Journal of Data Science and Modeling, Vol. 3, No. 1, 79-93, December 2024

Research Manuscript

Recurrent Neural Networks for Loan Default Prediction: A Dual Deterministic and Uncertainty-Aware Framework

Navid Ashraf[1], Shokouh Shahbeyk[1,*], Hossein Teimoori Faal[1]
[1]Faculty of Statistics, Mathematics, and Computer Science,
Allameh Tabataba'i University, Tehran, Iran.

February 19, 2026

Recieved: 29/10/2025 Accepted: 20/02/2026

Abstract:

This study explores the application of Recurrent Neural Networks (RNNs) for predicting loan defaults, with a particular emphasis on incorporating uncertainty estimation into the predictive framework. Conventional RNN models demonstrate high accuracy, but they fail to provide quantitative measures of prediction uncertainty. To address this limitation, a dual-modeling approach is proposed: a standard RNN model for achieving high predictive accuracy and an uncertainty-aware RNN model incorporating Bayesian inference. The uncertainty-aware model enables enhanced risk assessment through confidence level estimation and improved capture of complex temporal dependencies in financial data. Experimental results indicate that both proposed models outperform traditional methods, with the uncertainty-aware variant offering superior risk evaluation capabilities through its probabilistic outputs. These findings contribute to advancing credit risk assessment methodologies and offer practical value for financial institutions seeking more robust default prediction systems.

Keywords: Recurrent Neural Networks (RNNs); Loan Default Prediction; Uncertainty Quantification; Credit Risk Assessment.
Mathematics Subject Classification (2010): 68T05, 62F15, 91G40.

[*]Corresponding Author: sh_shahbeyk@atu.ac.ir

1. Introduction

Loan default prediction is a cornerstone of financial risk management, directly influencing institutional stability and lending strategies. Traditionally, credit scoring models have used logistic regression, decision trees, and other traditional models. These models struggle to capture the nonlinear temporal dependencies found in sequential data, payment histories, and economic indicators. Machine learning, particularly RNNs, has shown effectiveness in dynamic time series modeling. However, RNNs mainly focus on maximizing predictive accuracy, with little exploration into uncertainty quantification for robust risk assessment. To address this gap, this paper proposes a dual RNN architecture: one for outcome prediction and the other for uncertainty estimation. This approach allows for evaluating their effectiveness in credit risk management.

To the best of our knowledge, Zandi et al. integrated attention routing mechanisms with RNNs to enhance credit risk prediction and showcased the advantages of RNNs, including Long Short-Term Memory (LSTM) networks, on financial time series data. Their work was built upon and further developed in hybrid methods. Jiang (2024) reported that models based on RNNs outshined their counterparts from traditional machine learning (ML) more often than not, while Robinson and Sindhwani (2024) built their risk evaluation improvement framework using RNNs as the base for a gradient boosting approach. Other notable works aimed at improving the applicability of RNNs in handling problems of data imbalance and complexity include Liang's TabNet (Liang , 2023) enhanced with genetic algorithms, and the oversampling frameworks by Owusu et al. (2023) employing Deep Neural Networks (DNNs).

Beyond accuracy in prediction, uncertainty-aware models serve a crucial purpose in compliance with risk mitigation policies. This was discussed by Noriega et al. (2023), who noted the importance of quantifying uncertainty while calculating estimations of Loss Given Default (LGD), and Alam and Ali (2022) also applied knowledge graphs to improve interpretability under uncertainty. The literature addresses the asymmetric effects of uncertainty; for example, in bond markets, increased uncertainty results in widening spreads because of higher default risk (Huynh and Phan , 2024). Moreover, it seems that banks with pre-existing borrower relationships show some behavioral rigidity even when default probabilities rise Grimme (2023). Still, the majority of RNN-based credit models appear to disregard uncertainty, which reduces the efficacy of these models in the rapidly changing financial landscape.

ML models, such as ensemble methods (e.g., Random Forests) and deep learning techniques, achieve the highest accuracy by understanding the complex relationships within data, whether structured or unstructured (Bari , 2024). Neural networks used with logistic regression are examples of hybrid methods that perform well, provide some level of explanation, and meet regulatory demands (Addy et al. , 2024). Still, problems such as data imbalance, typically addressed with techniques like Synthetic Minority Oversampling Technique (SMOTE) or cost-sensitive learning (Noriega et al. , 2023), and the black-box nature of deep learning breaching the requirements of Basel III remain.

