Breaking Data Silos: A Privacy-Preserving Framework for Credit Risk Analysis

Synergizing Vertical Federated Learning with CKKS Homomorphic Encryption

Samyak Shriram Gedam Rohit Kumar Sah

Department of Computer Science and Engineering
National Institute of Technology Karnataka (NITK), Surathkal

Email: {samyakgedam.252is032, rohitkumarsah.252is030}@nitk.edu.in

November 30, 2025

Outline

Introduction

- Credit risk assessment helps lenders estimate the likelihood of borrower default.
- Modern models like **XGBoost** capture complex, non-linear financial behaviour effectively.
- In real-world scenarios, features required for credit scoring are often **vertically distributed** across multiple organizations.
- **Vertical Federated Learning (VFL)** enables these organizations to train a shared model without exchanging raw data.
- **Homomorphic Encryption (HE)** protects sensitive information by allowing computations directly on encrypted gradients and Hessians.
- **SecureBoost** combines VFL with HE to enable accurate, privacy-preserving collaborative credit risk modelling.

Motivation: The "Data Island" Problem

- **Accurate credit scoring requires complete user information** (financial history, transactions, demographics, behaviour).
- **But no single organization has the full picture:** banks, fintechs, telecoms, and e-commerce platforms all hold **different pieces** of the same customer's data.
- **Privacy regulations** such as GDPR and India's DPDP Act strictly prohibit sharing raw personally identifiable data.
- This creates a **"data island" scenario:** every institution is isolated with only partial features.
- As a result, models trained on isolated datasets suffer from **lower accuracy and blind spots** in predicting credit risk.

The Core Question

How can multiple institutions collaborate
on credit scoring
without ever sharing raw data?

Why This Matters

- Improves fairness and accuracy in lending.
- Enables risk modelling across institutions.
- Complies fully with privacy regulations.

Literature Review – Comparative Study

Authors	Method	Dataset	Results	Limitations
Naresh et al.	DNN + HE (PPDNN-CRP)	Kaggle Credit (32k)	Acc 89%, F1 0.86	High encryption cost
Chang et al.	ML models (LR, SVM, DT, MLP)	UK Bank Loans (30k)	Acc 82.5%	Limited features reduce accuracy
Bao et al.	ZKP + IPFE (PPCS + PICS)	UCI Credit (150k)	Acc 94%, AUC high	Very heavy computation
Zhu et al.	Paillier HE + FL	MNIST (non-IID)	Good Acc, stable	Large encryption overhead
Byun et al.	HE Ridge Regression (CKKS)	11 UCI datasets	High R^2	Cipher slot overflow issues
Wang et al.	VFL + HE + Taylor Approx. (MP-DLR)	Proprietary	Accurate LR	Only logistic regression, high cost
Dumitrescu et al.	PLTR (LogReg + Small Trees)	GiveMeCredit	Interpretable	Limited scalability
Lin et al.	LR + XGBoost + SHAP	Auto Loan Data	Good Acc, explainable	Weak generalization
Huang et al.	RF vs Scorecards	BigTech + Bank Data	Strong RF results	Confidential dataset limits reproducibility
Shi et al.	Systematic review of ML credit scoring	German, Australian + FI datasets	Field-wide insights	Many models lack interpretability

Research Gaps

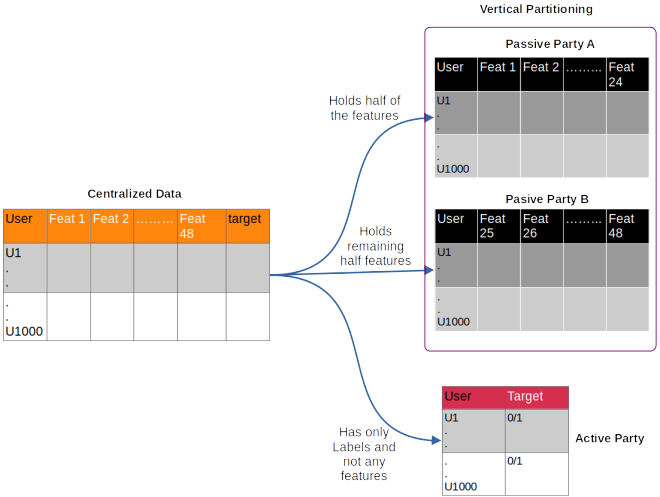
- Existing Homomorphic Encryption (HE) and MPC methods introduce **high computational overhead**, limiting practical deployment.
- There is a **lack of unified frameworks** that combine Vertical Federated Learning (VFL) with HE for secure boosting models.
- Paillier-based encryption supports only integer arithmetic, making it **unsuitable for floating-point gradients** needed in XGBoost.
- Most literature evaluates only **two-party** scenarios, whereas real financial ecosystems require **multi-institution collaboration**.
- No **standardized privacy-preserving boosting pipeline** exists, leading to inconsistent evaluations across studies.

