iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.81803

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

Intelligent Banking Analytics System for Loan Approval and Customer Churn Prediction Using Machine Learning

Gokul U[1], Murali S[2], Anuj B[3] Ajith A[4], Mrs. D. Ruba[5]
[1, 2, 3, 4]*UG Student, Dept. of Information Technology, Meenakshi College of Engineering, Chennai – 600 078*
[5]*Assistant Professor, Dept. of Information Technology, Meenakshi College of Engineering, Chennai – 600 078 | Anna University, Chennai – 600 025*

Abstract: ***In the rapidly evolving financial sector, the demand for secure, efficient, and accurate loan approval systems has increased significantly. Traditional processes involve manual verification, subjective decision-making, and time-consuming procedures leading to delays. This paper presents LoanOracle, a machine learning-based prediction and approval system that employs Logistic Regression, Decision Tree, and Random Forest for loan eligibility prediction, and Voting Classifier (GaussianNB + AdaBoost + KNN) and Extra Tree Classifier for customer churn prediction. Models are trained on historical banking datasets evaluating credit history, income, employment status, and repayment capacity. The Django-based web application provides real-time feedback, an admin dashboard for monitoring, and analytical reporting — reducing processing time, eliminating human bias, and enhancing customer satisfaction.***

Keywords: Machine Learning, Loan Approval Prediction, Customer Churn, Random Forest, Voting Classifier, Extra Tree Classifier, Django, Scikit-learn, Predictive Modelling, Intelligent Banking Analytics

I. INTRODUCTION

In today's fast-paced financial environment, banks and financial institutions face growing challenges processing large volumes of loan applications while simultaneously retaining their existing customer base. Traditional loan approval systems rely on manual processes — slow, inconsistent, and prone to human bias. Customer churn, the loss of clients to competing institutions, poses a significant threat to banking profitability. Machine learning (ML) enables automation of complex decisions previously requiring expert judgment. This paper presents LoanOracle, an Intelligent Banking Analytics System simultaneously addressing: (1) automated loan approval prediction and (2) customer churn prediction using algorithms including Logistic Regression, Decision Trees, Random Forest, Voting Classifier, and Extra Tree Classifier. The full-stack Django web application is powered by Scikit-learn with secure Supabase data storage and a responsive HTML/CSS/JavaScript frontend. The system is guided by five UN SDGs: SDG 8 (Economic Growth), SDG 9 (Innovation), SDG 1 (No Poverty), SDG 10 (Reduced Inequalities), and SDG 16 (Strong Institutions).

II. LITERATURE SURVEY

Ugwuishiwu et al. (2022) explored Random Forest, Decision Tree, SVM, and Logistic Regression for loan eligibility — Random Forest achieved highest accuracy. Majumdar et al. (2022) highlighted the critical importance of data preprocessing and feature selection. Boddepalli (2022) demonstrated that classification algorithms reduce manual intervention, though performance depends on training data quality. Kadam et al. (2021) concluded Naïve Bayes and Random Forest provide superior accuracy and efficiency. Murthy (2020) emphasised automated banking systems using Random Forest to reduce loan defaults. The present work builds upon these findings by integrating both loan prediction and churn detection into a unified, deployed system.

III. SYSTEM ANALYSIS

A. *Existing System Limitations*

1) Relies on manual verification — causes delays and inconsistencies in approval.
2) Rule-based evaluation misses deeper data-driven insights and predictive accuracy.
3) Lacks real-time processing and automated decision-making capabilities.
4) Cannot handle large application volumes efficiently — poor scalability.
5) High susceptibility to human bias; no personalised recommendations for applicants.

B. *Proposed System*

LoanOracle uses ML models trained on historical datasets to analyse applicant data and provide accurate, real-time loan and churn decisions. The Python/Scikit-learn backend integrates with Supabase for secure storage. A React/HTML frontend delivers a responsive user experience. An admin dashboard provides real-time monitoring, approval management, and analytical reports.