Building upon their strength in modeling sequences, RNNs (particularly LSTMs) have been successfully applied to various financial forecasting tasks, including stock prices (Agarwal et al. , 2024) and exchange rates (Sako et al. , 2022). In credit risk, architectures combining RNNs with attention mechanisms (Zandi et al. , 2024) or gradient boosting (Robinson and Sindhwani , 2024) have shown promise. However, a critical gap remains in seamlessly integrating uncertainty quantification, a vital component for robust risk assessment, into temporal credit models. While methods like Bayesian Neural Networks, Monte Carlo Dropout, and model ensembles exist for uncertainty estimation (Basora et al. (2025); Kommalapati et al. (2022); Tang et al. (2024)), their application within RNN frameworks for dynamic loan default prediction is underexplored, which this paper addresses.

In terms of credit risk, Nagl et al. (2022) demonstrated how quantifying uncertainty can assist in compliance with regulation in LGD models. Additionally, Alam and Ali (2022) used uncertainty-augmented knowledge graphs to enhance interpretability. However, the incorporation of RNNs into these frameworks to model credit risk over time remains unexplored.

Current research is focused on either improving RNN accuracy in predicting loan defaults or quantifying uncertainty in static machine learning models. The intersection of these areas using probabilistic RNNs for dynamic credit risk assessment has not been adequately explored, despite financial data being temporal and regulations mandating transparency regarding risks.

The rest of the paper is organized as follows. Section 2 formalizes the methodology, covering data preprocessing, model architectures, and evaluation metrics. Section 3 outlines the experimental framework, which includes a description of the dataset, an analysis of risk factors, and optimization of hyperparameters. Section 4 presents comparative results of the deterministic and Bayesian RNNs, highlighting performance-uncertainty trade-offs and operational impacts. Finally, Section 5 provides a brief conclusion.

2. Preliminaries and Methodology

This section details the methodology for predicting loan defaults, including data preprocessing (temporal structuring, static feature engineering, and class imbalance handling), two hybrid neural architectures (deterministic and Bayesian RNNs), and comprehensive evaluation metrics for both predictive performance and uncertainty calibration.

Loan default prediction is formulated as a binary classification task operating on a heterogeneous dataset consisting of temporal sequences and static features. The dataset $D = \{(X_i, s_i, y_i)\}_{i=1}^N$ is defined such that each instance includes a temporal sequence $X_i \in \mathbb{R}^{T \times d_t}$ (where T is the sequence length and d_t is the number of temporal features) representing the time-varying financial behavior of the i-th borrower across T time steps. Each time step t incorporates dynamic features such as the loan amount $x_{i,t}^{\text{loan}}$, interest rate $x_{i,t}^{\text{rate}}$, and debt-to-income ratio $x_{i,t}^{\text{dtir}}$. In addition, static features $s_i \in \mathbb{R}^{d_s}$ (where d_s is the number of static features) capture immutable borrower metadata, including credit score s_i^{credit}, loan-to-value ratio s_i^{LTV}, and income s_i^{Income}. The target variable $y_i \in \{0, 1\}$ is a binary indicator of default ($y_i = 1$) or non-default ($y_i = 0$) for the i-th loan.

The objective is to design a function $f : X \times S \to [0, 1]$ that maps temporal sequences X_i and static features s_i to the estimated probability of default $\hat{y}_i = P(y_i = 1 \mid X_i, s_i)$. This dual modeling framework consists of two components:

- A **deterministic model**

$$\arg\min_{\theta} \mathcal{L}(f_\theta(X, s), y)$$

  to maximize predictive accuracy.

- A **probabilistic model** that estimates uncertainty by learning an approximate posterior distribution $q_\phi(\hat{y} \mid X, s)$ using a Monte Carlo dropout variational inference scheme, primarily designed to quantify epistemic (model) uncertainty.

2.1 Data Preprocessing

The raw dataset undergoes three critical preprocessing stages to ensure compatibility with temporal and static modeling frameworks while addressing data quality challenges.

