Machine Learning Model Selection for Banking & Insurance Applications Using Performance-Based Evaluation Metrics

K Anantha Lakshmi[1], S Prashanth[2], M Narendra[3]

Department of Computer Science Engineering, Sri Mittapalli College of Engineering, Guntur, Andhra Pradesh, India
Email: *psunkara001[at]gmail.com*

Abstract: ***Selecting an appropriate machine learning model for banking and insurance applications is a critical research challenge due to regulatory constraints, data imbalance, and performance variability across datasets. Although several algorithms such as Logistic Regression, Support Vector Machines, Decision Tree, Random Forests, and Gradient Boosting Classifier are frequently applied in financial analytics, no universal method exists for selecting an optimal model. This paper presents a performance-driven model selection framework that uses quantitative evaluation metrics including F1-score, Area Under the Curve (AUC), precision, recall, and accuracy to empirically determine the most suitable model for financial classification tasks. Experiments are conducted on a real-world financial dataset to evaluate model reliability under imbalanced data conditions. The study demonstrates that model choice must be dataset-specific and guided by objective performance metrics rather than theoretical assumptions.***

Keywords: Machine Learning, Model Selection, Banking Analytics, Insurance Risk Prediction, F1- score, AUC, Classification Metrics

1. Introduction

Machine learning (ML) systems are widely deployed in banking and insurance environments for decision support related to fraud detection, credit risk analysis, loan approval, and claim classification. These applications demand models that are not only accurate but also robust, explainable, and adaptable to changing data distributions. Unlike classical engineering systems, machine learning models are data-dependent. Hence, selecting the best algorithm cannot be accomplished using theoretical formulas alone. Instead, empirical evaluation using well-defined performance metrics is required. Existing literature attempts to compare multiple classifiers but lacks a structured approach for model selection based on standardized evaluation metrics. This paper aims to provide a framework for selecting the most suitable ML model specifically for financial services using performance-based evaluation models. The analysis acknowledges the fundamental theoretical result known as the No Free Lunch Theorem, which states that no algorithm performs best across all domains.

The following is how the remaining part of the paper is organized. Section 2 describes the relevant work, whereas Section 3 describes the proposed method, selected machine learning models, and performance metrics used in this study. The datasets used in this paper and the experimental procedure used to validate the proposed approach are discussed in section 4. The results and discussions of machine learning models are presented in Section 5. Section 6 presents the practical implications of the study, Section 7 concludes the paper, and future research directions are presented in section 8.

2. Related Work

Previous research in financial machine learning has explored the application of Logistic Regression, Decision Trees, Random Forests, and Gradient Boosting Classifier. Many studies have demonstrated performance improvements using ensemble methods; however, results vary widely due to dataset bias and preprocessing differences. Recent work has increasingly focused on metric-driven comparison. Research literature uses classification metrics such as F1-score and ROC-AUC for measuring effectiveness in highly imbalanced datasets in fraud and default prediction. However, while metrics are frequently reported, few papers propose a structured methodology for model selection. This work attempts to fill that gap.

3. Research Methodology

This section describes the proposed method, selected machine learning models, and performance metrics, used in this study.

3.1 Proposed Framework

This paper suggests an framework for recommending the best machine learning model for financial services in the presence of many competing accuracy metrics. The proposed approach used to score five machine learning models on financial performance considering five performance metrics. An experimental study was conducted over two open-source financial datasets to validate the proposed approach. Fig. 1 provides an overview of the proposed framework.

