Grenze International Journal of Engineering and Technology, June Issue

Scalable Backend Architecture with Hybrid Machine Learning for MSME Operations Optimization: Demand Forecasting and Credit Risk Assessment

Akshit Kaushik[1], Akshay Bansal[2] and Akhya Rastogi[3]
[1-3]Dept. of CSE (AIML), ABES Engineering College, Ghaziabad, U.P., India
Email: akshit.22b1531050@abes.ac.in, akshay.22b1531026@abes.ac.in, akhya.22b1531179@abes.ac.in

***Abstract*— India's micro, small and medium enterprises (MSMEs) contribute close to 30% of national GDP which is very significant, yet the vast majority still uses manual and inefficient paper ledger method to maintain their data. Because the margins are low in the MSMEs, the process of maintaining stock replenishment and credit record maintenance becomes crucial and due to the traditional methods these two activities are prone to inaccuracies. We present a cloud deployable intelligent platform designed for this context, using a Node.js / Express.js REST backend and MongoDB document database store with two dedicated machine learning models running on flask server. The first service is a two stage demand forecasting engine in which Facebook's Prophet decomposes trend and multi period seasonality, with an XGBoost regressor that works on the systematic residuals left by that decomposition. It was tested on five stores from the Rossmann Store Sales benchmark, this hybrid model achieved a mean SMAPE based accuracy of 86.79% (peak 95.88%), beating Naive, ARIMA, Seasonal Naive and standalone Prophet; Diebold-Mariano tests confirm significance at $p < 0.01$. The second model is a credit default classifier that works on a schema built dedicatedly for the Indian udhaar lending culture, where festival period and salary week are key indicators maintained in every record by an automated ETL (Extract Transform Load) layer. Evaluated for 28,638 real loans from the Kaggle Credit Risk benchmark, XGBoost achieved ROC-AUC=0.8672 and a cross validated ROC-AUC score of 0.9431 ±0.0034, performing better than Logistic Regression, Decision Tree and Random Forest (McNemar test, $p < 0.05$).**

***Index Terms*— MSMEs; demand forecasting; Prophet-XGBoost hybrid; credit risk assessment; informal credit; ETL pipeline; festival period signals; salary cycle; scalable backend.**

I. Introduction

When we walk into any Indian colony or market place, it is almost certain we will find a kirana store. India has over 13 million such stores and more than 63 million MSMEs overall. Retail of this kind makes for almost 75% of all consumer transactions, serving millions of households that formal supermarket chains found difficult reaching [1]. Stock levels are maintained by memory or paper ledger, customer credit is written using pen on ruled paper, and reorder decisions are made entirely by instincts, making process prone to inaccuracy. This inaccuracy directly causes financial risk. When a shopkeeper overstocks a product because last Diwali was unusually busy, the loss will be certain and immediate.

Grenze ID: 01.GIJET.12.2.395_24

Studies estimate 15–25% of informal credit is never recovered [2] that accounts for loss. For a shop on margins below five percent, losing even two or three percent to the bad credit wipes out the profit of the month. Existing tools like Khatabook and OkCredit digitise the notebook but are not intelligent and do not predict anything [3]. Prior literature validated machine learning for organised retail demand [4] and institutional credit scoring [5], but it does not target the informal Indian shops where festival calendars and salary cycles produce patterns absent from Western datasets, making them key features for consideration [6]. This paper addresses these gaps with three contributions:

- A production grade backend whose ETL controllers maintains festival period, salary week and seasonal context automatically at every sale and transaction that requires no manual input from the business owner.
- AProphet+XGBoost two stage forecasting pipeline achieving 86.79% mean accuracy for five stores with Diebold Mariano confirmed significance [7], [8].
- An XGBoost credit default classifier on an informal credit schema, reaching ROC-AUC=0.8672 despite cold start absence of Indian retail default labels [10].

