Enhanced Credit Card Default Prediction Using SMOTE, LSTM, and Stacked Ensemble Methods

Progress Report

Muhammad Khalid Al Ghifari
Department of Informatics
Syiah Kuala University
Banda Aceh, Indonesia
khalid.22@mhs.usk.ac.id

Andrie Fadhlullah Wahby
Department of Informatics
Syiah Kuala University
Banda Aceh, Indonesia
andri33@mhs.usk.ac.id

***Abstract*—This progress report presents the current status of our credit card default prediction project. We have successfully completed data preprocessing, feature engineering, and implementation of all six baseline models. Preliminary results show that XGBoost achieves the highest accuracy (81.92%) and AUC (0.7794), while Deep Neural Network shows the best F1-score (0.4820). However, all baseline models exhibit poor recall (23-39%), confirming the challenge of detecting defaults in imbalanced datasets. We have also implemented SMOTE-enhanced XGBoost, which improved recall from 36.77% to 48.08%, demonstrating the effectiveness of class balancing. The LSTM hybrid model and stacked ensemble are currently under development. Next steps include completing advanced models, conducting comprehensive evaluation, and performing SHAP analysis for model interpretability.**

***Index Terms*—credit card default, class imbalance, SMOTE, LSTM, ensemble learning, financial risk prediction**

I. Introduction

Credit card default prediction is crucial for financial institutions to manage risk and minimize losses. The primary challenge lies in the inherent class imbalance of credit default datasets, where non-default cases significantly outnumber default cases (typically 3-4:1 ratio). This imbalance causes traditional machine learning models to bias toward the majority class, resulting in high overall accuracy but poor detection rates for actual defaulters.

This project aims to enhance credit card default prediction by addressing class imbalance through Synthetic Minority Over-sampling Technique (SMOTE), incorporating temporal payment patterns via Long Short-Term Memory (LSTM) networks, and leveraging model diversity through stacked ensemble learning. We use the UCI Credit Card Default dataset containing 30,000 customer records from Taiwan with 23 original features.

Our research builds upon the work of Bhandary and Ghosh [1], who compared six machine learning models and found that while deep neural networks achieved 81.80% accuracy, recall remained at only 38.69%. Our goal is to significantly improve recall (target $>$ 50%) while maintaining acceptable overall accuracy, enabling better identification of high-risk customers.

This progress report summarizes our accomplishments to date, presents preliminary experimental results, and outlines remaining work to complete the project.

II. Related Work

Credit default prediction has evolved from traditional statistical methods to advanced machine learning approaches. Early studies relied on linear discriminant analysis (LDA) and logistic regression (LR) due to their interpretability [7]. However, these methods assume linear relationships and cannot capture complex interactions between features [8].

Support vector machines (SVMs) emerged as a robust alternative for classification problems [9], while ensemble methods like Random Forest and XGBoost demonstrated superior performance through combining multiple weak learners [10]. Bhandary and Ghosh [1] recently compared six models on the UCI dataset, finding that deep neural networks achieved 81.80% accuracy but only 38.69% recall, highlighting the recall-accuracy trade-off in imbalanced datasets.

Recent research has focused on addressing class imbalance. SMOTE, introduced by Chawla et al. [11], generates synthetic minority samples to balance class distributions. Suhadolnik et al. [5] demonstrated that ensemble models with boosting algorithms outperform traditional methods in financial risk assessment when combined with proper sampling techniques.

Temporal pattern recognition has gained attention in financial prediction. LSTM networks, designed to capture long-term dependencies [12], have shown promise in time-series financial data. While previous studies focused on static feature analysis [1], [4], the sequential nature of payment behaviors over multiple months remains underexplored.

Stacked ensemble learning, which combines predictions from multiple base models through a meta-learner [13], has proven effective in banking crisis prediction [6]. Our work extends these approaches by systematically combining SMOTE, LSTM, and stacking to address class imbalance and temporal patterns simultaneously.

III. Methodology

A. Dataset and Preprocessing

We utilized the UCI Credit Card Default dataset [14], containing 30,000 customer records from Taiwan with 23 features and one binary target variable (default status). The dataset exhibits a 3.52:1 class imbalance ratio, with 23,364 non-default cases (77.88%) and 6,636 default cases (22.12%).