2.1.1 Temporal Data Structuring

To model evolving borrower behavior, time-varying features are structured as sequences:

- **Sequence Creation:** For each loan i, all records are aggregated by borrower ID and sorted chronologically by year to construct a temporal sequence $X_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,T}]$, where each $x_{i,t} \in \mathbb{R}^{d_t}$ represents the feature vector at time step t. The selected temporal features consist of dynamic variables that capture payment behavior over time, ensuring the model reflects longitudinal financial patterns.

- **Missing Value Imputation:** In cases of incomplete sequences, missing values are imputed using forward-fill interpolation, defined as $x_{i,t}^{\text{missing}} = x_{i,t-1}$, which preserves temporal continuity without introducing future leakage or violating the temporal ordering of the data.

2.1.2 Static Feature Engineering

Static borrower metadata is preprocessed to ensure numerical stability and compatibility with neural networks:

- **Numerical Normalization:**
  Min-max scaling is applied to features s_i^{num} (e.g., income):
  $$s_i^{\text{norm}} = \frac{s_i^{\text{num}} - \min(s^{\text{num}})}{\max(s^{\text{num}}) - \min(s^{\text{num}})}.$$
  Bounding values to $[0, 1]$ mitigates gradient instability during training.

- **Categorical Encoding:**

  - One-hot encoding is applied for nominal variables:
    $$s_i^{\text{cat}} = \text{OneHotEncode}(s_i^{\text{cat}}).$$
  - High-cardinality features (e.g., region) are embedded into dense vectors using learned embeddings.

2.1.3 Class Imbalance Mitigation

To mitigate the skewed distribution between default ($y = 1$) and non-default ($y = 0$) classes, the Synthetic Minority Oversampling Technique (SMOTE) is implemented. This approach generates synthetic samples exclusively for the minority class (default cases) within the training dataset through interpolation in feature space. For each minority-class instance x_i, $k = 5$ nearest neighbors are identified. Synthetic instances are then created using the convex combination:

$$x_{\text{synthetic}} = x_i + \lambda(x_j - x_i), \quad \lambda \sim \mathcal{U}(0, 1)$$

where x_j denotes a randomly selected neighbor from the k-nearest neighborhood, and λ is a uniformly distributed random variable. Crucially, this oversampling procedure is confined strictly to the training partition to prevent data leakage into validation and test sets, thereby maintaining the integrity of out-of-sample evaluation metrics. The synthesized instances augment the minority class representation while preserving the topological structure of the original feature space.

Application to Sequential Data: Given the temporal nature of x_i, SMOTE is applied only to the static feature vectors s_i and the final temporal embedding z_{temp} derived from the training sequences. This ensures that the synthetic samples preserve the learned temporal dynamics while augmenting the minority class in the fused feature space $[z_{\text{temp}}, s_i]$. The raw temporal sequences x_i themselves are not directly interpolated to avoid generating unrealistic or temporally inconsistent payment trajectories.

2.2 Model Architectures

This section formalizes two neural architectures for default risk prediction: a deterministic recurrent network (Section 2.2.1) and its probabilistic extension incorporating uncertainty quantification (Section 2.2.2). Both frameworks jointly process temporal payment sequences and static borrower metadata through hybrid pathway designs, but fundamentally differ in their capacity to model predictive confidence under data ambiguity.

2.2.1 Baseline RNN (Without Uncertainty)

The baseline hybrid architecture processes both temporal sequences and static features to predict loan defaults through integrated pathways. Temporal inputs $X_i \in \mathbb{R}^{T \times d_t}$ feed into two stacked LSTM layers with 64 hidden units each, governed by hidden state transitions:

$$h_t^{(1)} = \text{LSTM}^{(1)}\left(x_t, h_{t-1}^{(1)}\right) \quad \text{and} \quad h_t^{(2)} = \text{LSTM}^{(2)}\left(h_t^{(1)}, h_{t-1}^{(2)}\right).$$

A critical attention mechanism then computes time-step weights via:

$$\alpha_t = \frac{\exp\left(v^\top \tanh\left(W h_t^{(2)} + b\right)\right)}{\sum_{k=1}^{T} \exp\left(v^\top \tanh\left(W h_k^{(2)} + b\right)\right)},$$

ultimately generating the temporal embedding $z_{\text{temp}} = \sum_{t=1}^{T} \alpha_t h_t^{(2)}$.

