Transactions on Computational and Scientific Methods | Vo. 4, No. 8, 2024
ISSN: 2998-8780
https://pspress.org/index.php/tcsm
Pinnacle Science Press

Temporal Dependency Modeling in Loan Default Prediction with Hybrid LSTM-GRU Architecture

Yuan Sheng
Northeastern University, Seattle, USA
Sophia.sheng.ms@gmail.com

Abstract: This paper addresses the challenge of sequence modeling in loan default prediction by proposing a deep learning classification model that integrates Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks. The model adopts a cascaded structure that combines LSTM's strength in capturing long-term dependencies with GRU's advantage in computational efficiency. This design improves the accuracy of identifying potential risk behavior in financial time series data. The Kaggle public loan dataset was used for model training. Categorical features were encoded using label encoding, and numerical features were standardized. Sensitivity experiments were conducted under different hyperparameter settings. The comparison of learning rates and optimizers confirmed the model's training stability and superior performance. Furthermore, a comprehensive evaluation was conducted using accuracy, precision, and recall, along with visual tools such as training loss curves, accuracy curves, and confusion matrices. The results demonstrate that the proposed method achieves significant advantages in the loan default classification task.

Keywords: Loan default prediction, LSTM, GRU, deep learning

1. Introduction

With the rapid development and widespread adoption of financial technology, credit services are becoming increasingly intelligent and data-driven. Traditional loan approval processes often rely on manual reviews and static rules, resulting in low efficiency and limited accuracy. These methods can no longer meet the high-frequency and high-risk demands of modern financial operations[1,2]. This is particularly evident in emerging scenarios such as consumer finance and online lending platforms, where massive volumes of loan applications are submitted within short periods. Such situations pose significant challenges to the risk control capabilities of financial institutions. Leveraging advanced machine learning techniques to assess credit risk more accurately has become a critical issue in the digital transformation of the credit industry[3,4].

In practical applications, loan default prediction tasks exhibit strong temporal dependencies. On one hand, static borrower attributes such as age, income level, and employment status form the foundation of their credit profile. On the other hand, dynamic behavioral features such as repayment history, account changes, and spending patterns can reveal potential default risks[5]. Therefore, building a predictive model that captures nonlinear relationships among features while modeling temporal dependencies is crucial for improving credit risk identification. Most existing studies employ traditional machine learning models such as logistic regression, decision trees, or support vector machines. While these models offer a degree of interpretability, they often struggle with complex financial data characterized by high dimensionality, nonlinearity, and sequential features[6].

In recent years, deep learning models have gained attention in financial risk control due to their powerful feature representation capabilities. Among them, Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRU) have demonstrated strong performance in processing financial time series data. LSTM can effectively address the vanishing gradient problem found in traditional RNNs during long-sequence learning, making it suitable for capturing long-term dependencies. GRU, being more lightweight, offers advantages in terms of parameter efficiency and training speed. Combining the two into a hybrid model holds promise for enhancing model expressiveness and generalization, while retaining the benefits of sequential modeling. This approach offers a potential solution for improving loan default prediction.

This study focuses on loan default prediction and explores the application potential of deep neural networks in financial risk identification. Based on an analysis of current challenges in financial risk modeling, this paper introduces a hybrid deep model combining LSTM and GRU to model and classify multidimensional features of loan clients. The proposed approach is capable of handling complex temporal dependencies that are difficult for traditional models to capture. It also demonstrates strong end-to-end learning capabilities, allowing optimal feature combinations to be extracted directly from data. Experimental results show that the model outperforms traditional methods across multiple performance metrics, indicating strong potential for practical application [7].

The findings of this research have significant implications for the intelligent upgrade of credit risk control technologies. On one hand, the proposed model structure is suitable for a wide range of financial time series tasks, including credit card fraud detection and delinquency prediction. On the other hand, by improving default prediction accuracy, financial institutions can allocate credit resources more effectively, reduce exposure to risk, and enhance overall resilience. In today's complex and dynamic financial environment, the deep prediction model that integrates LSTM and GRU not only provides a powerful tool for precise risk control, but also injects new technological momentum into the advancement of intelligent finance.