Features include demographic information (credit limit, gender, education, marital status, age), payment status for six months (April-September 2005), bill amounts, and payment amounts. Data preprocessing involved:

- Removing the ID column
- Verifying no missing values (dataset is complete)
- Applying StandardScaler normalization to ensure feature scales do not dominate model learning
- Splitting data into 80% training (24,000 samples) and 20% testing (6,000 samples) using stratified sampling to maintain class distribution

B. Feature Engineering

We performed feature engineering to create nine additional features that capture important financial behaviors:

- **BILL_SUM**: Total bill amount over 6 months
- **PAY_SUM**: Total payment amount over 6 months
- **BILL_AVG**: Average bill amount
- **PAY_AVG**: Average payment amount
- **PAY_BILL_RATIO**: Payment-to-bill ratio (financial health indicator)
- **CREDIT_UTIL_RATE**: Credit utilization rate (bill / credit limit)
- **TOTAL_DELAYS**: Count of delayed payments (PAY > 0)
- **AVG_DELAY**: Average payment delay across 6 months
- **PAYMENT_TREND**: Recent vs. old payment comparison (PAY_AMT1 - PAY_AMT6)

These engineered features increased our feature set from 23 to 32 features, providing models with richer information about customer payment behaviors and financial status.

C. Baseline Models Implementation

Following Bhandary and Ghosh [1], we implemented six baseline models:

1. Linear Discriminant Analysis (LDA): Projects data onto a lower-dimensional space maximizing class separability using the discriminant function $Z = \beta_0 + \sum_{i=1}^{n} \beta_i x_i$.

2. Logistic Regression (LR): Models the probability of default using the logistic function $P = \frac{1}{1+e^{-(a+bx)}}$.

3. Support Vector Machine (SVM): With RBF kernel, finds the optimal separating hyperplane by minimizing $\frac{1}{n}\sum_{i=1}^{n} \varsigma_i + \lambda||w||^2$ subject to $y_i(w^T x_i - b) \geq 1 - \varsigma_i$.

4. XGBoost: Uses gradient boosting with objective function $L_t = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t)$, with 100 estimators, max depth 6, and learning rate 0.1.

5. Random Forest: Constructs 100 decision trees using Gini impurity $Gini = 1 - \sum_{i=1}^{c}(p_i)^2$, with max depth 10.

6. Deep Neural Network (DNN): Four hidden layers (64-32-16-8 neurons) using ReLU activation and dropout (0.3, 0.3, 0.2) for regularization, trained for 25 epochs with Adam optimizer.

All models were trained on scaled data and evaluated using 10-fold stratified cross-validation to ensure robust performance estimates.

D. SMOTE Implementation

To address class imbalance, we applied SMOTE [11] to the training set. SMOTE creates synthetic minority class samples by interpolating between existing minority samples and their k-nearest neighbors (k=5). This balanced the training set from 18,691:5,309 to 18,691:18,691 (1:1 ratio), ensuring the model learns both classes equally without discarding majority samples.

We trained XGBoost on the SMOTE-balanced data to evaluate the effectiveness of class balancing on model performance.

E. Evaluation Metrics

We evaluated models using metrics suitable for imbalanced classification:

- **Accuracy**: $\frac{TP+TN}{TP+TN+FP+FN}$
- **Precision**: $\frac{TP}{TP+FP}$
- **Recall (Sensitivity)**: $\frac{TP}{TP+FN}$ - primary metric
- **Specificity**: $\frac{TN}{TN+FP}$
- **F1-Score**: $2 \times \frac{Precision \times Recall}{Precision+Recall}$
- **G-Mean**: $\sqrt{Recall \times Specificity}$
- **AUC**: Area under ROC curve

Given the business context where detecting defaults is critical, we prioritized recall over accuracy.

IV. Preliminary Experiments and Results

A. Experimental Setup

All experiments were conducted using Python 3.12 with scikit-learn 1.3, XGBoost 2.0, TensorFlow 2.15, and imbalanced-learn libraries. Models were trained on Google Colab with standard computing resources. We used 10-fold stratified cross-validation for baseline models to ensure robust performance estimates.

B. Baseline Model Results

Table I presents the performance of six baseline models on the test set. XGBoost achieved the highest accuracy (81.92%) and AUC (0.7794), while DNN showed the best F1-score (0.4820) and G-mean (0.6027). However, all baseline models exhibited poor recall, ranging from 0.2308 (SVM) to 0.3869 (DNN), indicating limited ability to detect actual defaults.

Traditional methods (LDA, LR, SVM) showed higher specificity (>0.97) but lower sensitivity, while modern methods (XGBoost, RF, DNN) achieved better balance between precision and recall. These results successfully replicate the findings of Bhandary and Ghosh [1], confirming the baseline performance benchmarks.