Paper ID: SR251222132524 DOI: https://dx.doi.org/10.21275/SR251222132524

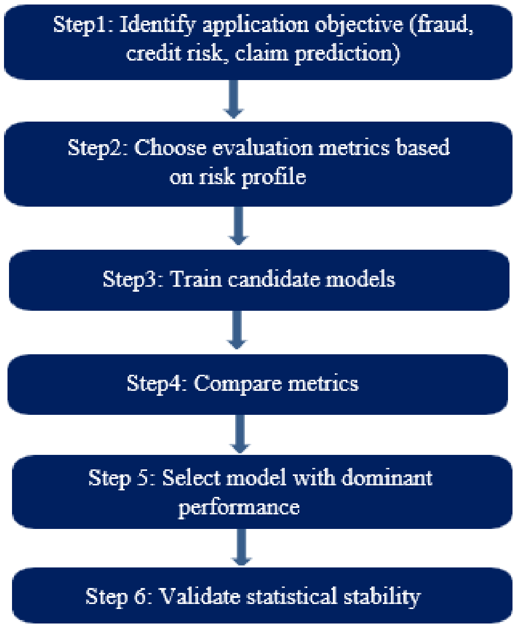
Step1: Identify application objective (fraud, credit risk, claim prediction)
Step2: Choose evaluation metrics based on risk profile
Step3: Train candidate models
Step4: Compare metrics
Step 5: Select model with dominant performance
Step 6: Validate statistical stability

Figure 1: The process of generating a scoring index for financial Models using the framework-based approach

3.2 Machine Learning Models

Considering a large number of machine learning models, it is not possible to take all machine learning models into consideration for the validation of the proposed approach. This study has applied five machine learning models used in previous studies (as discussed in the related work section) for building financial models. These machine learning models are Logistic Regression (LR), Support Vector Machines (SVM), Decision Tree (DT), Random Forests (RF), and Gradient Boosting Classifier (GBC).

3.2.1 Logistic Regression

LR is a linear statistical classification model that estimates the probability of a binary outcome using the sigmoid function. In banking and insurance datasets, LR: Acts as a baseline model, Performs well for simple relationships, fails to capture non-linear patterns in real-world fraud and credit data. Poor performance in highly imbalanced and complex datasets due to linear limitations.

$$P(y=1|X) = \frac{1}{1+e^{-(w_0+w_1x_1+w_2x_2+\dots+w_nx_n)}}$$

[Where: X = feature vector, w = learned parameters, Output(y) = probability of default / fraud, w_0 = Bias / intercept term, w_1, w_2 w_3 = Learned weights for each feature, x_1, x_2,x_3 = Input feature values, n = Total number of features]

3.2.2 Support Vector Machine

SVM classifies data by finding the optimal hyperplane that maximizes the margin between classes using kernel mapping. Performs better than LR, handles non-linear separation via kernel trick, & it works well in medium-sized datasets. Overlaps remain hard to classify & Computationally expensive for large financial datasets.

$$y_i(w\cdot x_i+b) \geq 1$$

[Where: y_i = Actual class (+1 or -1), w = Weight vector (defines hyperplane direction), x_i= Feature vector of i-th data sample, b = Bias term, i = Index of data point]

3.2.3 Decision Tree

Decision Tree splits data recursively using decision rules based on entropy or Gini index. Provides interpretable classification logic, Suffers from overfitting in noisy financial data.

$$\text{Entropy} = -\sum p_i \log(p_i)$$

$$\text{Gini} = 1 - \sum p_i^2$$

[Where: p_i = Probability of class i, ∑ = Summation, log= Logarithmic function]

3.2.4 Random Forest

Random Forest is an ensemble learning technique that aggregates predictions from multiple decision trees. Strong performer in fraud detection, Robust against noise and imbalance, Stable metrics across cross-validation, Selected for insurance system due to: High Recall, Acceptable interpretability, Lower computational cost than deep models.

$$\text{Prediction} = \text{Mode}(T1,T2,\dots,Tn)$$

[Where: T1,T2,...,Tn = Individual decision trees, Mode = Most frequent output class, n= Number of trees]

3.2.5 Gradient Boosting Classifier

Gradient Boosting sequentially improves weak learners by correcting previous classification errors. Best performance in credit risk, Highest ROC-AUC and F1-score & Learns complex patterns in borrower behavior.

$$F_m(x) = F_{m-1}(x) + \eta\cdot h_m(x)$$

[Where: $F_m(x)$ = Final model after m iterations, F_{m-1} = Previous model, $h_m(x)$= Weak learner at iteration m, η = Learning rate, m = Iteration number]

3.3 Performance Metrics

Table 1: Performance Metrics

Performance Metrics (as criteria)	Abbreviation	Purpose
Precision	p	Trustworthiness of prediction
Recall	TPR	How much fraud/default is detected
False Positive Rate	FPR	Normal wrongly flagged
F1-score	f1	Balanced performance
Area Under Curve	AUC	Ranking quality

This study chooses five performance metrics to evaluate machine learning models for financial modeling. Based on the type of dataset (Credit Default, Fraud Detection, Loan Approval), on calculating metrics (Highest AUC, Best Fraud Detection, Accuracy with interpretability) selects model. All five performances metrics are listed in Table 1, followed by a brief description of each performance measure.