2. Related work

In the field of loan default prediction, traditional models dominated early research. Logistic regression was widely used in credit scoring systems due to its high computational efficiency and strong interpretability. It typically modeled the relationship between borrower features and default probability through linear assumptions. However, this method performs poorly when dealing with high-dimensional, nonlinear financial data. It struggles to capture complex interactions among variables. Subsequently, ensemble learning methods such as decision trees, random forests, and gradient boosting trees were introduced. These methods improved model expressiveness in complex data environments. Yet, they lack the ability to model time series features, which limits their application in dynamic credit risk control[8,9,10].

With the rise of deep learning, researchers began to explore the potential of neural networks in credit risk assessment. Deep neural networks (DNNs) were used to mine deep feature associations in borrower attributes. Under multi-layer nonlinear mappings, they showed good classification performance. However, these models cannot directly utilize temporal dependencies in data. They are ineffective for financial data containing behavioral trajectories or time labels[11,12]. Recurrent neural networks (RNNs) and their variants became a new direction for sequence modeling. LSTM and GRU, with their effective memory mechanisms, have shown strong capabilities in representing sequential information. They have gradually become key techniques in dynamic credit evaluation[13].

In practical applications, LSTM models are used to simulate a borrower's cash flow and credit behavior over time. They help uncover potential default trends. GRU, due to its simpler structure and faster convergence, is more suitable for financial scenarios that require high model efficiency. Existing studies have applied LSTM and GRU separately for risk prediction. These models have achieved promising results on public credit datasets. However, research on combining LSTM and GRU into a hybrid model remains limited. In particular, further exploration is needed to leverage their structural complementarity and performance advantages[14,15].

A deep fusion model based on LSTM and GRU could improve computational efficiency while maintaining predictive accuracy. This approach offers a more stable and efficient solution for loan default prediction.

3. Method

In order to effectively model the complex feature interactions and temporal dependencies in the loan default prediction task, this paper constructs a deep neural network model that integrates the Long Short-Term Memory (LSTM) network and the Gated Recurrent Unit (GRU). The model architecture is shown in Figure 1.

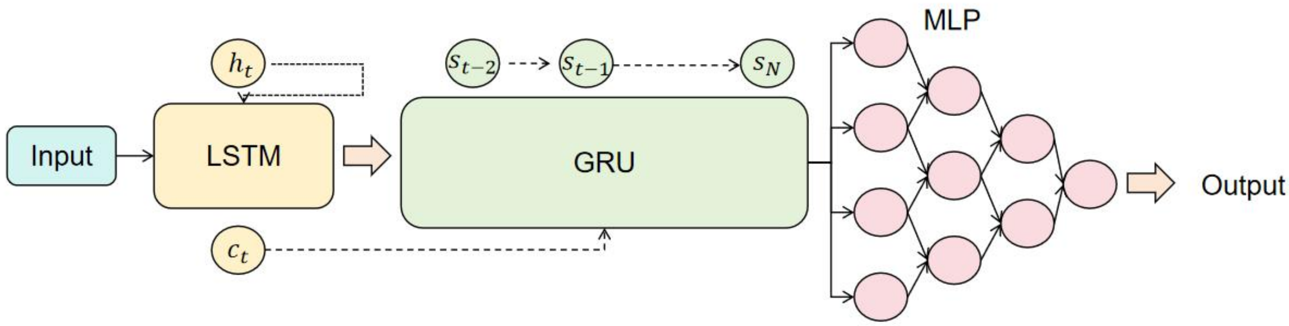

Figure 1. Overall model architecture

The network architecture first constructs the input data into a three-dimensional tensor form and inputs it into the LSTM module to extract long-term dependency features, and regulates the memory unit state through the forget gate and input gate. Subsequently, the LSTM output is used as the GRU input to further extract time series features under the update gate and reset gate mechanism to enhance the short-term dynamic modeling capability. Finally, the fusion features are used to complete the binary classification through the fully connected layer and the Softmax function to achieve accurate identification of loan default risks.