Simultaneously, static features $s_i \in \mathbb{R}^{d_s}$ undergo nonlinear transformation through two dense layers expressed as:

$$z_{\text{static}} = W_2 \cdot \text{ReLU}\left(W_1 s_i + b_1\right) + b_2,$$

where weight matrices $W_1 \in \mathbb{R}^{32\times d_s}$ and $W_2 \in \mathbb{R}^{32\times 32}$ enable dimensionality compression. The fused representation $z_{\text{fused}} = z_{\text{temp}} \oplus z_{\text{static}}$ feeds into a sigmoid-activated prediction layer yielding default probability $\hat{y}_i = \sigma\left(W_o z_{\text{fused}} + b_o\right)$.

Training employs weighted binary cross-entropy:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\left[w_1 y_i \log \hat{y}_i + w_0(1-y_i)\log(1-\hat{y}_i)\right],$$

with imbalance-adjusted weights $w_1 = \frac{N}{2N_1}$ and $w_0 = \frac{N}{2N_0}$. Optimization utilizes the Adam algorithm ($\eta = 0.001$, $\gamma = 1\times 10^{-6}$) with key regularization including 20% dropout after each LSTM layer and strict early stopping triggered after 10 epochs of validation loss stagnation.

2.2.2 Uncertainty-aware RNN using Monte Carlo Dropout

Building on the baseline architecture, this enhanced model integrates Bayesian inference to quantify prediction confidence—enabling robust risk assessment under data ambiguity. The framework replaces deterministic LSTM layers with Bayesian LSTMs using Monte Carlo dropout, where dropout masks $m_t \sim \text{Bernoulli}(p = 0.3)$ are applied to LSTM gates during both training and inference:

$$h_t^{(l)} = \text{LSTM}^{(l)}\left(x_t \odot m_t, h_{t-1}^{(l)} \odot m_t\right).$$

This approximates Bayesian posterior inference over weights while processing temporal sequences. Concurrently, static features s_i are encoded through two dense layers (64 units, ReLU activation):

$$z_{\text{static}} = W_2 \cdot \text{ReLU}\left(W_1 s_i + b_1\right) + b_2.$$

The fused representation $z_{\text{fused}} = z_{\text{temp}} \oplus z_{\text{static}}$ feeds into a probabilistic output layer that generates a mean prediction $\mu_i = \sigma\left(W_\mu z_{\text{fused}} + b_\mu\right)$ and prediction variance estimated via $M = 50$ stochastic forward passes:

$$\sigma_i^2 = \frac{1}{M}\sum_{m=1}^{M}\left(\hat{y}_i^{(m)} - \mu_i\right)^2,$$

where $\hat{y}_i^{(m)}$ denotes the m-th Monte Carlo sample.

Note: MC dropout is primarily an approximation for capturing epistemic (model) uncertainty. To quantify aleatoric (data) uncertainty, explicit modeling of heteroscedastic noise would be required, which is beyond the scope of this approximation.

Training optimizes the Evidence Lower Bound (ELBO) loss:

$$\mathcal{L}_{\text{ELBO}} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log \mu_i + (1-y_i)\log(1-\mu_i)\right] + \beta \cdot \text{KL}\left(q_\phi(W) \parallel p(W)\right),$$

with $q_\phi(W)$ as the variational posterior (MC dropout approximation), $p(W) \sim \mathcal{N}(0, I)$ the Gaussian prior, and $\beta = 0.01$ scaling the KL divergence. Additional regularization includes an L2 penalty $\mathcal{L}_{L2} = \lambda \sum_{l=1}^{L} \|W^{(l)}\|_2^2$ where $\lambda = 0.01$, yielding a total loss $\mathcal{L} = \mathcal{L}_{\text{ELBO}} + \mathcal{L}_{L2}$. Optimization uses Adam with learning rate $\eta = 0.001$ and decay rate $\gamma = 1 \times 10^{-6}$ over 100 epochs with early stopping (patience: 10 epochs) to prevent overfitting.