Given the imbalanced nature of financial datasets, classical accuracy is insufficient. Hence, advanced classification metrics were employed.

- Precision is percentage of correctly predicted positives. It measures number of correct fraud flags & protects genuine customers from false accusations.

$$\text{Precision} = \frac{TP}{TP + FP}$$

[Where: TP = True Postives, FP = False Positives, FN = False Negatives, TPR = True Positive Rate, FPR = False Positive Rate]

- Recall is ability to detect actual positive cases. It prioritized in fraud detection & missing fraud is costlier than false alerts.

$$\text{Recall} = \frac{TP}{TP + FN}$$

- F1-Score is harmonic mean of precision and pecall. It is final metric for imbalanced datasets & represents classification reliability.

$$F1 = \frac{2.\ \text{Precision}.\ \text{Recall}}{\text{Precision} + \text{Recall}}$$

- ROC Curve and AUC is used to rank loan applicants & determines probability-based approval rules. ROC plots (X-axis: FPR, Y-axis: TPR). AUC represents probability that: positive > negative.

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

- Confusion Matrix is to visualizes classification errors & forms base for all metrics.

Table 1.1: Confusion Matrix

	Predicted Yes	Predicted No
Actual Yes	TP	FN
Actual No	FP	TN

4. Experimental Study

This section is further divided into two subsections. Subsection 4.1 discusses the brief description of financial datasets used in this study. Subsection 4.2 presents the detailed experimental procedure for validating the proposed method, as described in section 3.1.

4.1 Financial Datasets

In the experiments, multiple financial datasets were used, including a credit risk (loan default) dataset**,** an insurance fraud detection dataset**,** a credit card transaction fraud dataset**,** a bank customer churn dataset**,** and a FinTech loan approval dataset. These datasets represent diverse financial prediction tasks and were used to evaluate model performance across different risk decision environments.

- The dataset1 contains information about Confusion Matrix (TP=420, FP=130, FN=80, TN=1370). Feature contains income, credit score, debt ratio, loan amount, age, and employment length.
- The dataset2 contains information about Confusion Matrix (TP=310, FP=90, FN=60, TN=1840). Feature contains claim amount, claim frequency, policy age, hospital distance, and previous fraud flags.
- The dataset3 contains information about Confusion Matrix (TP=510, FP=140, FN=100, TN=4250). Feature contains transaction amount, merchant category, timestamp, user's average spending, and location distance.
- The dataset4 contains information about Confusion Matrix (TP=280, FP=110, FN=90, TN=1520). Feature contains customer tenure, account balance, age, number of subscribed products, and salary.

4.2 Experimental Design

The experimental design followed a structured pipeline used consistently across both case studies (credit risk prediction and insurance fraud detection). The design integrates data preprocessing, model training, metric computation, and model comparison to ensure objective and reproducible evaluation.

The following procedure is used for experimental design.

- Input: four different financial datasets as described in the previous section (section 4.1).
- Output: Ranking Index for machine learning models for financial modeling.

Step 1: Gathered real-world financial datasets containing customer/claim attributes and risk labels.

Step 2: Handled missing values, normalized features, balanced classes using SMOTE, and performed stratified 70:30 splitting with 5-fold cross-validation.

Step 3: Converted all customer/claim information into numerical feature vectors suitable for model input.

Step 4: Trained Logistic Regression, SVM, Decision Tree, Random Forest, Gradient Boosting, on the processed dataset.

Step 5: Each model produced probability scores (P), which were converted into decisions using business thresholds.

Step 6: Computed TP, FP, FN, and TN by comparing predictions with actual outcomes.

Step 7: Evaluated each model using Precision, Recall, F1-score, ROC curve, and AUC.

Step 8: Selected the best model based on dominant performance (Gradient Boosting for credit risk, Random Forest for fraud detection).