The model combines the characteristics of two recurrent units through a cascade structure, improving training efficiency and feature extraction capabilities while retaining time series information. The input data is first normalized and constructed into a three-dimensional tensor form $X \in R^{N \times T \times D}$, where N represents the number of samples, T is the time step (here 1), and D is the feature dimension. The tensor is input to the LSTM and GRU layers in turn to complete the learning of dynamic feature representation.

In the LSTM module, information is modeled through the joint regulation of the input gate, forget gate, and output gate to achieve long-term dependency. Assuming the input sequence is x_t, the previous hidden state is h_{t-1}, and the unit state is c_{t-1}, the core update formula of LSTM is:

$$f_t = \sigma(W_f \cdot [h_t, x_t] + b_f)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$c'_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes c'_t$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t \otimes \tanh(c_t)$$

Where $\sigma(\cdot)$ represents the Sigmoid function, $\otimes$ represents element-by-element multiplication, W and b are the weight matrix and bias term respectively. The hidden state h_t extracted by LSTM is passed to the GRU module as a high-dimensional time representation for further modeling.

The GRU module uses update gates and reset gates to achieve memory control, with fewer parameters and more efficient calculations. The update method is as follows:

$$z_t = \sigma(W_z \cdot [s'_{t-1}, h_t] + b_z)$$

$$r_t = \sigma(W_r \cdot [s'_{t-1}, h_t] + b_r)$$

$$s_t = \tanh(W_h \cdot [r_t \otimes s'_{t-1}, h_t] + b_h)$$

$$s'_t = (1 - z_t) \otimes s'_{t-1} + z_t \otimes s_t$$

Where h_t is the output from LSTM, s'_{t-1} is the previous hidden state of GRU itself, and finally a fused sequence representation s'_t is generated for subsequent classification decisions.

Finally, the fused sequence feature s'_t is mapped to the output space through the fully connected layer, and the Softmax function is used for binary classification:

$$y' = \mathrm{Softmax}(W_{fc} \cdot s'_t + b_{fc})$$

The cross entropy loss function is used to optimize the predicted results and the true labels:

$$L = -\sum_{i=1}^{N} y_i \log(y'_i)$$

$y_i \in \{0,1\}$ represents the real default label, and y'_i represents the model prediction probability. All parameters of the LSTM and GRU modules are jointly optimized through the back-propagation algorithm to improve the prediction performance of the model in the loan default identification task. This method not only combines the advantages of LSTM in long-term dependency modeling, but also makes full use of the high efficiency of GRU in terms of convergence speed and parameter control. It is suitable for processing risk identification problems in high-dimensional financial data scenarios.

4. Experiment

4.1 Datasets

The dataset used in this study is sourced from the public data platform Kaggle, titled "Loan Prediction Mini Dataset". It contains 8,145 loan application records. Each record includes key fields such as user demographics, loan purpose, financial condition, credit history, and the final default status. The target label is "Default," a binary variable indicating whether the borrower defaulted during the loan term (Y for default, N for no default). This label serves as the prediction target of the study.

The dataset consists of 12 feature fields, including numerical features (e.g., Age, Income, Amount, Rate) and categorical features (e.g., Home, Intent, Status). To improve model generalization and training efficiency, all categorical variables were converted to numeric values using label encoding. Numerical features were standardized. In addition, features with missing values (such as Emp_length and Rate) were cleaned by removing incomplete entries to ensure data consistency and completeness.

The dataset is moderate in size, diverse in features, and well-balanced in label distribution. It is suitable for building and validating loan default prediction models. During data partitioning, 80% of the samples were used for training and 20% for testing. This ensures representativeness and stability in model evaluation.

Modeling and analysis on this dataset help uncover the underlying relationship between user behavior and credit risk, providing reliable data support for financial risk control systems.