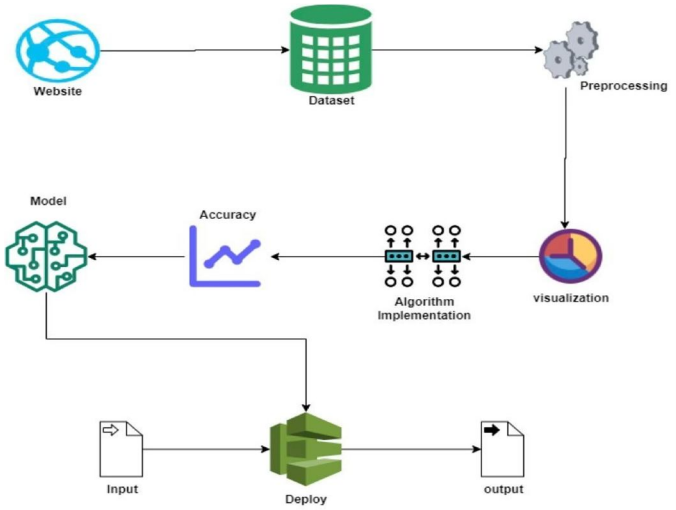


Fig. 1. Proposed System Architecture and Data Flow

IV. SYSTEM MODULES

1) User Input Module: User-friendly interface for loan applications; validates and securely transmits data.
2) Preprocessing Module: Handles missing values, encodes categorical variables, removes outliers, normalises features.
3) Feature Extraction Module: Identifies key features (income, credit history, loan amount); reduces noise and redundancy.
4) Loan Prediction Module: Logistic Regression, Decision Tree, and Random Forest predict eligibility with probability scores.
5) Risk Analysis Module: Evaluates application risk, analyses repayment capacity, identifies high-risk applicants.
6) Churn Prediction Module: Voting Classifier and Extra Tree Classifier identify at-risk customers from bank transaction features.
7) Storage & Security Module: Secure database storage with authentication-based access control and data integrity enforcement.
8) Admin Management Module: Dashboard for monitoring applications, managing approvals, generating reports and trend analysis.

V. METHODOLOGY

A. *Dataset and Preprocessing*

Two datasets are used: (1) CHURN.csv — 10,000 bank customer records with features: CreditScore, Age, Tenure, Balance, NumOfProducts, HasCrCard, IsActiveMember, EstimatedSalary, Exited (target); (2) LOAN_ELIGIBLE.csv — loan records with Gender, Married, Dependents, Education, Self_Employed, ApplicantIncome, CoapplicantIncome, LoanAmount, Loan_Amount_Term, Property_Area, Loan_Status (target).

Preprocessing: irrelevant columns removed (RowNumber, CustomerId, Surname, Geography); categorical variables label-encoded via Scikit-learn LabelEncoder; missing values handled via dropna(); class imbalance resolved using RandomOverSampler from imbalanced-learn (original churn ratio: 79.63% Stayed vs 20.37% Exited).

B. *ML Algorithms*

Algorithm	Task	Advantage
Logistic Regression	Loan	Fast; interpretable
Decision Tree	Loan	Non-linear splits
Random Forest	Loan	Highest accuracy
Voting Classifier	Churn	Ensemble; low variance
Extra Tree Classifier	Churn	Fast; low overfit

Table I. ML Algorithms in LoanOracle

C. Training Configuration

Data split 80:20 (train_test_split, random_state=42, stratify=y). Five-fold cross-validation (cross_val_score) assesses generalisation. Models serialised with joblib as CHURN.pkl and LOAN.pkl, integrated into Django for real-time inference. Metrics: accuracy, precision, recall, F1-score, confusion matrix, and Hamming loss.

