0.8882	0.0856
A + commitment history/ratios	33	0.9604	0.9572	0.9102	0.0716
B: full 41 features	41	0.9617	0.9561	0.9007	0.0745

Table 5. Predefined feature ablation on the same test partition. These variant comparisons are exploratory; confidence intervals were not calculated for every pairwise ablation.

Table 5 presents exploratory ablation results on the same test partition. The 33-feature variant that adds commitment history and burden ratios has the highest balanced accuracy (91.02%) and lowest Brier score (0.0716), while the full 41-feature model has the highest point estimate of Average Precision (0.9617). Permutation-importance outputs identify commitment_ratio (+0.0744 ± 0.0106) and previous_committed (+0.0230 ± 0.0051) as the largest commitment-feature importance estimates, followed by income_volatility (+0.0064) and previous_net_cash (+0.0042). These estimates should be treated as exploratory: the variants were compared on the test partition, not on a separate confirmation set, and correlated features can share or redistribute measured importance.

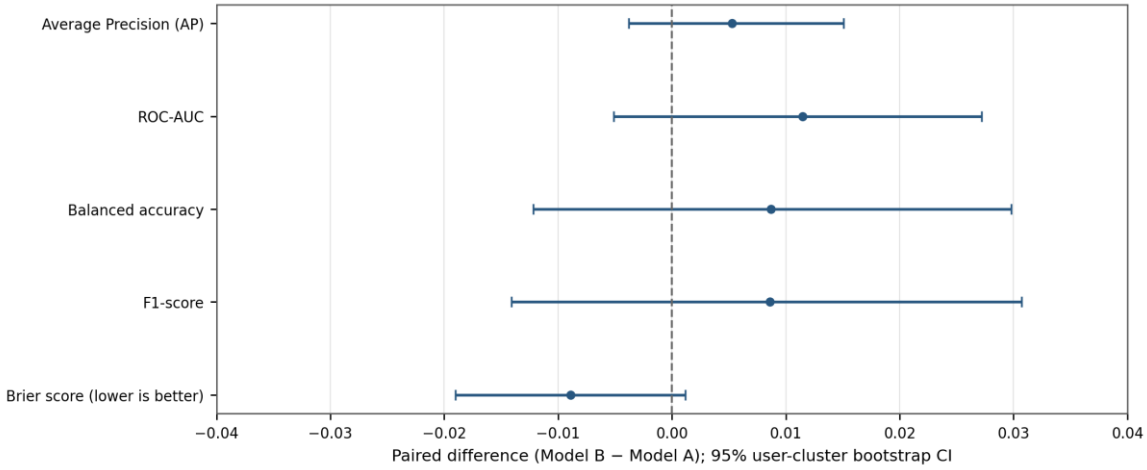


Fig. 2. Paired user-cluster bootstrap confidence intervals for selected metric differences. Each interval crosses zero.

VI. DISCUSSION

On the held-out test period, Model B has higher point estimates for Average Precision (+0.0053), ROC-AUC (+0.0115), balanced accuracy (+0.87 percentage points), and F1-score (+0.0086), as well as a lower Brier score (difference −0.0089). The paired user-clustered 95% confidence intervals reported for Average Precision, ROC-AUC, balanced accuracy, F1-score, and Brier score all include zero. At the fixed 0.50 threshold, Model B yields five fewer false positives but one additional false negative. The evidence is therefore compatible with a modest gain, no meaningful difference, or metric-specific trade-offs; it does not justify a claim of predictive superiority.

The experiment also demonstrates why the baseline definition matters. The initial runner contained transaction-level rates and class-specific burden aggregates that drew on broader historical data while dividing some amounts by only a recent six-month denominator. Some initial baseline features also used commitment-class information in "variable" spending and category-dispersion calculations. For this comparison, the feature windows were aligned and the shared 27 baseline features were rebuilt from aggregate outflows and generic transaction statistics without grouping by commitment class or merchant category. The results reported here are from the audited comparison and supersede the preliminary table in the original experiment package.

The revised baseline still includes historical aggregate net cash flow, which naturally reflects all types of outflows. Model B adds an explicit decomposition of recent outflows into labeled commitment groups and associated burden and dynamics features. This conditional feature-enrichment effect is the specific contribution evaluated here.