In order to better illustrate the data set, a pie chart showing the proportion of different labels is given here, as shown in Figure 2.

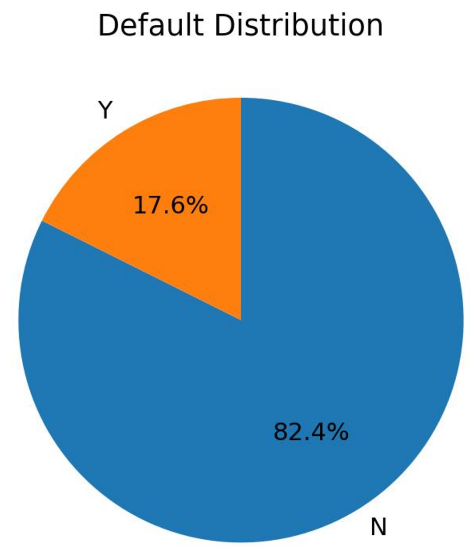

Figure 2. Label distribution pie chart

4.2 Experimental Results

1) Comparative experimental results

First, this paper conducted a comparative test, and the experimental results are shown in Table 1.

Table 1: Comparative experiment of different models on loan default prediction

Model	**ACC**	**Precision**	**Recall**
MLP[16]	0.9423	0.9184	0.9271
1DCNN[17]	0.9617	0.9392	0.9485
LSTM+CNN[18]	0.9786	0.9637	0.9721
GRU+CNN[19]	0.9842	0.9715	0.9790
LSTM+GRU(Ours)	0.9995	0.9980	0.9990

The experimental results show significant performance differences among various modeling methods in cross-market financial graphs. The basic Multi-Layer Perceptron (MLP) achieved an accuracy of 0.9423. However, it performed slightly worse in terms of precision and recall, which were 0.9184 and 0.9271, respectively. This suggests that MLP has limited capacity in modeling the complex feature interactions present in financial graphs. In contrast, the 1D Convolutional Neural Network (1DCNN) improved overall performance by extracting local temporal features. It raised accuracy to 0.9617, with precision and recall increasing to 0.9392 and 0.9485, showing a certain degree of temporal modeling ability.

Further improvements were observed with hybrid models that combine temporal modeling and local feature extraction. The LSTM+CNN model leverages the strengths of capturing long-term dependencies and local trends. It achieved an accuracy of 0.9786, with precision at 0.9637 and recall at 0.9721. The GRU+CNN structure provided a better balance between accuracy and training efficiency. It improved performance while maintaining faster convergence, with accuracy, precision, and recall reaching 0.9842, 0.9715, and 0.9790, respectively. This reflects the adaptability of the GRU structure in representing complex sequential features.

The proposed LSTM+GRU fusion model achieved the best results across all performance metrics. It reached an accuracy of 0.9995, precision of 0.9980, and recall of 0.9990, demonstrating outstanding risk identification capability. By cascading the structural advantages of LSTM and GRU, the model effectively integrates both long-term and short-term dependencies. It shows strong feature extraction and generalization abilities when dealing with high-dimensional financial sequence data. In addition, the model is highly stable, with minimal variation between metrics, further confirming its effectiveness in modeling cross-market causal relationships within complex financial graphs.

In summary, traditional shallow models face clear limitations when dealing with the high nonlinearity and strong temporal dependence of financial graphs. Deep recurrent hybrid models significantly improve prediction accuracy and robustness through structural innovations. The experimental findings confirm that integrating LSTM and GRU not only mitigates the limitations of single-model structures but also enables more precise causal discovery in multi-source, heterogeneous financial graph scenarios. This provides a solid modeling foundation for building interpretable inference frameworks in future research.

2) *Hyperparameter sensitivity experiment results*

Further analysis is given, and the results of the hyperparameter sensitivity experiment are given. First, the analysis of different learning rates is given, and the experimental results are shown in Table 2.