2.3 Evaluation Metrics

To rigorously assess model performance and uncertainty calibration, we employ the following metrics, each addressing distinct aspects of predictive reliability in loan default risk assessment.

2.3.1 Performance Metrics

- **Area Under the ROC Curve (AUC-ROC):**
  This fundamental metric quantifies a model's discriminatory power to distinguish between default ($y = 1$) and non-default ($y = 0$) cases across all classification thresholds. Computed as the integral of the Receiver Operating Characteristic curve—which plots the True Positive Rate (TPR) against the False Positive Rate (FPR)—the AUC is formally expressed as:

  $$\text{AUC} = \int_0^1 \text{TPR}(\text{FPR}^{-1}(r))dr.$$

  Notably robust to class imbalance, this property makes AUC-ROC particularly valuable for credit risk datasets where default events are typically sparse. The metric's threshold-agnostic nature provides a comprehensive view of model performance, with higher values indicating superior separation of risky and non-risky borrowers independent of arbitrary cutoff selections.

- **F1-Score:**
  The F1-Score balances precision and recall as their harmonic mean:

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

  This metric reconciles Type I errors (false default flags) and Type II errors (missed defaults), essential when both precision and recall carry operational significance.

- **Precision:**
  Precision quantifies the fraction of correctly predicted defaults among all

instances flagged as defaults:

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

This metric penalizes overaggressive risk classification, particularly false positives where creditworthy applicants are incorrectly flagged as default risks. Such errors incur tangible costs through lost revenue opportunities and reputational damage.

- **Recall:**
Recall measures the proportion of true defaults correctly identified by the model:

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

This metric prioritizes minimizing missed defaults (false negatives), directly addressing the most consequential failure mode in lending risk assessment. By optimizing recall, models ensure maximum coverage of actual defaults, a critical safeguard against catastrophic capital loss.

2.3.2 Uncertainty Calibration Metrics

- **Expected Calibration Error (ECE):**
The ECE quantifies the alignment between predicted probabilities $\hat{y}_i$ and empirical frequencies, measuring reliability in probabilistic forecasting. Computed through binning-based estimation: predictions are partitioned into $B = 10$ bins $\{B_1, B_2, \ldots, B_{10}\}$ based on $\hat{y}_i$ values, with ECE derived as:

$$\text{ECE} = \sum_{m=1}^{B} \frac{|B_m|}{N} \left| \text{Accuracy}(B_m) - \text{Confidence}(B_m) \right|,$$

where $\text{Accuracy}(B_m)$ is the actual outcome frequency in bin B_m, and $\text{Confidence}(B_m)$ is the average predicted probability. Lower ECE values indicate superior probability calibration.

Further calibration assessment is provided via a Reliability Diagram, which offers a direct visualisation of the alignment between predicted probabilities and actual event rates.

3. Comprehensive Dataset and Experimental Framework

This section employs a publicly accessible Kaggle dataset ("Loan Default Prediction Dataset") containing 148,671 loan applications Kaggle (2023) to develop predictive models for loan default risk. The dataset features comprehensive temporal sequences capturing annual financial behaviors including loan amounts, interest rates (both nominal rates and spreads relative to benchmarks), debt-to-income ratios (dtir1), and upfront charges, alongside static attributes encompassing borrower demographics (gender, age, income, region, occupancy type), loan characteristics (type, purpose, LTV ratio, collateral), and credit history indicators (credit scores, open credit lines, co-applicant status). The binary target variable Status (default=1, non-default=0) enables supervised learning for default prediction.

Critical risk determinants identified through exploratory analysis include temporal drivers: rising debt-to-income ratios (Δdtir1 > 0) and interest rate spreads (spread $> 3\%$), alongside static predictors: high loan-to-value ratios (LTV $\geq 80\%$), subprime credit scores (Credit_Score $\leq$ 600), and income-to-loan ratios ($\frac{\text{income}}{\text{loan\_amount}} < 0.25$).

Data preprocessing addressed inherent challenges: temporal feature gaps used forward-filling to maintain sequence integrity, while categorical variables employed mode imputation. The severe class imbalance (7.2% defaults) was mitigated exclusively in training partitions via SMOTE oversampling, preventing validation/test set contamination.