Table II. Model Performance Metrics

Model	Task	Acc.	F1
Logistic Reg.	Loan	~82%	0.81
Decision Tree	Loan	~85%	0.84
Random Forest	Loan	~91%	0.90
Voting Classif.	Churn	~86%	0.85
Extra Tree	Churn	~88%	0.87

VI. SYSTEM ARCHITECTURE

LoanOracle uses a three-tier architecture:
(1) Presentation Layer — HTML/CSS/JavaScript responsive frontend;
(2) Business Logic Layer — Django backend handling routing, form validation, ML inference, and authentication;
(3) Data Persistence Layer — MySQL/PostgreSQL/Supabase storing user profiles, loan records, and churn histories.
End-to-end flow: User submits form → Django extracts POST parameters → pre-loaded joblib model generates prediction → result rendered with confidence score → admin dashboard aggregates all predictions for reporting and monitoring.

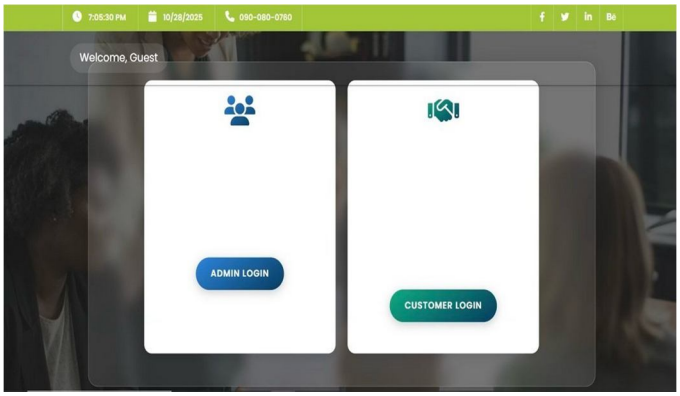


Fig. 2. Login Portal — Admin Login and Customer Login

VII. IMPLEMENTATION

Component	Technology
Language	Python 3.x
Web Framework	Django / Flask
ML Libraries	Scikit-learn, Pandas, NumPy
Frontend	HTML5, CSS3, JavaScript, React
Database	MySQL / Supabase
Visualisation	Matplotlib, Seaborn
Model Export	Joblib (.pkl)

Table III. Technology Stack

The Django *Deploy_9* view handles churn: extracts POST parameters, converts to NumPy array, invokes CHURN.pkl, returns 'Client has left the bank' or 'Client has not left the bank'. *Deploy1_11* handles loan prediction via LOAN.pkl — 'YES, you are eligible' or 'Sorry, not eligible'. The Voting Classifier combines GaussianNB, AdaBoostClassifier, and KNeighborsClassifier with hard voting. Django's built-in auth framework manages login, logout, and registration.

VIII. RESULTS AND DISCUSSION

LoanOracle was successfully implemented across all core modules. The interface provides intuitive navigation for registration, login, loan submission, churn analysis, and admin monitoring.