II. Literature Survey

ARIMA [12] is the baseline for univariate forecasting but it struggles with the multiple overlapping seasonality and holiday spikes. Prophet [7] worked and solved this with an additive decomposition of trend, Fourier seasonality and holidays, which consistently outperforming statistical methods on an event driven retail data [4]. XGBoost [8] works and dominates on tabular forecasting when lag and rolling features are engineered, and the M5 competition [11] confirmed that the hybrid decomposition plus correction architectures are state of the art. What published hybrids lack is the first class treatment of dedicated Indian specific signals: festival month surges and the salary-week spending bump. On the credit side, formal lending report ROC AUC score of 0.78–0.84 for XGBoost and Random Forest [5]. SMOTE [10] is standard for this class imbalance. Udhaar shared none of the properties of the formal credit, no bureau record, no collateral, often no written amount. No prior system maintains and capture udhaar signals at origination and delivers them inside a deployable retail platform, which is the gap this work closes on. Recent works confirms that the decomposition and correction hybrids performs better than single model on high-variance retail-data [11], and the tree based residual correction over statistical decompositions achieves high accuracy [15], directly supporting Prophet+XGboost design. For informal credits, the transaction derived behavioral features performs better than bureau-based scores for thin-file borrowers [16], while temprory signals like seasonal cash-flow cycles are amongst strongest default predictors in MSME contexts [17], yet no deployable platform captures these signals at origin, which is the gap that is addressed by this work.

III. System Architecture

The platform consists of three layers. The data layer has a Node.js/Express.js REST API with MongoDB. Document storage was chosen because the product documents and credit ledgers vary substantially between the retailers and business owner. The ETL layer triggers at every sale and every pay ment event. At interception it computes all temporal and contextual features and writes these records to the two ML collections ProductSaleTransaction and CreditTransaction before returning the API response. The critical design choice is that festivalPeriod and salaryWeek are evaluated at write time, not retrospectively: While the current deployment resides on single cloud virtual machine, the use of mongoDB's document-based store allows us to perform future horizontal scaling through sharding of the Database. Additionally, the flask-based intelligence layer is architected as a set of stateless microservices, allowing the process of independent containerization scaling of the Prophet and XGBoost to handle the peak transaction efficiently. The intelligence layer hosts two Flask microservices. The Node.js core calls them over HTTP connection with a 60 second timeout to handle Prophet's inference latency or timeout. Figure 1 maps the proposed schema to the Rossmann benchmark (73% direct or proxy compatibility); Figure 2 shows feature coverage and usage breakdown.

IV. Demand Forecasting

A. Two-Stage Hybrid

A Two-Stage Hybrid Stage 1 starts with Prophet [7] because it is good for the kind of patterns retail data generally has — weekly dip on Sundays. a spike on Diwali or a slow upward trend over months. We set it under multiplicative mode to handle the festival spike. Stage 2 employs XGBoost [8] which improves on Prophet.

Prophet mishandles combinations, so we train XGBoost not on sales directly but on the Prophet errors too (n=400, depth=4, lr =0.04). The features fall into four natural sections:

- How much was the sale recently — lag features going back 1, 7, 14 and 30 days.
- How stable was the demand — rolling mean and standard deviation over the same windows.
- Where we are in the calendar — day of week, month, quarter, week number, day index.

Feature mapping: our system schema vs. Rossmann dataset
Overall compatibility: 79% (5 direct, 4 proxy, 2 engineered, 3 missing)

Our Schema Field	Rossmann Equivalent	Used in Model	Mapping Type	Purpose
quantitySold	Sales	Yes	Direct	Primary target variable
festivalPeriod	StateHoliday	Yes	Direct	Public holiday demand spike
salaryWeek	Promo	Yes	Proxy	Demand booster signal
isWeekend	DayOfWeek (>=6)	Yes	Direct	Weekend demand pattern
season	Derived from Date	Yes	Engineered	Seasonal context
weekOfMonth	Derived from Date	Yes	Engineered	Intra-month demand cycle
month	Derived from Date	Yes	Direct	Monthly seasonality
dayOfWeek	DayOfWeek	Yes	Direct	Weekly seasonality
stockBeforeSale	Not Available	No	Missing	Inventory context
stockAfterSale	— Not Available —	No	Missing	Post-sale stock level
totalSaleAmount	Sales x Price	Partial	Proxy	Revenue proxy
unitPrice	— Not Available —	No	Missing	Per-unit price
productCode	Store (filtered)	Partial	Proxy	Single product – single store
OrganisationCode	Store	Partial	Proxy	Business entity -> store ID
N/A	SchoolHoliday	Yes	Rossmann Extra	School holiday demand dip
— N/A —	StoreType	Yes	Rossmann Extra	Store format signal
N/A	Assortment	Yes	Rossmann Extra	Product range signal