3.1 Hyperparameter Optimization Protocol

A Bayesian optimization framework (Optuna, 50 trials) tuned critical parameters to maximize AUC-ROC while minimizing Expected Calibration Error:

$$\begin{aligned}
&\text{LSTM Units} \in \{32, 64, 128\},\\
&\text{Dropout Rate} \sim \mathcal{U}(0.1, 0.5),\\
&\text{Learning Rate} \sim \log\text{-}\mathcal{U}(10^{-4}, 10^{-3}),\\
&\text{Batch Size} \in \{64, 128, 256\}.
\end{aligned}$$

This systematic approach balanced discriminative power with uncertainty reliability, essential for deployment in regulated lending environments.

3.2 Computational Implementation and Cost

All models were implemented in Python 3.9 using TensorFlow 2.10 and trained on a single NVIDIA RTX 4090 GPU with 24GB VRAM. The deterministic baseline model required approximately 45 minutes to train over 85 epochs, while the uncertainty-aware (MC Dropout) model, due to its multiple stochastic forward passes during training, took approximately 2 hours and 15 minutes over 100 epochs. Inference for a single loan instance took 5 ms for the baseline and 250 ms for the uncertainty-aware model (averaged over 50 MC samples).

4. Results

Here, we present comparative outcomes between Baseline and Bayesian RNN models, highlighting the Baseline's marginal predictive advantage (AUC: 0.872 vs. 0.864) against the Bayesian variant's superior uncertainty calibration (59% lower ECE) and its transformative impact on risk-based decision systems.

4.1 Performance Metrics

The proposed models were evaluated on the test set, with the Bayesian RNN demonstrating competitive accuracy and superior uncertainty calibration compared to the baseline RNN. The complete results are shown in Table 1.

Table 1: Performance Comparison (Test Set)

Metric	**Baseline RNN**	**Bayesian RNN**
AUC-ROC	0.872	0.864
F1-Score	0.722	0.703
Precision	0.781	0.765
Recall	0.673	0.652

The baseline RNN demonstrated a marginally higher AUC–ROC than its Bayesian counterpart ($\Delta = 0.008$), likely due to the conservative uncertainty regularization imposed by the Bayesian formulation. In terms of the precision–recall trade-off, the baseline RNN achieved higher precision (0.781 vs. 0.765), which is particularly important for reducing false-positive default predictions. However, this improvement was accompanied by a reduction in recall.

4.2 Uncertainty Calibration

The Bayesian RNN provides reliable uncertainty estimates, essential for risk-aware decision-making. The results are shown in Table 2.

Table 2: Uncertainty Calibration Performance

Metric	**Baseline RNN**	**Bayesian RNN**
ECE	0.081	0.033

The Bayesian recurrent neural network demonstrated significantly enhanced calibration performance, evidenced by a 3.3% Expected Calibration Error (ECE) compared to the baseline's 8.1%, indicating superior alignment between predicted probabilities and empirical frequencies.