VII. LIMITATIONS AND THREATS TO VALIDITY

- Synthetic benchmark and proxy target: all 11,365 transactions are simulated, and the outcome is next-month negative net cash flow—not an observed overdraft, account-balance breach, insolvency, or inability to pay. Real-world predictive performance is untested.
- Pre-labeled commitments: Model B uses the dataset's transaction_class labels to construct commitment features. The upstream detector is not evaluated, and the model does not include explicit next-payment-date or days-until-payment features.
- Missing months: user-months with no transaction rows are reindexed to zero. This occurred in 30 test target months and assumes complete transaction coverage rather than missing statements; real deployments would need safeguards for incomplete data.
- Known-user temporal test: users overlap across training and test periods. Results describe forecasting future periods for users with prior history, not generalization to unseen users. The test set contains 422 snapshots from 161 users over three feature months.
- Threshold and uncertainty: confusion-matrix measures use a fixed 0.50 threshold. Cost-sensitive thresholding and post-hoc calibration were not evaluated. Bootstrap intervals condition on the fitted models and saved predictions and do not include model-fitting variability; no confidence interval is reported for ECE.
- Environment: the audited rerun was verified under Python 3.12, scikit-learn 1.9.0, pandas 3.0.5, NumPy 2.5.2, and SciPy 1.18.0. Reported metrics are tied to the included pipeline script and pinned environment.
- Initial lookback coverage: the transaction series begins on 30 December 2023, so the earliest six-month feature windows may contain less than six months of observed history. Sensitivity to excluding these earliest snapshots was not reported.

VIII. CONCLUSION AND FUTURE WORK

On this synthetic transaction benchmark, the commitment-aware model had modestly higher point estimates for Average Precision and ROC-AUC, improved balanced accuracy and F1-score at the reported threshold, and a lower Brier score. Its precision was higher and it produced five fewer false positives, but recall was slightly lower and it produced one additional false negative. The paired user-clustered bootstrap confidence intervals reported for Average Precision, ROC-AUC, balanced accuracy, F1-score, and Brier score included zero. The study therefore does not establish a reliable incremental benefit from the added commitment features. Future work should validate the approach on complete, de-identified real-world statements, evaluate the upstream commitment detector against independently verified labels, check incomplete-statement safeguards and unseen-user generalization, and test payment-timing features and decision thresholds under a pre-specified protocol.

REFERENCES

[1] M. Monteux, M. C. Arcuri, G. Gandolfi, and M. Riani, "Transaction-based machine learning under restricted feature availability for short-term credit distress under a structural shock," Finance Research Letters, vol. 111, Art. 110642, 2026, doi: 10.1016/j.frl.2026.110642.

[2] A. Ibrain, V. Hernández, and L. Peinado, "Unveiling recurring financial patterns: Novel unsupervised filtering algorithms for enhanced forecasting," in Proc. 5th ACM International Conference on AI in Finance (ICAIF '24), 2024, pp. 406–418, doi: 10.1145/3677052.3698596.

[3] D. Kotios, G. Makridis, G. Fatouros, and D. Kyriazis, "Deep learning enhancing banking services: A hybrid transaction classification and cash flow prediction approach," Journal of Big Data, vol. 9, Art. 100, 2022, doi: 10.1186/s40537-022-00651-x.

[4] S. García-Méndez, M. Fernández-Gavilanes, J. Juncal-Martínez, F. J. González-Castaño, and Ó. Barba Seara, "Identifying banking transaction descriptions via support vector machine short-text classification based on a specialized labelled corpus," IEEE Access, vol. 8, pp. 61642–61655, 2020, doi: 10.1109/ACCESS.2020.2983584.

[5] C. Lesner, A. Ran, M. Rukonic, and W. Wang, "Large scale personalized categorization of financial transactions," in Proceedings of the Thirty-First AAAI Conference on Innovative Applications of Artificial Intelligence (IAAI-19), vol. 33, no. 1, pp. 9365–9372, 2019, doi: 10.1609/aaai.v33i01.33019365.

[6] S. Lorpunmanee and S. Kamonsantiroj, "Efficient mining recurring patterns of inter-transaction in time series," Journal of Advanced Computational Intelligence and Intelligent Informatics, vol. 23, no. 3, pp. 402–413, 2019, doi: 10.20965/jaciii.2019.p0402.

[7] Z. Yang, B. Biibosunov, S. Biibosunova, G. Kalmanbetova, and X. Wang, "A deep learning framework for automated transaction classification and budget prediction in personal financial management," Discover Artificial Intelligence, vol. 6, Art. 885, 2026, doi: 10.1007/s44163-026-02025-1.

[8] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," The Annals of Statistics, vol. 29, no. 5, pp. 1189–1232, 2001, doi: 10.1214/aos/1013203451.

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

[10] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets," PLoS ONE, vol. 10, no. 3, Art. e0118432, 2015, doi: 10.1371/journal.pone.0118432.

[11] G. W. Brier, "Verification of forecasts expressed in terms of probability," Monthly Weather Review, vol. 78, no. 1, pp. 1–3, 1950, doi: 10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.

[12] B. Efron and R. J. Tibshirani, An Introduction to the Bootstrap. Boca Raton, FL, USA: Chapman & Hall/CRC, 1993.