Direct match (light gray) | Engineered from date | Not available in Rossmann (dark gray)
Proxy match | Rossmann bonus feature

Fig. 1 Schema mapping: proposed fields vs. Rossmann

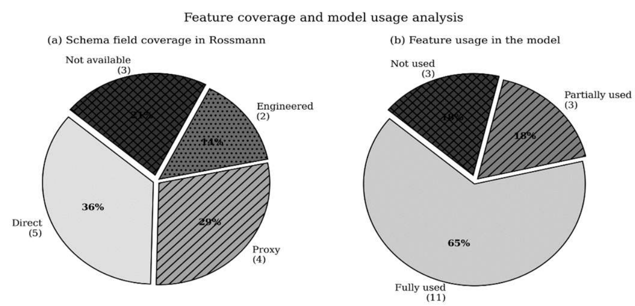

Fig. 2 Left: schema field coverage in Rossmann (73%). Right: feature usage across the trained model.

- The business-context signals that the ETL layer wrote into the database at transaction time. The final equation is simply the sum of the two stages — Prophet handles the structure and XGBoost handles the correction, clipped at zero because negative sales are meaningless:

$$\hat{y}_{hyb(t)} = \hat{y}_{P(t)} + \hat{y}_{XGB(t)}, \hat{y}_{hyb(t)} \geq 0 \quad (1)$$

One important thing was the future lag features. For the test window we do not know about tomorrow's actual sales, so we cannot fill the lag1 feature with the real value. Instead we opted to propagate a rolling average one day at a time each day estimating the next feed, avoiding data leakage.

B. Experimental Setup and Results

The Rossmann Store sales dataset was chosen as the main benchmark because of its high cardinality time series and European holiday seasonality provide a relatable and complex environment to test the Prophet-XGBoost hybrid model's ability to handle multi period spikes [14]. This approach serves as a" Cold Start" proxy for Indian MSMEs where similar high-variance patterns exists because of festivals like Diwali and Holi. Five Rossmann stores (262,562,85,2,251) covering all stereotypes were evaluated on a 48-day holdout. Baselines: Naive, Seasonal Naive, ARIMA (2,1,2), Prophet alone. The hybrid achieved 86.79% mean accuracy with 12.74% standard deviation, leading on all baselines. Store 562 reaches 95.88%, a 20.2% SMAPE reduction compared to Prophet (4.12% vs. 5. 16%).Store 85 reaches 94.60%(13.9% reduction). Store 262 reaches

95.11%(3.2% reduction). Store 2 yields the largest absolute correction, SMAPE drops from 38.48% to 18.10% confirming XGBoost provides greatest lift where prophet leaves systematic residuals. Store 251 gains with non-significant DM results against Seas. Naïve (p=0.813) and Prophet(p=0.245) confirms an irreducible noise floor in high variance stochastic stores. Tables I and II report per-store metrics and mean accuracy using SMAPE and MASE as evaluation criteria [13].

TABLE I: DEMAND FORECASTING RESULTS—SELECTED STORES

Store	Model	MAE	RMSE	SMAPE (%)	MASE	Accuracy (%)
262	Naïve	3744	5462	17.33	0.850	82.67
	Seas. Naïve	2307	2909	11.59	0.524	88.41
	ARIMA (2,1,2)	3193	4419	14.49	0.725	85.51
	Prophet	1070	1441	5.05	0.243	94.95
	Prophet+XGB	996	1255	4.89	0.226	95.11
562	Prophet	911	1225	5.16	0.364	94.84
	Prophet+XGB	722	986.	4.12	0.288	95.88
85	Prophet	463	612	6.27	0.225	93.73
	Prophet+XGB	377	480	5.40	0.183	94.60
2	Prophet	490	726	38.48	0.231	61.52
	Prophet+XGB	406	651	18.10	0.191	81.92
251	Prophet	1737	2322	37.55	0.226	62.45
	Prophet+XGB	2179	2910	33.52	0.283	66.48

TABLE II: MEAN ACCURACY AND DIEBOLD–MARIANO SUMMARY

Model	Mean Accuracy (%)	Standard Deviation	DM wins/5
Prophet + XGBoost	86.79	12.74	-
Seasonal Naïve	85.31	6.00	4/5p < 0.01
Prophet only	81.50	17.82	1/5p < 0.10
ARIMA (2,1,2)	70.71	18.39	5/5p < 0.01
Naive	57.58	23.95	5/5p < 0.01