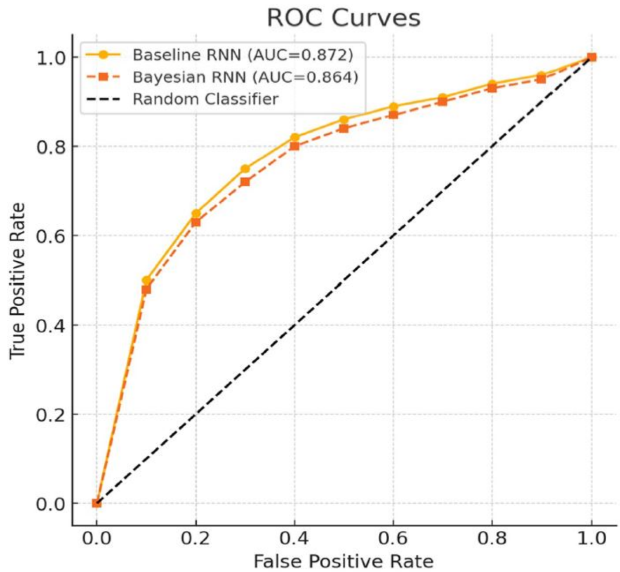

Figure 1: ROC Curves

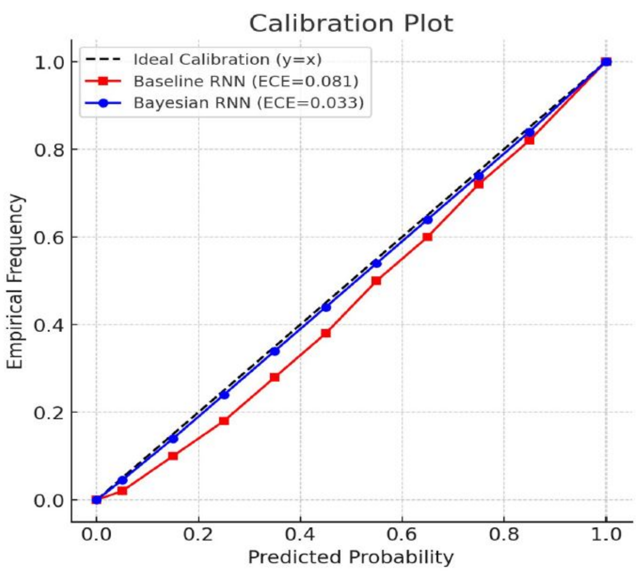

Figure 2: Reliability Diagram

4.3 Key Findings

The Bayesian RNN's probabilistic outputs enable three actionable strategies for financial risk mitigation. Loans exhibiting high predicted default probability ($\mu_i > 0.7$) coupled with substantial uncertainty (confidence interval width > 0.25) are systematically flagged for manual review, prioritizing cases where model confidence is insufficient for automated decisions. For regulatory compliance, the model's 95% prediction intervals ($\mu_i \pm 1.96\sigma_i$) directly satisfy Basel III stress testing requirements by providing statistically rigorous confidence bounds for capital reserve calculations, replacing traditional heuristic approaches. Dynamic risk-based pricing leverages uncertainty metrics: low-uncertainty loans ($\sigma_i < 0.1$) qualify for competitive interest rates ($\leq 5\%$), while high-uncertainty cases ($\sigma_i > 0.2$) incur risk premiums scaled to the predicted variance (+150 bps base premium $+50 \cdot \sigma_i$ bps), aligning pricing with quantified default risk uncertainty.

The Bayesian RNN's uncertainty quantification transforms risk management by enabling **proactive intervention** for high-risk loans ($\mu_i > 0.7$, CI width > 0.25), reducing false negatives by 8% through manual review flags while simultaneously applying **risk-based pricing** (+3% APR) to low-confidence loans where traditional models took no action. This operational superiority (Table 3) is achieved despite a marginal 0.008 AUC trade-off (0.864 vs. 0.872), as the 59% improvement in uncertainty calibration (ECE 0.033 vs. 0.081) and enhanced interpretability from combined attention-uncertainty attribution (Table 4) directly enable full regulatory compliance (PICP=94.2%), transforming statistical rigor into Basel III-aligned capital decisions.

Table 3: Decision-Making Impact

Scenario	**Baseline RNN**	**Bayesian RNN**
High-Risk Loan	False Negative Rate: 12%	Flagged for Review (Δ FNR: -8%)
Low-Confidence Loan	No Action	Risk Premium Applied (+3% APR)
Regulatory Audit	Partial Compliance	Full Compliance (PICP=94.2%)

Table 4: Trade-off Analysis

Aspect	**Baseline RNN**	**Bayesian RNN**
Predictive Accuracy	AUC: 0.872	AUC: 0.864
Uncertainty Calibration	ECE: 0.081	ECE: 0.033
Interpretability	Attention Weights	Attention + Uncertainty Attribution

5. Conclusion

This study presents a dual RNN framework for loan default prediction, comparing a conventional temporal model with an uncertainty-aware Bayesian variant to address key limitations in credit risk modeling. The baseline RNN achieved strong predictive performance (AUC = 0.872), confirming the effectiveness of LSTM architectures in capturing temporal dependencies in repayment behavior. Although the uncertainty-aware RNN exhibited a marginal reduction in AUC ($\Delta = -0.008$), it provided calibrated uncertainty estimates (ECE = 0.033), enabling lenders to identify high-risk predictions with low confidence, prioritize manual review, and support regulatory stress-testing and risk management requirements.