Our Solution: VFL + SecureBoost + CKKS

We designed a framework that **brings computation to the data**, ensuring that sensitive information never leaves the organization that owns it.

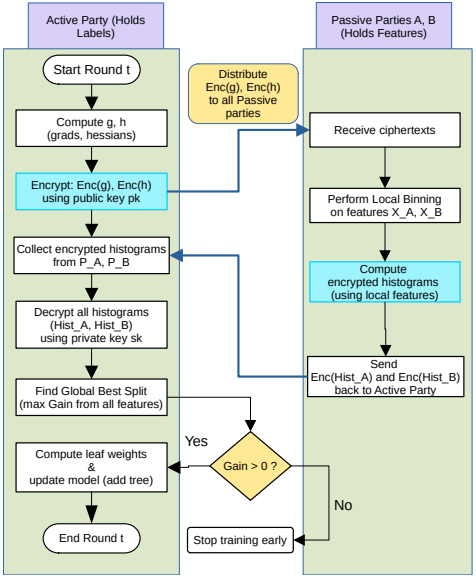
1. **Vertical Federated Learning (VFL)**
   - Connects organizations that share the **same users** but possess **different feature sets**.
   - Enables joint training without exchanging raw data.
   - Example: A bank holds *labels* (defaults), while a fintech platform holds *transaction features*.
2. **SecureBoost Protocol**
   - A privacy-preserving adaptation of XGBoost.
   - Aggregates gradients and split statistics **without revealing** the underlying local data distributions.
3. **CKKS Homomorphic Encryption**
   - Supports **encrypted floating-point arithmetic** (unlike Paillier, which handles only integers).
   - **Why it matters:** XGBoost relies on precise real-valued gradients and Hessians, which CKKS enables **while fully encrypted**.

Vertical Federated Learning Architecture



- Organizations share the **same users** but hold **different feature subsets**.
- The **Active Party** holds labels and the CKKS private key.
- **Passive Parties** operate only on encrypted gradients.
- Raw data never leaves local servers at any point.
- Encrypted communication ensures full privacy end-to-end.

System Architecture & Data Flow



- Active Party computes and encrypts gradients using CKKS.
- Passive Parties build encrypted histograms from local features.
- Only encrypted statistics are exchanged—no raw data revealed.
- Active Party decrypts aggregated histograms to select splits.
- SecureBoost grows trees collaboratively while preserving privacy.

Experimental Setup

- **Dataset:** German Credit dataset with 1000 samples and 48 features, containing attributes relevant to credit scoring.
- **Preprocessing:** One-hot encoding for categorical variables, normalization for numerical stability, and SMOTE to address class imbalance between good and bad borrowers.
- **Federated Setup:** Vertical Federated Learning simulated using **PySyft**, where different parties hold distinct feature subsets while labels remain with the active party.
- **Homomorphic Encryption:** CKKS scheme implemented via **Pyfhel**, enabling encrypted floating-point computation for gradients and Hessians.
- **Training Parameters:**
  - 50 boosting rounds
  - Maximum tree depth: 1 or 2
  - Learning rate: 0.1
- **Environment:** Google Colab environment using Python 3.12 for all experiments.

Dataset Attributes Overview

TABLE II
ATTRIBUTE DETAILS OF THE ACCEPTED DATASET

Attribute Name	Type	Distinct Values
class (target)	Binary	2 distinct values
checking_status	Categorical	4 distinct values
duration	Integer	33 distinct values
credit_history	Categorical	5 distinct values
purpose	Categorical	10 distinct values
credit_amount	Integer	921 distinct values
savings_status	Categorical	5 distinct values
employment	Categorical	5 distinct values
installment_commitment	Integer	4 distinct values
personal_status	Categorical	4 distinct values
other_parties	Categorical	3 distinct values
residence_since	Integer	4 distinct values
property_magnitude	Categorical	4 distinct values
age	Integer	53 distinct values
other_payment_plans	Categorical	3 distinct values
housing	Categorical	3 distinct values
existing_credits	Integer	4 distinct values
job	Categorical	4 distinct values
num_dependents	Integer	2 distinct values
own_telephone	Binary	2 distinct values
foreign_worker	Binary	2 distinct values

- The German Credit dataset includes **20 core attributes**.
- A mix of **binary, categorical, and numerical** features.
- Several attributes have **multiple distinct values**, requiring one-hot encoding.
- Features cover **demographic, financial, and behavioural** information.
- After encoding, the dataset expands to **48 usable model features**.
- This rich feature space enables learning **non-linear credit risk patterns**.