Fig. 3(a). Registration Form

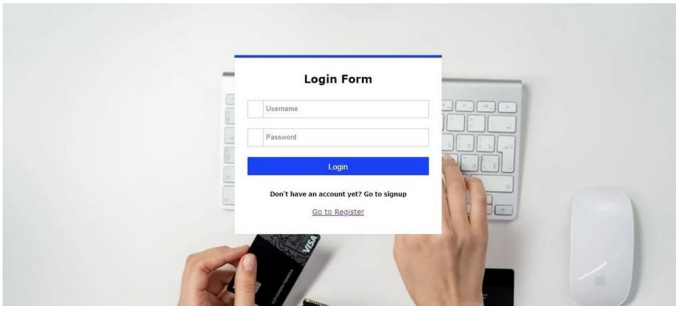


Fig. 3(b). Login Form

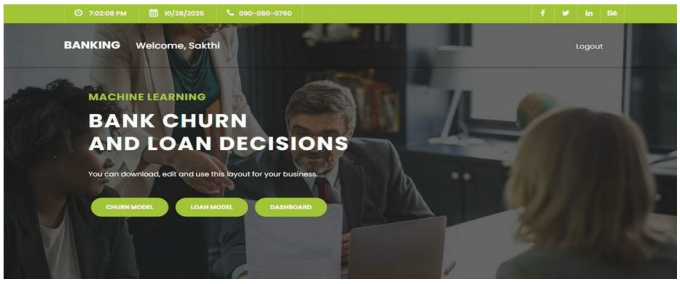


Fig. 3(c). Main Dashboard — Churn Model, Loan Model, Dashboard

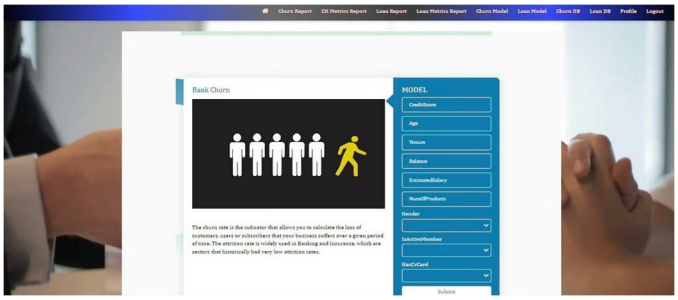


Fig. 3(d). Bank Churn Prediction Input with All Financial Parameters

Fig. 3 shows the main dashboard exposing three core modules. The churn model accepts CreditScore, Age, Tenure, Balance, NumOfProducts, HasCrCard, IsActiveMember, and EstimatedSalary to predict customer exit probability.

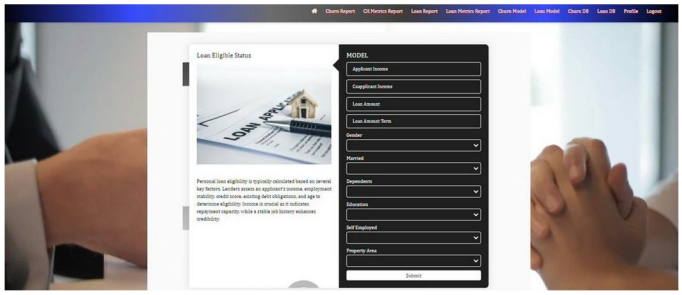


Fig. 4(a). Loan Eligible Status — Model Input Form

Fig. 4(b). Apply for a Loan — Personal and Professional Details Form

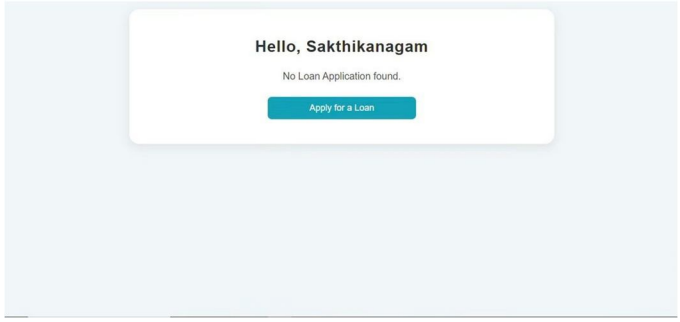


Fig. 4(c). User Dashboard — No Loan Application Found

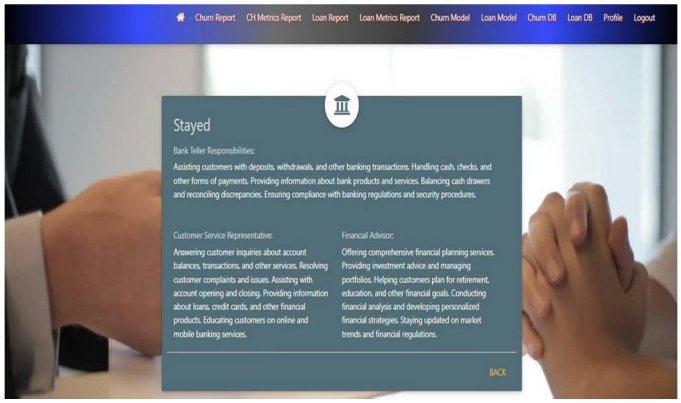


Fig. 4(d). Churn Result with Bank Teller Career Recommendations

Fig. 4 shows the loan application form collecting gender, marital status, dependents, education, and employment details. The system provides personalised role guidance (Bank Teller, Customer Service Representative, Financial Advisor) for retained customers based on churn outputs.