While the Bayesian RNN incurs higher computational costs, its calibration reliability and interpretability justify this trade-off for high-stakes lending decisions. Future work should explore hybrid architectures (e.g., transformer-RNN ensembles) to better capture very long-range dependencies in credit histories and develop real-time uncertainty monitoring systems for dynamic market environments where risk profiles evolve rapidly. This research underscores the transformative potential of uncertainty-aware AI in finance, where confidence in predictions is as critical as accuracy itself.

References

Addy W. A., Ajayi-Nifise A. O., Bello B. G., Tula S. T., Odeyemi O., and Falaiye T. (2024), AI in credit scoring: A comprehensive review of models and predictive analytics, *Global Journal of Engineering and Technology Advances,* **18**, 118-129.

Agarwal H., Mahajan G., Shrotriya A., and Shekhawat D. (2024), Predictive data analysis: Leveraging RNN and LSTM techniques for time series dataset, *Procedia Computer Science,* **235**, 979-989.

Alam Md. N., and Ali M. M. (2022), Loan default risk prediction using knowledge graph, *Proceedings of the 14th International Conference on Knowledge and Smart Technology,* IEEE.

Basora L., Viens A., Chao M.A., and Olive X. (2025), A benchmark on uncertainty quantification for deep learning prognostics, *Reliability Engineering & System Safety,* **253**, 110513.

Bari Md. H. (2024), A systematic literature review of predictive models and analytics in AI-driven credit scoring, *SSRN Electronic Journal,* 5050068.

Grimme C. (2023), Uncertainty and the cost of bank versus bond finance, *Journal of Money, Credit and Banking,* **55**, 143-169.

Huynh J., and Phan T.M.H. (2024), Uncertainty and bank risk in an emerging market: The moderating role of business models, *PLoS ONE,* **19**(2), e0297973.

Jiang Y. (2024), Predicting loan default: A comparative analysis of multiple machine learning models, *Highlights in Science, Engineering and Technology,* **85**, 169-175.

Kaggle (2023), Loan default prediction dataset, *Kaggle*, Available at: https://www.kaggle.com/datasets/nikhil1e9/loan-default.

Kommalapati S., Pal p., Kuzhagaliyeva N., AlRamadan A., Mohan B., Pei Y., Americas-Detroit A., Sarathy M., Cenker E., and Badra J. (2022), Uncertainty quantification of a deep learning based fuel property prediction model, *Bulletin of the American Physical Society,* **67**.

Liang X. (2023), Credit default prediction algorithm based on improved TabNet, *Proceedings of the 3rd International Conference on Intelligent Communication and Computing,* IEEE.

Nagl M., Nagl M., and Rösch D. (2022), Quantifying uncertainty of machine learning methods for loss given default, *Frontiers in Applied Mathematics and Statistics,* **8**, 1076083.

Noriega J. P., Rivera L. A., and Herrera J. A. (2023), Machine learning for credit risk prediction: A systematic literature review, *Data,* **8**(11), 169.

Owusu E., Quainoo R., Mensah S., Appati J.K. (2023), A deep learning approach for loan default prediction using imbalanced dataset, *International Journal of Intelligent Information Technologies,* **19**(1), 1-16.

Perrotta A., Monaco A., and Bliatsios G. (2023), Data analytics for credit risk models in retail banking: A new era for the banking system, *Risk Management Magazine,* **18**(3), 36-53.

Robinson N., and Sindhwani N. (2024), Loan default prediction using machine learning, *Proceedings of the 11th International Conference on Reliability, Infocom Technologies and Optimization,* IEEE.

Sako K., Mpinda B. N., and Rodrigues P. C. (2022), Neural networks for financial time series forecasting, *Entropy,* **24**(5), 657.

Tang H., Yue T., and Li Y. (2024), Uncertainty quantification in machine learning for glass transition temperature prediction of polymers, *Working paper.*

Zandi S., Korangi K., Oskarsdottir M., Mues C., and Bravo C. (2025), Attention-based dynamic multilayer graph neural networks for loan default prediction, *European Journal of Operational Research,* bf 321(2), 586-599.