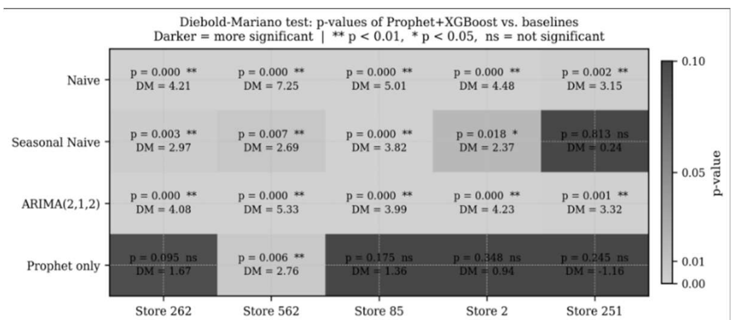

Fig. 3 Diebold–Mariano p-value heat map.

V. CREDIT RISK ASSESSMENT

A. Schema and Pipeline

A credit transaction defaults when delayDays>15 (ETL computed on each payment trigger event).Table III lists all the 14 features and the label as well. festivalPeriod credit carries elevated default risk; salaryWeek credit benefits from near-term repayment capacity. XGBoost (n = 300, depth = 4, lr = 0.05) is trained with SMOTE balancing and evaluated via 5-fold stratified cross validation. McNemar's test confirms significance against each baseline.

B. Training Data

To address the 'Cold Start' problem in the unorganized Indian retail market, where centralized credit record are either absent or are informal we first trained our classifier with a 10,000 records synthetic dataset which was designed to represent the specific Indian risk factors. The synthetic dataset was generated and used with base repayment probability 0.72, also adjusted for salaryweek (+0.12), small credits (+0.08), festivalperiod (−0.10), large credits (−0.12) and week end issuance capability (−0.06). Five percent label noise and Gaussian feature noise were included to prevent non-realistic class separation.

C. Real World Validation

The Kaggle Credit Risk dataset (28,638 records after preparation and cleaning, 21.7%default rate) used for the real-world benchmark. Table IV shows fields correspondence. Table V reports classification performance on both datasets. Fig. 4, 5 and 6 provide supporting visualisations

TABLE III: CREDIT RISK FEATURE SCHEMA

Feature	Source	Role
Credit Amount	Direct	Credit size
Amount Paid	Direct	Repayment progress
Remaining Amount	Direct	Outstanding balance
Payment Count	Direct	Instalment behaviour
Days to first payment	Direct	Payment urgency
Total Days to repay	Direct	Repayment speed
Delay Days	Direct	Delay severity
Was Late	Direct	Late-Payment flag
Issue Day of Week	Direct	Weekend risk
Issue Month	Direct	Seasonal Risk
Festival Period	Direct	Festival impulse risk
Salary Week	Direct	Cash-availability
Repayment ratio	Derived	Overall repayment rate
Credit Utilization	Derived	Debt Burden
Was Default	Label	Prediction Target

TABLE IV: SCHEMA MAPPING: CREDIT TRANSACTION VS. KAGGLE CREDIT RISK

SCHEMA FIELD	KAGGLE EQUIVALENT	TYPE
Credit Amount	Loan_amnt	Direct
Repayment Ratio	1−loan_percent_income	Proxy
Payment Count	person_emp_length	Proxy
Delay Days	cb_default_on_file	Proxy
Was Default	loan_status	Direct
Festival Period	Not available	Neutral
Salary Week	Not available	Neutral

TABLE V: CREDIT RISK—SYNTHETIC (10K) AND REAL (28.6K)

DATA	MODEL	ACC. (%)	PRECISION	RECALL	F1	AUC	CV-AUC
Synthetic	Log. Reg.	94.10	89.31%	86.41%	0.878	0.931	-
	Dec. Tree	94.50	90.66%	86.61%	0.886	0.917	-
	Ran. Forest	94.50	90.32%	87.02%	0.886	0.930	-
	Grad. Boost	94.40	90.11%	86.82%	0.884	0.933	-
	XGBoost	94.55	90.51%	87.02%	0.887	0.928	0.967±0.002
Real	Log. Reg.	79.42	51.93%	67.20%	0.586	0.824	-
	Dec. Tree	82.68	59.22%	64.46%	0.617	0.846	-
	Ran. Forest	83.66	60.52%	70.67%	0.652	0.864	-
	Grad. Boost	84.57	64.16%	65.19%	0.647	0.866	-
	XGBoost	84.86	65.30%	64.30%	0.648	0.867	0.943±0.003