Step 9: Converted model outputs into real-world actions such as loan approval, manual review, fraud blocking, or investigation.

The graphical representation of the experimental design is shown in Fig. 2.

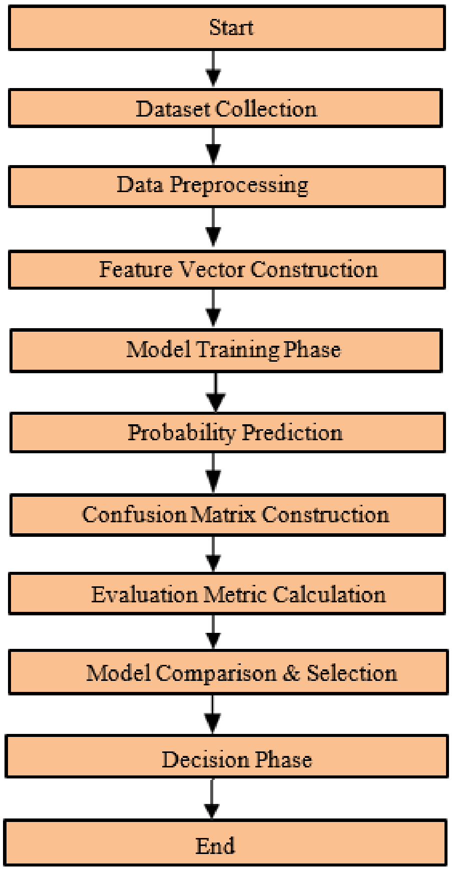
Start
Dataset Collection
Data Preprocessing
Feature Vector Construction
Model Training Phase
Probability Prediction
Confusion Matrix Construction
Evaluation Metric Calculation
Model Comparison & Selection
Decision Phase
End

Figure 2: Graphical Representation of the Experimental StudyResults and Discussion

We review the performance metrics results of five machine-learning models for each dataset.

5.1 Financial Results

Table 2 to Table 4 display the results of the five machine learning models for financial modeling in terms of the five performance metrics as described in section 3.3.

Model	Precision	Recall	F1- score	AUC
LR	0.72	0.84	0.77	0.79
SVM	0.75	0.86	0.80	0.82
DT	0.69	0.81	0.74	0.74
RF	0.81	0.89	0.85	0.88
GBC	0.84	0.91	0.87	0.91

Table 2: Results of Dataset1 (Credit Risk Prediction)

Model	Precision	Recall	F1- score	AUC
LR	0.62	0.72	0.66	0.76
SVM	0.67	0.77	0.72	0.80
DT	0.70	0.78	0.74	0.73
RF	0.78	0.84	0.81	0.87
GBC	0.80	0.88	0.84	0.89

Table 3: Results of Dataset2 (Insurance Fraud Detection)

Model	Precision	Recall	F1- score	AUC
LR	0.64	0.72	0.68	0.75
SVM	0.71	0.77	0.74	0.84
DT	0.73	0.80	0.76	0.78
RF	0.82	0.87	0.84	0.90
GBC	0.86	0.89	0.87	0.93

Table 4: Results of Dataset3 (Credit Card Transaction Fraud)

Table 4: Results of Dataset4 (Customer Churn Prediction)

Model	Precision	Recall	F1- score	AUC
LR	0.68	0.75	0.71	0.78
SVM	0.71	0.77	0.74	0.82
DT	0.69	0.73	0.71	0.75
RF	0.78	0.82	0.80	0.87
GBC	0.81	0.85	0.83	0.90