Cross-validation results showed consistent performance across folds:

TABLE I
BASELINE MODEL PERFORMANCE ON TEST SET

Model	Accuracy	Recall	F1-Score	AUC
LDA	0.8090	0.2475	0.3619	0.7200
LR	0.8100	0.2369	0.3530	0.7300
SVM	0.8105	0.2308	0.3477	0.7100
XGBoost	**0.8192**	0.3677	0.4736	**0.7794**
RF	0.8163	0.3663	0.4661	0.7785
DNN	0.8180	0.3869	**0.4820**	0.7690

- LDA CV Accuracy: 0.8101 ± 0.0068
- LR CV Accuracy: 0.8080 ± 0.0066
- XGBoost CV Accuracy: 0.8209 ± 0.0040
- RF CV Accuracy: 0.8210 ± 0.0033

C. SMOTE-Enhanced XGBoost Results

Table II compares XGBoost performance with and without SMOTE. SMOTE-enhanced XGBoost improved recall from 0.3677 to 0.4808 (+30.8%), demonstrating the effectiveness of class balancing. While accuracy decreased slightly to 79.70%, the F1-score increased to 0.5116, indicating better overall performance on the minority class.

TABLE II
IMPACT OF SMOTE ON XGBOOST PERFORMANCE

Model	Accuracy	Recall	F1-Score	AUC
XGBoost	0.8192	0.3677	0.4736	0.7794
XGB + SMOTE	0.7970	**0.4808**	**0.5116**	0.7664
Improvement	-2.71%	**+30.8%**	**+8.02%**	-1.67%

The confusion matrix analysis reveals that SMOTE-enhanced XGBoost correctly identified 638 out of 1,327 default cases in the test set, compared to 488 for baseline XGBoost—an increase of 150 correctly detected defaults. The 3% accuracy decrease is an acceptable trade-off in financial contexts where missing a default case (false negative) is far more costly than a false alarm (false positive).

D. Exploratory Data Analysis Findings

Our exploratory analysis revealed several important insights:

Target Distribution: The 22.12% default rate confirms significant class imbalance (3.52:1 ratio), justifying the need for SMOTE.

Payment Delays: Analysis of payment status features (PAY_0 through PAY_6) shows that 22.73% of customers had delayed payments in September 2005 (most recent month), decreasing to 10.26% in April 2005. This temporal pattern suggests recent payment behavior is a strong indicator of default risk.

Credit Utilization: The median credit utilization rate is 0.16, but defaulters show significantly higher utilization (mean 0.23) compared to non-defaulters (mean 0.14).

Engineered Features: The TOTAL_DELAYS feature shows a strong correlation with default (0.40), emerging as the most important predictor. Customers with 3+ payment delays have a much higher default probability.

Demographic Patterns: Gender, education, and marital status show minimal importance, suggesting that behavioral patterns are more predictive than static demographics.

E. ROC Curve Analysis

ROC curves for all baseline models show that XGBoost and Random Forest achieve the highest AUC (0.77-0.78), indicating strong discrimination ability. All models significantly outperform random classification (diagonal line). The curves demonstrate that while overall discrimination is good, the optimal operating point needs to favor higher recall, which SMOTE helps achieve.

V. DISCUSSION

Our preliminary results confirm the challenge of class imbalance in credit card default prediction. Baseline models achieve high accuracy (81-82%) but poor recall (23-39%), missing the majority of actual default cases. This aligns with findings from Bhandary and Ghosh [1] and validates our approach of prioritizing recall improvement.

The SMOTE results are particularly encouraging. The 30.8% recall improvement demonstrates that addressing class imbalance can significantly enhance default detection. The 3% accuracy decrease is acceptable given the business context: assuming an average default loss of $10,000 per customer, detecting 150 additional defaults saves approximately $1.5 million per 6,000 customers, far outweighing the cost of additional false alarms.

Feature engineering proved valuable, with TOTAL_DELAYS emerging as the most important predictor. This suggests that monitoring accumulated payment delays could serve as an effective early warning system for financial institutions.

The successful replication of baseline results validates our experimental methodology and provides a solid foundation for evaluating advanced approaches. The consistent cross-validation performance indicates that our models are not overfitting and should generalize well to new data.

However, several challenges remain. The current best recall (48.08%) is approaching but has not yet reached our target of 50%. The F1-score of 0.5116, while improved, indicates room for further enhancement. These results motivate the need to complete the LSTM hybrid model and stacked ensemble to achieve our performance goals.

VI. NEXT STEPS

To complete the project, we will focus on the following tasks:

A. Immediate Priority (Weeks 7-8)

1. LSTM Hybrid Model Development

- Complete implementation of dual-branch architecture
- Temporal branch: Process payment history (6 months × 3 features) through two LSTM layers (64, 32 units)
- Static branch: Process demographic features through dense layer (32 units)

- Merge branches and train for 30 epochs with early stopping
- Evaluate temporal pattern learning capability

2. Stacked Ensemble Implementation

- Train base learners (XGBoost, Random Forest, Logistic Regression) on SMOTE-balanced data
- Implement Logistic Regression meta-learner with 5-fold cross-validation
- Evaluate ensemble performance
- Target: Recall $> 50\%$, F1-Score $> 50\%$