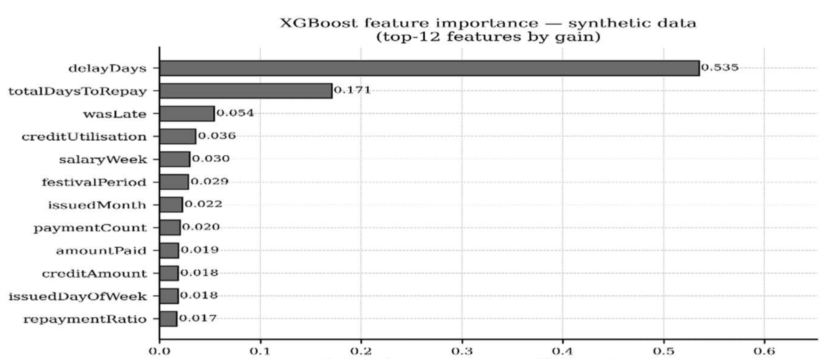

Fig. 4 Delay Days and repayment Ratio dominate. Festival Period and salary Week both rank in the top ten, validating the novel schema contribution

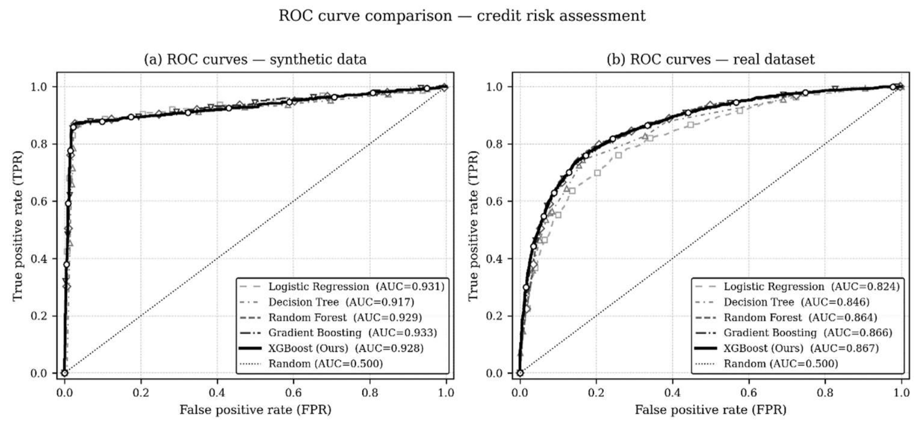

Fig. 5 ROC curves. Left: synthetic (XGB AUC=0.928). Right: real Kaggle (XGB AUC=0.867), exceeding the 0.75–0.85 industry benchmark range

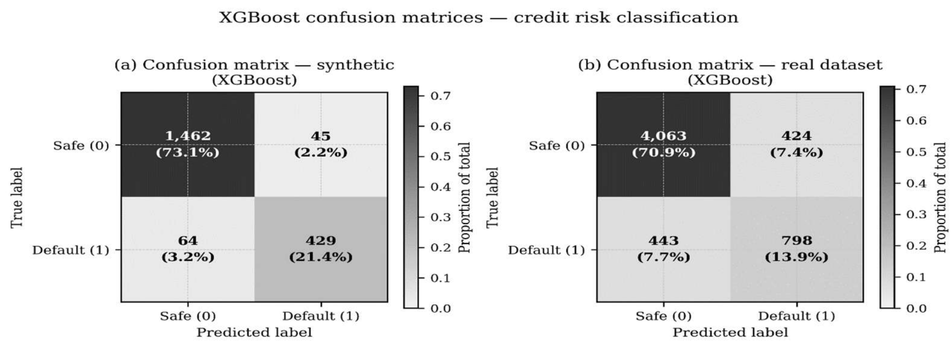

Fig.6 XGBoost confusion matrices. Left: synthetic. Right: real Kaggle dataset.