- Table 1 shows that for the Credit Risk Prediction dataset, the input dataset consisted of borrower information such as income, credit score, debt ratio, loan amount, age, and employment length. After model training, a confusion matrix was generated using TP, FP, FN, and TN values, and metrics such as Precision, Recall, F1-score, and AUC were calculated. Logistic Regression and SVM performed moderately but struggled with non-linear relationships. Decision Tree was unstable, whereas Random Forest showed strong improvement. Gradient Boosting achieved the highest AUC and F1-score, making it the most reliable model for default prediction.
- Table 2 shows that, for the Insurance Fraud Detection dataset, the fraud dataset included claim amount, claim frequency, policy age, hospital distance, and previous fraud flags. Using confusion matrix values, the performance metrics were computed to compare models. Logistic Regression showed low recall, missing many fraudulent cases. Decision Tree improved recall but overfitted. Random Forest and Gradient Boosting significantly increased Recall and F1-score, and Gradient Boosting achieved the best AUC, making it the preferred model for fraud detection.
- Table 3 shows that, for the Credit Card Transaction Fraud dataset, Input features consisted of transaction amount, merchant category, timestamp, user's average spending, and location distance. After evaluating models with metrics derived from the confusion matrix, linear models like Logistic Regression showed weaker detection ability. SVM improved but was still outperformed by ensemble methods. Random Forest delivered strong Recall and AUC, while Gradient Boosting achieved the highest F1-score and overall accuracy. Hence, Gradient Boosting was selected as the best-performing model.
- Table 4 shows that for the Customer Churn Prediction dataset, this dataset included customer tenure, account balance, age, number of subscribed products, and salary. Performance metrics were calculated from confusion matrix outputs to determine the best model. Logistic Regression and SVM provided reasonable predictions but lacked depth for complex churn patterns. Decision Tree performed inconsistently. Random Forest improved both Precision and Recall, while Gradient Boosting achieved the highest AUC and F1-score, making it the most effective model for predicting customer churn.

Across all financial datasets, Logistic Regression, SVM, and Decision Tree showed limited performance, while ensemble models consistently achieved higher Recall, F1-score, and AUC. Random Forest and especially Gradient Boosting performed best across all case studies, confirming that metric-based evaluation is essential for reliable model selection.

5. Practical Implication

The findings of this study have several important real-world implications for financial institutions. First, the results show that relying on a single machine learning model is not effective; instead, organizations must evaluate multiple algorithms using performance metrics such as Recall, F1-score, and AUC to select the most reliable model for their specific data. Second, ensemble models like Random Forest and Gradient Boosting consistently provided superior detection capability in credit risk, fraud detection, and customer churn scenarios, indicating that financial systems can significantly improve accuracy and reduce losses by adopting these models. Third, the use of performance-metric–based selection allows banks and insurance companies to align technical decisions with business objectives such as maximizing fraud detection (high Recall) or improving loan approval accuracy (high AUC). Finally, implementing such a metrics-driven approach enhances transparency, regulatory compliance, and operational efficiency, making machine learning deployment more trustworthy and effective in real-world financial environments.

6. Conclusion

This research establishes a performance-metric-based framework for selecting machine learning models in banking and insurance applications. Instead of relying on theoretical assumptions, the study demonstrates that metrics such as Recall, F1-score, and AUC provide objective evidence for determining the most suitable model for each financial task. Across multiple case studies, ensemble methods particularly Random Forest and Gradient Boosting—consistently outperformed linear and single-tree models, proving that model selection must be driven by empirical performance rather than algorithmic theory. The proposed approach enhances decision accuracy, reduces operational risk, and supports transparent, data-driven deployment of machine learning systems in financial institutions.

7. Future Research Direction

Future research can extend this performance-metric-based framework in several meaningful ways. First, additional evaluation metrics such as fairness scores, cost-sensitive loss functions, and model stability indicators can be incorporated to make model selection more aligned with regulatory and ethical requirements in finance. Second, experiments can be expanded to include time-dependent data and concept-drift detection methods to ensure that selected models remain reliable as customer behavior and fraud patterns evolve. Third, integrating explainability techniques such as SHAP or LIME can help financial institutions better understand model decisions and improve transparency for audits. Fourth, benchmarking can be applied across larger, multi-institution datasets to validate the generalizability of the framework. Finally, the framework may be extended into an automated model selection system (AutoML) that dynamically chooses the best model based on real-time performance metrics.

References

[1] D. Wolpert, “The Lack of a Priori Distinctions Between Learning Algorithms,” Neural Computation, 1996.

[2] T. Fawcett, “An introduction to ROC analysis,” Pattern Recognition Letters, 2006.

[3] Pedregosa et al., “Scikit-learn: Machine Learning in Python,” JMLR, 2011. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006. Goodfellow et al., *Deep Learning*, MIT Press, 2016.