B. Analysis and Interpretation (Weeks 9-10)

3. Comprehensive Model Comparison

- Generate ROC curves for all models
- Perform statistical significance testing (McNemar's test)
- Analyze confusion matrices for business impact
- Calculate financial metrics (cost-benefit analysis)

4. SHAP Analysis for Explainability

- Apply SHAP (SHapley Additive exPlanations) to best-performing model
- Identify top features contributing to default predictions
- Generate waterfall plots for individual predictions
- Compare feature importance across different models
- Provide actionable insights for risk management

5. Final Report and Presentation

- Complete comprehensive results section
- Develop business recommendations based on findings
- Create presentation slides
- Finalize GitHub repository with documentation
- Prepare for project presentation

C. Expected Final Outcomes

Based on preliminary results, we anticipate:

- Stacked ensemble achieving recall $> 50\%$ and F1-score $> 50\%$
- LSTM demonstrating effective temporal pattern learning (AUC > 0.75)
- Overall improvement of 25-30% in recall compared to baseline
- Detection of 150-200 additional default cases per 6,000 customers
- Clear identification of top 5-10 risk factors through SHAP analysis

D. Potential Challenges

We anticipate several challenges:

- LSTM training time may require computational resource optimization
- Hyperparameter tuning for ensemble may be time-intensive
- Balancing recall improvement with acceptable accuracy trade-off
- Ensuring model interpretability for business stakeholders

We have allocated sufficient time in our schedule to address these challenges and maintain project timeline.

VII. Conclusion

This progress report demonstrates substantial accomplishment toward our research objectives. We have successfully completed data preprocessing, feature engineering, and all baseline model implementations. Our preliminary results validate the research approach: SMOTE-enhanced XGBoost achieved 30.8% recall improvement, demonstrating that addressing class imbalance significantly enhances default detection.

The successful replication of Bhandary and Ghosh [1] baseline results establishes credibility and provides solid benchmarks for evaluating our advanced approaches. The remaining work—LSTM hybrid model, stacked ensemble, and comprehensive analysis—is well-defined and achievable within the project timeline.

We are confident that the completed project will demonstrate significant improvements in credit card default prediction and provide valuable insights for financial risk management. The next phase will focus on completing advanced models and conducting thorough analysis to deliver actionable recommendations for practitioners.

Acknowledgment

We thank Syiah Kuala University for providing computational resources and support for this research. We also acknowledge the valuable feedback from our project mentor during the development of baseline models.

References

[1] R. Bhandary and B. K. Ghosh, "Credit Card Default Prediction: An Empirical Analysis on Predictive Performance Using Statistical and Machine Learning Methods," *Journal of Risk and Financial Management*, vol. 18, no. 1, p. 23, 2025, doi: 10.3390/jrfm18010023.

[2] C.-H. Chang, "Information asymmetry and card debt crisis in Taiwan," *Bulletin of Applied Economics*, vol. 9, no. 2, pp. 123–145, 2022.

[3] GlobalData, *Taiwan Cards and Payments Market Report Overview*. GlobalData Report Store, 2023.

[4] M. Hamdi, S. Mestiri, and A. Arbi, "Artificial intelligence techniques for bankruptcy prediction of Tunisian companies: An application of machine learning and deep learning-based models," *Journal of Risk and Financial Management*, vol. 17, no. 4, p. 132, 2024, doi: 10.3390/jrfm17040132.

[5] N. Suhadolnik, J. Ueyama, and S. Da Silva, "Machine learning for enhanced credit risk assessment: An empirical approach," *Journal of Risk and Financial Management*, vol. 16, no. 12, p. 496, 2023, doi: 10.3390/jrfm16120496.

[6] S. Puli, N. Thota, and A. C. V. Subrahmanyam, "Assessing machine learning techniques for predicting banking crises in India," *Journal of Risk and Financial Management*, vol. 17, no. 4, p. 141, 2024, doi: 10.3390/jrfm17040141.

[7] D. J. Hand and W. E. Henley, "Statistical classification methods in consumer credit scoring: A review," *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, vol. 160, no. 3, pp. 523–541, 1997, doi: 10.1111/j.1467-985X.1997.00078.x.

[8] I.-C. Yeh and C.-H. Lien, "The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients," *Expert Systems with Applications*, vol. 36, no. 2, pp. 2473–2480, 2009, doi: 10.1016/j.eswa.2007.12.020.

[9] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, pp. 189–215, 2020, doi: 10.1016/j.neucom.2019.10.118.

[10] 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*, 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[11] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002, doi: 10.1613/jair.953.

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[13] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992, doi: 10.1016/S0893-6080(05)80023-1.

[14] I.-C. Yeh, "Default of credit card clients," *UCI Machine Learning Repository*, 2016, doi: 10.24432/C55S3H.