D. Results

On the real dataset (28,638 records, 21.7% default rate), XGBoost achieves ROC-AUC=0.8672, accuracy=84.86%, precision=65.30% and recall=64.30%, exceeding the industrial benchmark of 0.75-0.85 [5]. McNemar's test confirms the significant outperformance over Logistic Regression, Decision Tree and Random Forest (all $p<0.001$). The confusion matrix Fig 6. Shows balanced error rates of 7.4 % false negative and 7.7% false positives. The CV-AUC of 0.9431 ± 0.0034 reflects stable generalisation under SMOTE-balanced cross validation. Fig. 4 Confirms Delay Days (0.535) and Total Days to Repay (0.171) as dominant features. Salary Week and Festival Period ranks in top six even when set to neutral.

VI. Discussion and Conclusion

This hybrid forecasting model performs better as compared to all baselines on the stores with stable seasonal patterns. Store 251's modest gradual lift confirms that the XGBoost corrects only the systematic residuals, random variance was not learnable. The credit classifier performs well and exceeds the 0.75–0.85 industry benchmark despite the fact it was trained on synthetic data, indicating that the schema generalises. Despite initial training on the synthetic data due to the absence of structurally sound informal credit records, the model demonstrated strong generalization by attaining consistent performance on real world dataset. This indicates that the feature schema and the learning framework captures transferable risk patterns, validating their application in modern MSME environment. The complete stack deploys on a single cloud Virtual Machine at a cost that is affordable to the kirana grade businesses. Three directions remain for future work:

- live deployment that will be used to collect authentic udhaar(informal credits) default data labels, a pilot deployment across 3-5 stores in the local area. This will allow for first empirical validation of features like festival Period and salary Week feature weights.
- hierarchical multi product forecasting for the growing MSMEs

- offline first sync for semi-rural areas with unreliable connectivity so that their operations do not get hampered with the unavailability of internet connectivity.

REFERENCES

[1] "Assessing the Need for the Adoption of Digitalization in Indian Small and Medium Enterprises," De Gruyter Engineering, vol. 6, no. 1, 2024.
[2] "Roadmap for Digital Technology to Foster India's MSME Ecosystem —Opportunities and Challenges," PMC Springer, 2021.
[3] R. Mittal and Sheetal, "Fostering Competitiveness of Indian MSMEs Through IT and Digitalization," Inf. Technol. Dev., vol. 31, no. 4, 2025.
[4] R. Fildes, S. Ma, and S. Kolassa, "Retail Forecasting: Research and Practice," Int. J. Forecast., vol. 38, no. 4, pp. 1283–1318, 2022.
[5] "Machine Learning for Credit Risk Prediction: A Systematic Literature Review," MDPI Data, vol. 8, no. 11, p. 169, 2023.
[6] P. Brandtner et al., "Applying ML in Retail Demand Prediction," Appl. Sci., vol. 13, no. 19, p. 11112, 2023.
[7] S. J. Taylor and B. Letham, "Forecasting at Scale," Am. Stat., vol. 72, no. 1, pp. 37–45, 2018
[8] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proc. ACM KDD, pp. 785–794, 2016.
[9] F. X. Diebold and R. S. Mariano, "Comparing Predictive Accuracy," J. Bus. Econ. Stat., vol. 13, no. 3, pp. 253–263, 1995.
[10] N. V. Chawla et al., "SMOTE: Synthetic Minority Over-sampling Technique," J. Artif. Intell. Res., vol. 16, pp. 321–357, 2002.
[11] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "The M5 Accuracy Competition," Int. J. Forecast., vol. 38, no. 4, pp. 1325–1336, 2022.
[12] G. E. P. Box and G. M. Jenkins, Time Series Analysis: Forecasting and Control. Holden-Day, 1970.
[13] R. J. Hyndman and A. B. Koehler, "Another Look at Measures of Forecast Accuracy," Int. J. Forecast., vol. 22, no. 4, pp. 679–688, 2006.
[14] T. Falatouri et al., "Predictive Analytics for Demand Forecasting," Procedia Comput. Sci., vol. 200, pp. 993–1003, 2022
[15] T. Januschowski et al., "Criteria for classifying forecasting methods," Int. J. Forecast., vol. 36, no. 1, pp. 167–177, 2020.
[16] C. Luo et al., "Fintech credit risk assessment for SMEs: Evidence from China," Finance Research Letters, vol. 55, p. 103933, 2023.
[17] S. Shen et al., "A cost-sensitive logistic regression credit scoring model based on multi-objective optimization," Complexity, vol. 2020, Article ID 7979809, 2020