Fig. 5(a). Bank Churn Prediction Database with All Features and Predictions

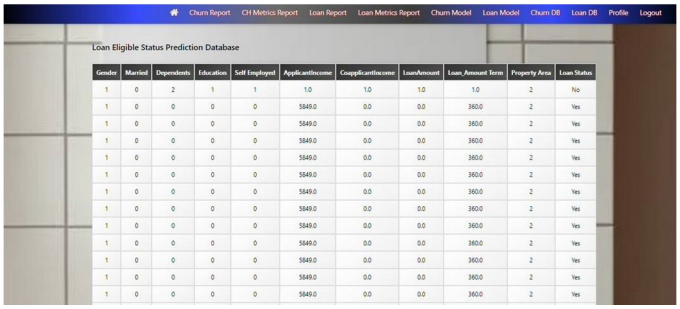


Fig. 5(b). Loan Eligible Status Prediction Database

Fig. 5 demonstrates system transparency. The Bank Churn Prediction Database displays all input features alongside predicted output (Stayed/Exited). The Loan Eligible Status Prediction Database records all application parameters and Loan_Status predictions (Yes/No), providing a fully auditable decision trail for regulatory compliance.

A. SDG Alignment

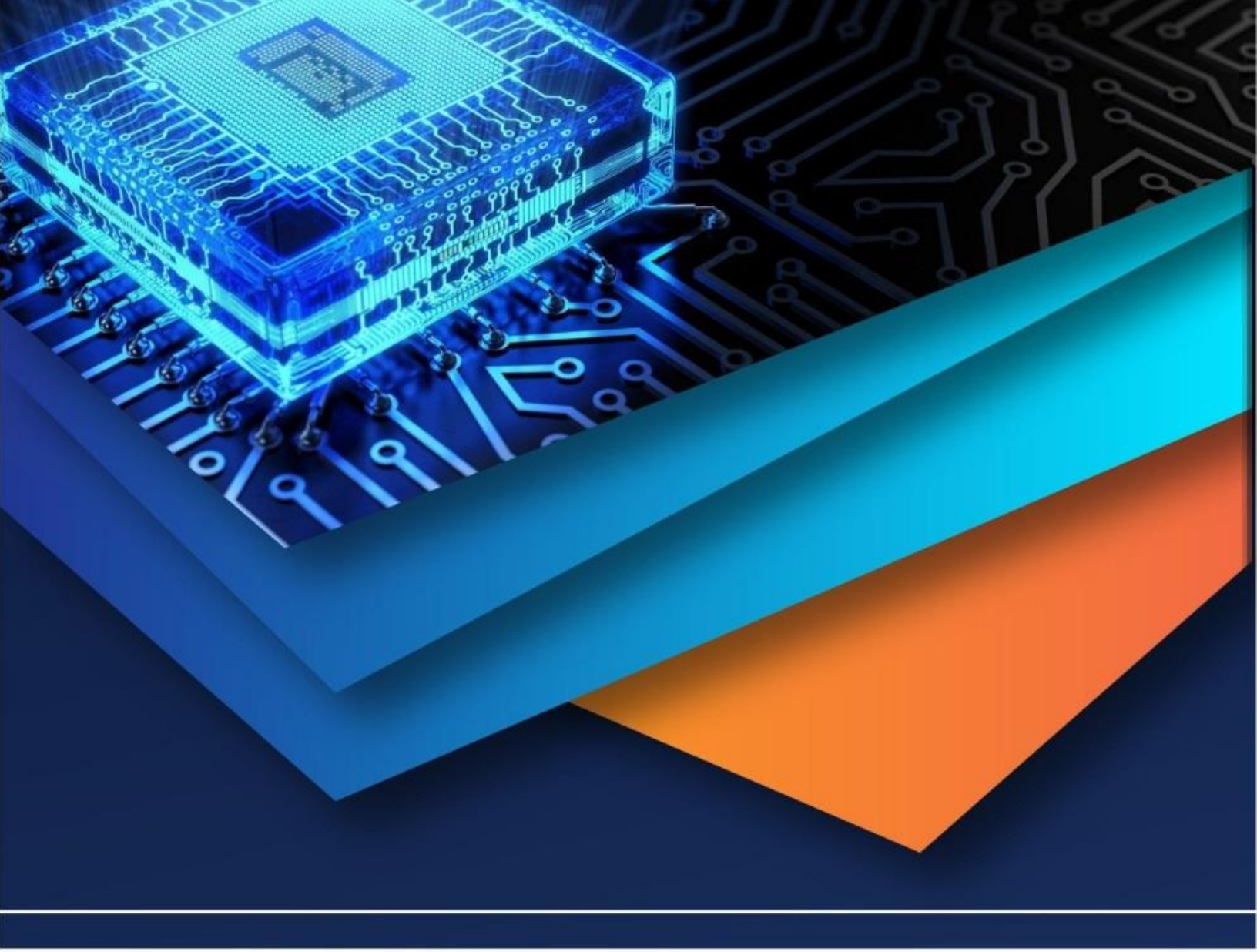
LoanOracle aligns with SDG 8 (automating loan decisions boosts economic productivity), SDG 9 (ML in banking builds smart digital infrastructure), SDG 1 (ML-based approval extends credit to low-income populations), SDG 10 (eliminating human bias ensures fairer outcomes), and SDG 16 (auditable AI decisions promote institutional accountability).