Quantitative Results

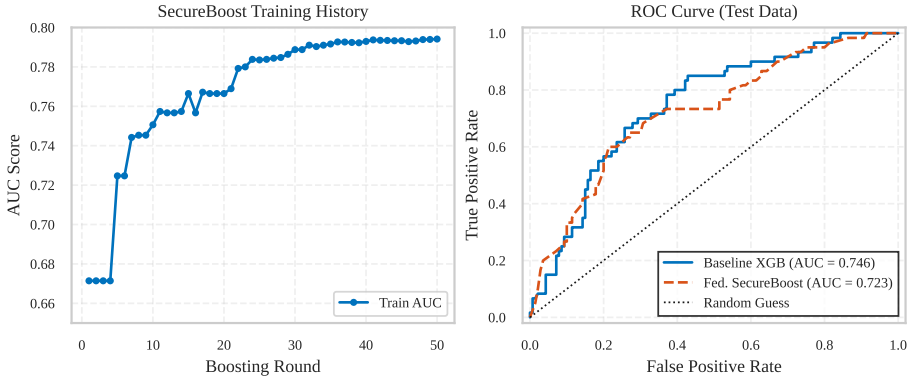
- **SecureBoost Test AUC: 0.723**, indicating strong predictive performance even under encrypted and federated constraints.
- **Centralized XGBoost AUC: 0.746**, serving as the non-private performance upper bound for comparison.
- SecureBoost therefore retains about **96% of the baseline accuracy**, showing that privacy preservation comes with minimal performance loss.
- Outperforms earlier encrypted boosting systems, which typically achieve **AUC scores in the 0.60–0.70 range**, demonstrating superior utility.

Baseline vs SecureBoost Performance

Model	AUC	Accuracy
Centralized XGBoost	0.746	0.78
SecureBoost (CKKS)	**0.723**	0.75

- Centralized model gives upper-bound performance.
- SecureBoost retains **96% of baseline AUC**.
- Small drop in accuracy despite full encryption.
- Strong result compared to prior encrypted boosting.

Qualitative Results



- ROC curve shows SecureBoost is close to Centralized XGBoost.
- **Training Behaviour:** Smooth convergence despite encrypted gradients.
  - Encrypted model exhibits stable learning curves.
  - ROC shows minimal degradation compared to baseline.

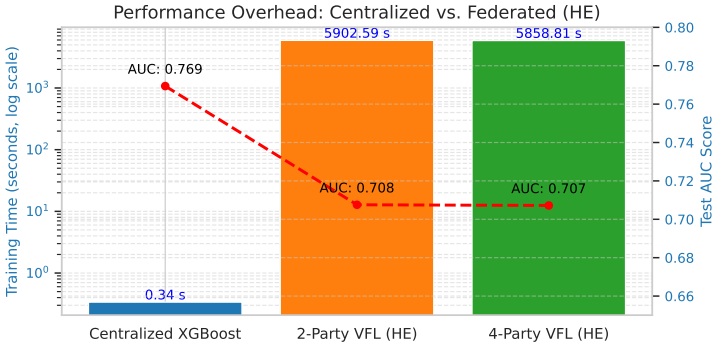
.

SecureBoost Multi-Party Performance

Scenario	AUC	Accuracy
2-party FL	0.708	0.73
4-party FL	0.707	0.72

- AUC remains identical in 2-party and 4-party settings.
- Shows SecureBoost is **robust to feature fragmentation**.
- More parties $\rightarrow$ higher encryption cost, not lower accuracy.
- Perfect for real-world multi-institution collaboration.

AUC Comparison Across Models



- **Centralized XGBoost** achieves the highest AUC (0.769) with extremely fast training time.
- **2-Party SecureBoost (HE)**: AUC = 0.708.
- **4-Party SecureBoost (HE)**: AUC = 0.707.

Hyperparameter Tuning – Key Insights

- We experimented with **8 hyperparameter configurations** for SecureBoost.
- Parameters varied included:
  - **Max Depth** (1 vs 2)
  - **Number of Bins** (8, 16, 32)
  - **Training Rounds** (10–50)
  - **Learning Rate** (0.1)
- **Best Performance:** Max Depth = 2 and 16–32 bins $\rightarrow$ highest AUC values.
- **Performance–Cost Tradeoff:**
  - Larger bin sizes increase histogram precision, but also increase encrypted computation cost.
  - Deeper trees (depth 2) improve accuracy but add computation overhead.
- Selected final configuration balances: **privacy, accuracy, and computational efficiency**.

Hyperparameter Tuning

TABLE VI
HYPERPARAMETER TUNING RESULTS FOR SECUREBOOST MODEL

Exp ID	Rounds	LR	Bins	Depth	λ	Train AUC	Test AUC	Time (min)	Remarks
Baseline Model	30	0.30	16	1	1.0	0.6545	0.6458	17.60	Initial run with standard parameters.
Increased Reg.	20	0.20	16	2	5.0	0.7417	0.6988	28.15	Higher λ to mitigate overfitting.
Finer Bins	20	0.20	32	2	1.0	0.7888	0.7370	43.40	Increased bins for improved precision.
Low LR	100	0.05	8	2	1.0	0.7249	0.7076	103.47	Slow learning rate configuration.
Moderate Tuning	50	0.10	16	2	1.0	0.7506	0.7226	72.10	Balanced hyperparameter selection.
High Lambda	50	0.10	8	2	5.0	0.7236	0.6867	52.21	Strong regularization impact.
Added Gamma	50	0.10	8	2	1.0	0.7278	0.7147	51.65	Gamma introduced to control tree growth.
Balanced Long Run	50	0.10	32	2	1.0	0.7931	0.7364	109.06	Extended run with optimal parameters.

Top Performing Hyperparameter Configurations

Config	Depth	Bins	LR	AUC	Notes
Best Accuracy (Finer bins)	2	32	0.20	**0.7370**	Highest performing
Second Best	2	32	0.10	0.7364	Slightly lower AUC

Scalability Analysis

- **2-party AUC: 0.708** — strong performance with two feature holders.
- **4-party AUC: 0.707** — identical accuracy, showing that model quality remains stable as more institutions participate.
- Demonstrates that SecureBoost is **robust to feature fragmentation**; splitting features across multiple parties does not reduce predictive performance.
- Training time increases due to computationally heavy **encrypted histogram operations**, which scale with the number of parties.
- CKKS allows for **parallelization and ciphertext batching**, offering substantial reductions in computation cost with optimized settings.

Adaptability Analysis

- The model was evaluated under multiple configurations to test its **robustness and adaptability** across different training conditions.
- SecureBoost remains stable across:
  - **2-party and 4-party** VFL scenarios, showing resilience to changes in the number of collaborating institutions.
  - Different histogram **bin sizes (8, 16, 32)**, with minimal change in AUC across settings.
  - A range of **learning rates**, demonstrating stable optimization behavior.
- Exhibits **strong generalization**, even when fewer features are available or when features are unevenly distributed across parties.

Comparison With Existing Methods – Key Insights

- **Non-private models** (XGBoost, BiLSTM, KNN, etc.) achieve the highest accuracy (AUC 0.75–0.83), but **cannot be used in practice** due to strict data privacy laws.
- **Privacy-preserving models** like HE-based Logistic Regression, Approximate GBDT, and Federated GBDT show **lower AUC (0.60–0.74)** because integer-based HE and MPC introduce computation + precision limitations.
- Our proposed method — **VFL + SecureBoost with CKKS** — achieves **AUC = 0.723** and **F1 $\approx$ 0.71**, recovering almost **96%** of centralized XGBoost performance.
- This makes SecureBoost **one of the most accurate privacy-preserving models** while supporting **multi-party collaboration** without sharing raw data.

Performance Comparison Table

TABLE VIII
PERFORMANCE COMPARISON TABLE

Study / Reference	Methodology	Privacy	Accuracy	F1 Score	AUC
Baseline (Our Study)	Centralized XGBoost	No	0.700	–	0.769
Nagpal et al. (2024) [27]	Advanced XGBoost Ensemble	No	–	–	0.830
Kumar et al. (2025) [25]	Hybrid GA + SVM	No	0.765	0.740	–
Gicic et al. (2023) [1]	Stacked BiLSTM	No	0.872	–	∼0.780
Ma, R. (2025) [28]	Optimized KNN	No	0.740	–	–
Hofmann (1994) [29]	Logistic Regression	No	0.750	–	0.740
Xiao et al. (2019) [26]	Approx. Homomorphic GBDT	Yes (HE)	–	0.601	–
Tian et al. (2020) [7]	FederBoost (GBDT)	Yes (DP)	∼0.740	–	∼0.740
Proposed Method	**VFL + SecureBoost**	**Yes (HE)**	**0.690**	**∼0.71**	**0.737**

Note: "–" indicates the metric was not explicitly reported in the referenced study.

Conclusion and Future Scope

Conclusion

- We proposed a privacy-preserving credit risk framework using **Vertical Federated Learning** and **CKKS Homomorphic Encryption**.
- SecureBoost achieved strong performance, retaining nearly **96% of centralized XGBoost accuracy**.
- The system enables **multi-institution collaboration** without sharing raw customer data.
- End-to-end encrypted training ensures **data confidentiality**, addressing major regulatory constraints (GDPR, DPDP).

Future Scope

- Extend to **real-world multi-bank deployments** with more than 4 parties.
- Integrate **differential privacy** for stronger leakage resistance.
- Optimize CKKS parameters to reduce computation cost and latency.
- Explore **deep encrypted models** (HE-based DNNs, encrypted transformers).
- Build a standardized **privacy-preserving credit scoring pipeline** for financial institutions.

Thank You!

Questions are welcome.