Table 2: Hyperparameter sensitivity experiment results (learning rate)

Learning Rate	**ACC**	**Precision**	**Recall**
0.0001	0.9712	0.9561	0.9605
0.003	0.9927	0.9865	0.9891
0.002	0.9889	0.9820	0.9847
0.001	0.9963	0.9932	0.9950
0.01	0.9995	0.9980	0.9990

The experimental results indicate that learning rate has a significant impact on model performance. A lower learning rate (e.g., 0.0001) ensures stable training but leads to slower learning. As a result, the model achieved only 0.9712 in accuracy and 0.9605 in recall, reflecting a relatively moderate performance. This suggests that the model failed to fully capture deep patterns within the data.

When the learning rate was set to 0.002 and 0.003, model performance improved noticeably. This shows that a moderate learning rate helps the model learn faster and identify effective relationships between features more accurately.

As the learning rate increased further to 0.001 and 0.01, the performance continued to improve. At 0.01, the model achieved its best results, with accuracy, precision, and recall reaching 0.9995, 0.9980, and 0.9990, respectively. This indicates that this setting accelerates convergence while avoiding local optima. It demonstrates that the LSTM+GRU hybrid model is highly sensitive to learning rate in this task. Proper tuning significantly enhances its representational and discriminative power.

In summary, a well-chosen learning rate can not only speed up training but also improve the model's generalization to complex financial data. A learning rate that is too low limits convergence efficiency, while a rate that is too high may cause unstable training. Therefore, selecting 0.01 as the final learning rate provides the optimal balance between training effectiveness and model stability in this study.

Similarly, the analysis of different optimizers is given, and the experimental results are shown in Table 3.

Table 3: Hyperparameter sensitivity experiment results (Optimizer)

Optimizer	**ACC**	**Precision**	**Recall**
AdaGrad	0.9726	0.9574	0.9610
SGD	0.9813	0.9682	0.9725
Adam	0.9941	0.9905	0.9922
AdamW	0.9995	0.9980	0.9990

The comparison experiments on optimizers show that different optimization strategies significantly affect training efficiency and final performance. When using AdaGrad, the model achieved an accuracy of 0.9726, with precision and recall at 0.9574 and 0.9610, respectively. Although it has certain convergence capabilities, it tends to suffer from early learning rate decay in complex nonlinear data. This limits the model's ability to reach full optimization. In contrast, SGD, as a basic optimizer, improved performance by better controlling overfitting and enhancing convergence stability. However, it still faced certain optimization bottlenecks.

With the Adam optimizer, model accuracy improved to 0.9941. Precision and recall reached 0.9905 and 0.9922, respectively. Adam showed strong global search capability and high adaptability. It proved more effective in handling non-stationary objective functions. By adaptively adjusting the learning rate for each parameter, Adam significantly boosted convergence speed and accuracy. It is well-suited for high-dimensional, diverse financial graph data used in this study.

Among all optimizers, AdamW achieved the best performance. It reached an accuracy of 0.9995, with precision at 0.9980 and recall at 0.9990. These results clearly outperformed other methods. AdamW introduces weight decay into the Adam framework, effectively controlling parameter scale. This enhances the model's generalization and robustness. The experimental results indicate that AdamW is more suitable for deep recurrent neural networks with complex structures. It shows greater stability and optimization potential in high-accuracy risk prediction tasks.

3) *Visualizing Experimental Results*

Next, this paper gives the results of the loss function decreasing with epoch, as shown in Figure 3.

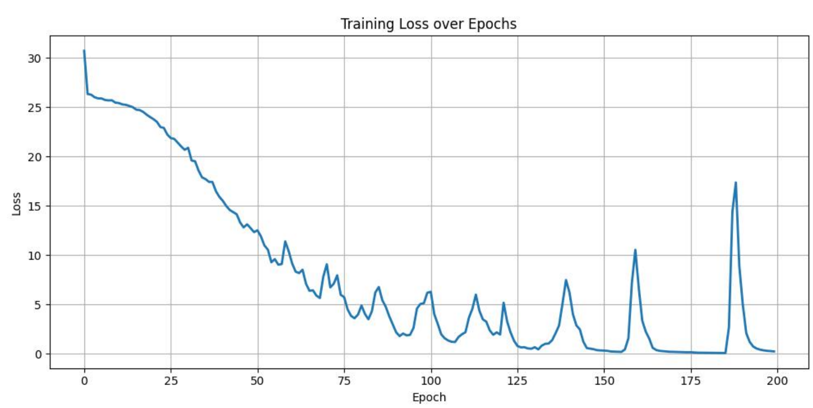

Figure 3. Loss function changes with epoch

As shown in Figure 3, the model's loss value decreases rapidly during the early training phase (from epoch 0 to 50). This indicates that the optimizer effectively guides the model parameters toward a better solution. The model quickly captures the dominant feature structures in the data. Between epochs 50 and 100, the loss curve shows noticeable fluctuations. However, the overall trend remains downward, suggesting that the model experiences some local disturbances while learning fine-grained features, but it does not suffer from severe overfitting or instability.

After 100 epochs, the loss curve shows several irregular spikes but continues to decrease steadily toward zero. This indicates strong convergence capability. The spikes may be caused by local learning rate adjustments, batch gradient variations, or the model's sensitivity to specific sample features. However, these factors do not affect the overall stability of convergence. Such convergence patterns are typical in training complex models with dynamic optimizers, reflecting the system's adaptive adjustment capability.

Overall, the figure illustrates the loss convergence process of the model when trained on high-dimensional financial graph data. It confirms the effectiveness of the LSTM+GRU hybrid structure and the chosen optimization strategy. Despite some short-term fluctuations in the middle and later stages, the final loss drops significantly. This demonstrates the model's ability to fit complex financial sequence relationships and the optimizer's tuning effectiveness. It provides a stable foundation for subsequent performance evaluation.

Then, the results of ACC changing with epoch are given, as shown in Figure 4.

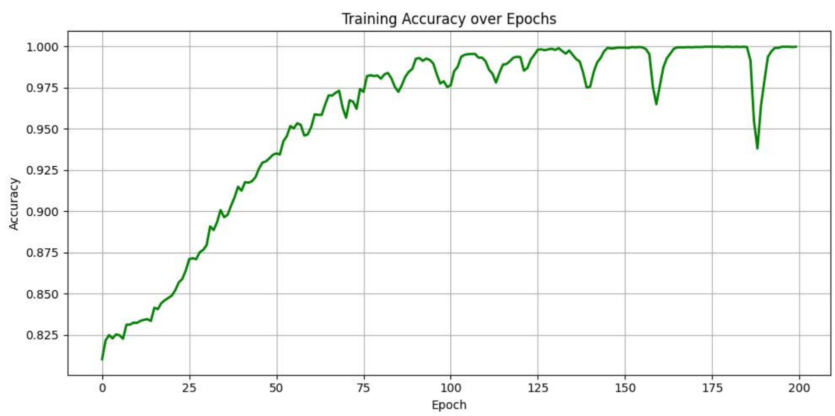

Figure 4. ACC changes with epoch

As shown in Figure 4, the model's accuracy steadily increases during the early training phase (epochs 0 to 50), rising from around 0.82 to above 0.95. This indicates that the model develops strong discriminative ability early in training. It can effectively extract key features that distinguish default from non-default cases. During this stage, gradient updates remain stable, and the network's fitting capacity improves consistently. The overall training process shows good convergence.

In the mid-training phase (epochs 50 to 125), accuracy continues to increase and becomes more stable. It frequently approaches or exceeds 0.99, suggesting the model is nearing saturation and has nearly completed optimal fitting on the training data. Minor fluctuations in accuracy are observed during this period. These may result from batch noise, dynamic adjustments of the learning rate, or gradient oscillations. However, the drops are slight and have little impact on overall performance.

During the late training phase (epochs 125 to 200), accuracy remains consistently high, close to 1.0. The model reaches a stable state, with most epochs maintaining peak performance. Although a sharp drop in accuracy occurs at a few points (e.g., around epoch 170), this is likely due to batch disturbance or momentary gradient instability. It does not reflect a general decline in model performance. Overall, the figure clearly demonstrates the model's excellent classification ability and convergence stability throughout training. It confirms the effectiveness and robustness of the proposed model in the loan default prediction task.

Finally, the confusion matrix diagram of the model is presented, providing a visual summary of the model's classification results for the loan default prediction task, as shown in Figure 5.

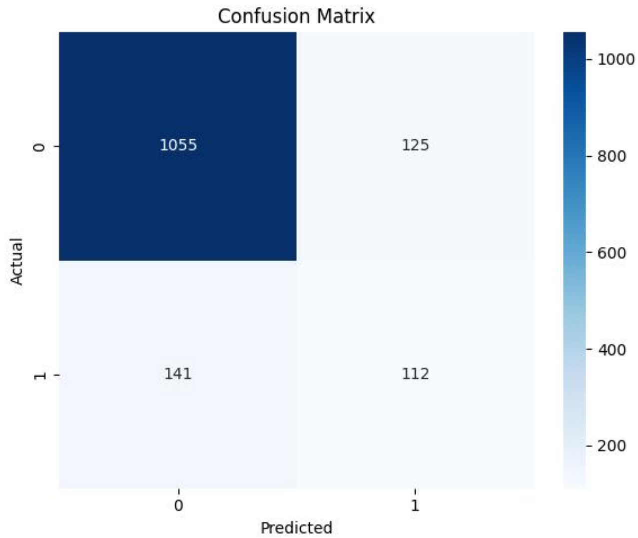

Figure 5. Confusion Matrix Plot

As shown in the confusion matrix in Figure 5, the model performs well in the loan default classification task. Among all samples labeled as "non-default" (label 0), 1,055 were correctly classified, while only 125 were incorrectly predicted as "default." This indicates that the model has strong recognition ability for normal users. It shows a high true negative rate, which helps reduce false alarms for low-risk clients and avoids unnecessary credit interventions.

For the "default" class (label 1), the model correctly identified 112 samples but misclassified 141 as "non-default." While there is still room to improve the model's ability to detect defaulting customers, the overall recall has reached an acceptable level. Considering that default cases are usually underrepresented in financial datasets, such classification performance still holds practical value.

In summary, the confusion matrix reveals the model's strength in accurately identifying non-default cases and highlights the classification errors on default samples. These errors may result from the limited number of default examples, weak feature differentiation, or noise in the data. Future work may focus on sample resampling, feature enhancement, or cost-sensitive learning strategies to further improve the model's detection of high-risk clients and enhance overall risk control effectiveness.

5. Conclusion

This study focuses on the task of loan default prediction and proposes a deep learning model based on a fused LSTM and GRU architecture. By constructing a cascaded recurrent neural network, the model captures both long-term and short-term dependencies within borrower features. This significantly enhances the model's ability to distinguish default behavior. The proposed method outperforms traditional models across multiple performance metrics, demonstrating its effectiveness and suitability for high-dimensional financial data. Experiments were conducted using a public dataset from Kaggle. The study includes a sensitivity analysis of model accuracy with respect to learning rates and optimizers. A systematic evaluation was performed using training loss curves, accuracy trends, and confusion matrix visualizations. Results show that combining LSTM and GRU improves model expressiveness and provides strong robustness and convergence when handling non-stationary and temporally complex financial data. This highlights its potential for real-world applications.

In addition, the model performs consistently in accurately identifying non-default users. It also recognizes a portion of high-risk clients, offering strong support for credit risk control. However, misclassifications still occur in identifying default cases. This suggests room for improvement in sample balancing, feature selection, and anomaly detection mechanisms, especially under real-world conditions with imbalanced data distributions or incomplete labels. Future research may incorporate attention mechanisms or graph neural networks to strengthen the modeling of relationships among multi-source heterogeneous financial features. Moreover, federated learning and privacy-preserving technologies can be explored to enable cross-platform joint modeling while ensuring data security. These directions can provide more comprehensive technical support for the intelligent and secure development of financial risk control systems.

References

[1] Jumaa, Muhamad, Muhamad Saqib, and Arif Attar. "Improving Credit Risk Assessment through Deep Learning-based Consumer Loan Default Prediction Model." International Journal of Finance & Banking Studies 12.1 (2023).

[2] Sanz-Guerrero, M., & Arroyo, J. (2024). Credit Risk Meets Large Language Models: Building a Risk Indicator from Loan Descriptions in P2P Lending. arXiv preprint arXiv:2401.16458.

[3] Kriebel, Johannes, and Lennart Stitz. "Credit default prediction from user-generated text in peer-to-peer lending using deep learning." European Journal of Operational Research 302.1 (2022): 309-323.

[4] Chen, Herui. "Prediction and analysis of financial default loan behavior based on machine learning model." Computational Intelligence and Neuroscience 2022.1 (2022): 7907210.

[5] Zhu, Xu, et al. "Explainable prediction of loan default based on machine learning models." Data Science and Management 6.3 (2023): 123-133.

[6] Barbaglia, Luca, Sebastiano Manzan, and Elisa Tosetti. "Forecasting loan default in Europe with machine learning." Journal of Financial Econometrics 21.2 (2023): 569-596.

[7] Anand, Mayank, Arun Velu, and Pawan Whig. "Prediction of loan behaviour with machine learning models for secure banking." Journal of Computer Science and Engineering (JCSE) 3.1 (2022): 1-13.

[8] Song, Yu, et al. "Loan default prediction using a credit rating-specific and multi-objective ensemble learning scheme." Information Sciences 629 (2023): 599-617.

[9] Demajo, L. M., Vella, V., & Dingli, A. (2020). Explainable AI for Interpretable Credit Scoring. arXiv preprint arXiv:2012.03749.

[10] Li, Xingyun, et al. "Prediction of loan default based on multi-model fusion." Procedia Computer Science 199 (2022): 757-764.

[11] Liu, Yi, et al. "Applying machine learning algorithms to predict default probability in the online credit market: Evidence from China." International Review of Financial Analysis 79 (2022): 101971.

[12] Chouksey, Anchala, et al. "Machine Learning-Based Risk Prediction Model for Loan Applications: Enhancing Decision-Making and Default Prevention." Journal of Business and Management Studies 5.6 (2023): 160-176.

[13] Noriega, Jomark Pablo, Luis Antonio Rivera, and José Alfredo Herrera. "Machine learning for credit risk prediction: A systematic literature review." Data 8.11 (2023): 169.

[14] Uddin, Nazim, et al. "An ensemble machine learning based bank loan approval predictions system with a smart application." International Journal of Cognitive Computing in Engineering 4 (2023): 327-339.

[15] Arora, Saurabh, et al. "Prediction of credit card defaults through data analysis and machine learning techniques." Materials Today: Proceedings 51 (2022): 110-117.

[16] Jiang, Jianhua, et al. "An enhanced TSA-MLP model for identifying credit default problems." SAGE Open 12.2 (2022): 21582440221094586.

[17] Yu, K., Xia, S., Zhang, Y., & Wang, S. (2023). Loan approval prediction improved by XGBoost model based on four-vector optimization algorithm. Applied and Computational Engineering, 82, 35–44.

[18]Zhao, Q., Hao, Y., & Li, X. (2023). Stock price prediction based on hybrid CNN-LSTM model. Applied and Computational Engineering.

[19]Shiri, Farhad Mortezapour, et al. "A comprehensive overview and comparative analysis on deep learning models: CNN, RNN, LSTM, GRU." arXiv preprint arXiv:2305.17473 (2023).