IX. CONCLUSION AND FUTURE WORK

This paper presented LoanOracle, an intelligent banking analytics system integrating loan approval prediction and customer churn prediction into a unified, deployed web application. By combining Logistic Regression, Decision Tree, Random Forest, Voting Classifier, and Extra Tree Classifier, the system provides accurate, real-time decisions that reduce manual effort, minimise human bias, and enhance efficiency. The modular Django-based architecture with secure database storage and an intuitive admin dashboard positions LoanOracle as a scalable solution for modern banking. Future enhancements: (1) real-time banking API integration for live credit score retrieval; (2) deep learning models for complex financial pattern recognition; (3) fraud detection module using anomaly detection; (4) mobile application for field-level accessibility; (5) AI-powered chatbot for application guidance; (6) cloud-based deployment with big data infrastructure.

REFERENCES

[1] Ugwuishiwu, C. H. et al. (2022) 'Machine Learning Models for Predicting Bank Loan Eligibility,' Intl. Journal of Advanced Trends in Computer Science and Engineering, Vol. 11, No. 2.
[2] Majumdar, A. et al. (2022) 'Loan Prediction using Machine Learning Techniques,' Intl. Journal of Computer Applications, Vol. 184, No. 25, pp. 1–6.
[3] Boddepalli, R. (2022) 'Loan Approval Prediction System Using Machine Learning,' Intl. Journal of Engineering Research and Technology, Vol. 11, No. 5, pp. 654–658.
[4] Kadam, A. S., Patil, S. and Kulkarni, P. (2021) 'Prediction for Loan Approval using Machine Learning Algorithm,' Intl. Journal of Scientific Research in Computer Science, Vol. 9, No. 3, pp. 45–50.
[5] Murthy, P. L. S. (2020) 'Loan Approval Prediction System Using Machine Learning,' Intl. Journal of Engineering and Technology, Vol. 8, No. 4.
[6] Breiman, L. (2001) 'Random Forests,' Machine Learning, Vol. 45, No. 1, pp. 5–32.
[7] Pedregosa, F. et al. (2011) 'Scikit-learn: Machine Learning in Python,' Journal of Machine Learning Research, Vol. 12, pp. 2825–2830.
[8] Sundararajan, S. and Ravi, V. (2020) 'Machine Learning Approach for Credit Risk Assessment,' Intl. Journal of Advanced Computer Science and Applications, Vol. 11, No. 6, pp. 250–256.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY