WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Bank Loan Prediction Using Machine Learning

Monika Belwal[1], Vivek Pratap Singh[2], Yash Tayal[3], Tanish Saxena[4], Swasti Jain[5], Shristi[6]

[1] Assistant Professor, Department of Computer Science and Engineering, IMS Engineering College, Ghaziabad, UP, monika.belwal@imsec.ac.in, vivekpratap025@gmail.com , yashtayal63@gmail.com , tanishsaxena79@gmail.com , jswasti35@gmail.com, imsrishti18@gmail.com, India.
[2,3,4,5,6] Student, Department of Computer Science and Engineering, IMS Engineering College, Ghaziabad, UP, India.

ABSTRACT :

The increasing demand for bank loans corresponds to a surge in loan applications received daily, prompting banks to undertake rigorous assessments of each applicant's eligibility. This process, often time-consuming and intricate, involves scrutinizing credit scores and risk assessment systems to evaluate creditworthiness. Despite these measures, instances of default persist, resulting in significant financial losses for institutions. To address this challenge, machine learning (ML) algorithms are being employed to analyze patterns in loan approval datasets and predict viable loan candidates. Leveraging various ML techniques such as Random Forest, Support Vector Machine, Naïve Bayes, Decision Tree, and K Nearest Neighbor, this study aims to identify key factors influencing loan prediction outcomes. By analyzing factors including age, income type, loan amount, credit history, employment details, and association type, the study aims to optimize prediction accuracy. Through rigorous comparison and evaluation, Logistic Regression emerges as the most effective model, achieving a sensitivity of 92 and outperforming other ML algorithms in terms of F1-Score, which stands at 96.

Keywords: Loan Sanction (LS), Machine Learning (ML), Support Vector Machine (SVM).

1. INTRODUCTION

The burgeoning digitization within the fiscal sector has led to a preference for online loan applications among individuals, with artificial intelligence (AI) emerging as a pivotal tool for data analysis across various industries. However, banks are encountering challenges in efficiently approving loans amidst the surge in daily operations, highlighting the criticality of error-free processes due to the substantial financial stakes involved. To address this, many banks have implemented automated systems powered by machine learning algorithms for loan prediction, offering benefits to both bank staff and loan applicants. This paper aims to present a streamlined and effective system for identifying promising loan candidates, automating the verification and confirmation process based on various applicant characteristics. By assigning weights to these characteristics and utilizing machine learning techniques, the Loan Prediction System can accurately assess an applicant's suitability and provide timely decisions, facilitating prioritization of loan operations. Key features such as gender, marital status, income, credit history, and property details are considered in this approach, which is detailed across six subsections covering literature review, dataset overview, methodology, algorithm selection, and conclusion.

2.LITERATURE SURVEY

Vaticination, essentially predicting future events or outcomes based on beliefs or analysis, is a common practice with varying degrees of seriousness and reliability. While some prognostications are deeply rooted in scientific computations, others rely more on intuition or mere guesswork. Regardless, predictions serve diverse purposes, aiding us in envisioning potential scenarios over different timeframes, whether short-term, long-term, or spanning decades. Prophetic analytics, a specialized branch of advanced analytics, employs a range of methodologies including data mining, statistical analysis, modeling, machine learning, and artificial intelligence to analyze present data and forecast future trends or occurrences, contributing to informed decision-making and strategic planning.

Drawing inspiration from Kumar Arun's (2016) study on predicting bank loan approvals using machine learning technologies like Support Vector Machines (SVM) and neural networks, we embarked on our own investigation. Their model provided valuable insights into the intricacies of loan authorization processes, serving as a foundation for our exploration. By leveraging their findings and methodologies, we developed a robust and reliable bank loan prediction model, enhancing our understanding and capabilities in this domain.
In Mohammad's (2010) study, the focus was on predicting whether a bank would grant a loan to a client, aiming for accuracy in model performance. Utilizing Logistic Regression with a sigmoid function, the model was developed to achieve this objective. The dataset, sourced from Kaggle, comprised separate sets for training and testing, necessitating data cleansing to address any missing values. Performance metrics such as sensitivity and specificity were employed to assess and compare model effectiveness. Ultimately, the model achieved an accuracy of 81, indicating its capability in prediction. Notably, the model's improvement stemmed from its inclusion of variables beyond checking account information, such as client demographics, credit

history, and loan characteristics, which are crucial in accurately assessing the likelihood of loan default. By leveraging logistic regression to calculate the probability of default, the study highlighted the potential for targeting suitable loan candidates with greater precision and efficacy.

In Pidikiti's (2019) study, the primary objective was to mitigate the risk associated with selecting safe borrowers for loan assignment, thereby optimizing time and capital utilization for banks. The paper was structured into four main sections: data collection, machine learning model comparison, system training using the selected model, and testing. Utilizing machine learning algorithms including Support Vector Machines, Logistic Regression, Decision Trees, and Gradient Boosting, the study examined loan data to identify the most effective approach. Notably, the decision tree algorithm emerged as the most accurate, achieving a sensitivity of 82 percent. Its success stemmed from its superior performance in addressing classification problems and its user-friendly nature, making it easily deployable and yielding interpretable results.

Pandey (2010) underscores the considerable challenge banks face in predicting loan defaulters, emphasizing the potential for substantial losses if not effectively addressed. However, by accurately predicting loan defaulters, banks can mitigate losses and improve profitability, making the exploration of loan approval prediction crucial. Machine learning methods play a pivotal role in this prediction process, with Pandey's study employing four bracket-based algorithms: Logistic Regression, Decision Trees, Support Vector Machines, and Random Forest. Notably, the Support Vector Machine approach emerged as the most accurate, boasting a high sensitivity of 79.67% in predicting loan acceptance. The study utilized a dataset comprising historical customer information from various banks that had previously granted a range of loans.

In Ndayisenga's (2021) collaboration with marketable banks, the focus was on predicting borrower behavior by developing and evaluating various models using data sourced from the Bank of Kigali. The dataset was partitioned into training and test sets, with the training data accounting for 70% and the test data for 30% of the total. Utilizing ensemble methods, the study aimed to identify the most effective machine learning strategies for predicting bank loan default. Among these methods, Gradient Boosting emerged as the most effective model, achieving an accuracy of 80.40% in predicting loan default, followed by XGBoosting. Conversely, decision trees, random forests, and logistic regression exhibited poorer performance in this context.

Tejaswini's (2020) study introduced a robust predictive modeling system designed to streamline the decision-making process for loan approvals based on guests' financial and credit scores. The primary objective was to establish a swift, user-friendly, and efficient system for identifying suitable loan candidates. Data was collected from various financial institutions, with the training dataset utilized to train the machine learning model. Each new applicant's information served as a test dataset upon entry into the loan application form. The study employed three machine learning algorithms—Logistic Regression (LR), Decision Tree (DT), and Random Forest (RF)—to predict customer loan approval. Testing results revealed that the Decision Tree algorithm outperformed Logistic Regression and Random Forest methods, achieving a higher sensitivity of 82.00%.

Kumar (2016) introduced a model aimed at predicting loan approval for individuals. The primary objective was to assess whether an individual could secure a loan by analyzing data using decision tree classifiers, which yielded a sensitivity of 82.00%. The datasets were obtained from Kaggle and divided into two categories: existing guests and new guests. Each new applicant's information was utilized as a set of test data to evaluate the model's performance.

Madane (2016) developed a model utilizing decision tree induction to analyze the credit scores of mortgage loans and applicant conditions. With credit scores being a crucial factor in loan approval, the study aimed to predict the safety of loan sanctioning. Interestingly, the analysis revealed that a significant number of low-income applicants were being approved for loans. Through reviewing various datasets and model performances, it was determined that the Random Forest algorithm exhibited the highest sensitivity among all models.

In their work, Shrishti et al. (2018) introduced a robust machine learning model aimed at predicting loan approval efficiently. The primary objective of this model was to expedite the loan approval process for applicants. Three types of machine learning algorithms—Logistic Regression, Decision Tree, and Random Forest—were employed in the study. Upon examining various datasets and model performances, it was found that the Random Forest algorithm demonstrated the highest sensitivity among all models.

Karthiban (2019) conducted a review focusing on machine learning algorithms for bank loan approval. In today's world, the majority of operations are driven by machine learning algorithms, yet their performance and accuracy remain challenging to perfect. Data for the study was obtained from a bank, and various machine learning algorithms were assessed for their performance in predicting loan approval. Grade Boosting emerged as the top-performing classifier, surpassing others in terms of accuracy, precision, recall, and F1 score, achieving a delicacy of 98.06% and an F1 score of 99.20%.

Table 2.1: SUMMARY OF SOME LITERATURE REVIEW

Authors (year)	Datas et Colle ction (sampl es)	Applied Models	Measur es (Propo sed model)
Moham mad , G. Arutjothi (2020)	Kaggle (1500 cases)	Logistic regression [Proposed]	Accura cy: 81.00 %

Pidikiti Supriya., M. Pavani, N. Saisushma, N. Kumari and K. Vikas (2019)	From previous customers of Bank (1000 cases and 7 numerical and 6 categorical attributes.)	Logic regression, Decision Tree [proposed model]and Gradient Boosting	Accuracy: 82.00 %
Nitesh Pandey (2021)	From past clients of different banks	Logistic Regression, Decision tree, Support	Accuracy: 79.67 %
M. Bhajibhakare, Pandey. (2010)	Kaggle data source	Vector Machine (SVM) [proposed model] and Random Forest	Precision: 46.00 % Recall: 95.00 % F1Score: 61.00 %
Ndayisenga (2021)	Bank of Kigali	Gradient Boosting [Proposed model] XGBoosting Decision trees Random forest, Logistic Regression	Accuracy: 80.4% Precision: 82.59 % Recal: 80.25 % F1Score: 81.00 %
			Recall: 82.00 % F1Score: 75.00 %
KUMAR, SOURAV(2021)	Kaggle data source	Decision Tree (DT) [Proposed model]	Accuracy: 76.40 % Precision: 59.00 % Recall: 79.83 %
NIKHIL MADANE (2019)	Online	Decision Tree (DT) [Proposed model]	Accuracy: 85%
Shrishti (2018)	Kaggle	Logistic Regression, Decision tree and Random Forest algorithm [proposed	Accuracy: 89.22 %

Tejaswin i (2020)	Finan cial Instituti on	Logistic Regressio n (LR), Decision Tree(DT) [Proposed model] and Random Forest (RF)	Accura cy: 82.00 % Precisi on: 83.00 %
		Deep Learning, Gradient Boosting [Proposed model], Generated linear model	Recal l: 99.30 % F1Sc ore: 99.20 %

		model]	
Karthiban, R. (2019)	Bank	Logistic Regression, Decision tree, Naive Bayes, Random Forest	Accur acy: 98.06 % Precis ion: 99.10 %

3.METHODOLOGY

To create a loan prediction model, we'll need to import essential libraries such as scikit-learn, pandas, and NumPy. The loan data will be loaded into a pandas DataFrame. From this dataset, two subsets will be created: a training set and a testing set. The predictive model will be trained using the training set and evaluated using the testing set. We'll choose a suitable machine learning algorithm, such as Random Forests, Decision Trees, or Logistic Regression, to predict loan approval. An instance of the chosen model will be instantiated and any necessary hyperparameters will be adjusted. The model will then be fitted to the training data using the fit() function. By analyzing patterns and relationships in the training data, the model will learn to classify each loan application as approved or denied. Finally, the model's predictions will be compared to the actual loan approval status in the testing set to evaluate its performance, as depicted in Figure 3.1.

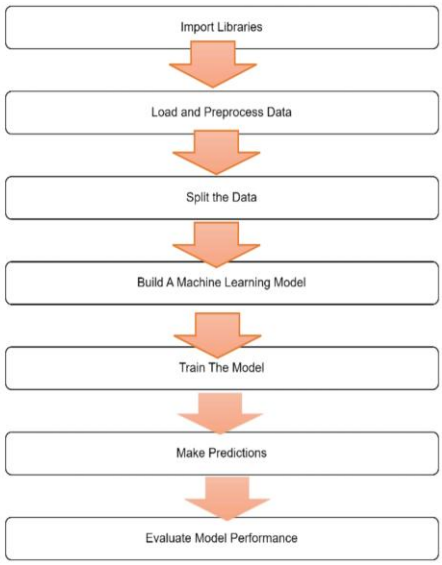


Figure 3.1: Flowchart of Loan Amount Prediction

3.1. Algorithms Used

3.1.1. Random Forest

A highly favored algorithm in the field of machine learning is Random Forest (RF), which is versatile and applicable to both classification and regression problems. Based on the concept of ensemble learning, Random Forest integrates multiple decision trees to address complex problems and enhance model performance. As its name implies, Random Forest constructs a classifier by building numerous decision trees on different subsets of the given dataset and then averages their predictions to improve the predictive accuracy of the model. Unlike relying solely on the decision of one tree, Random Forest combines the predictions from each decision tree and makes its final prediction based on the consensus of the majority of trees. This approach is illustrated in Figure 3.2, highlighting the effectiveness and sophistication of the Random Forest system.

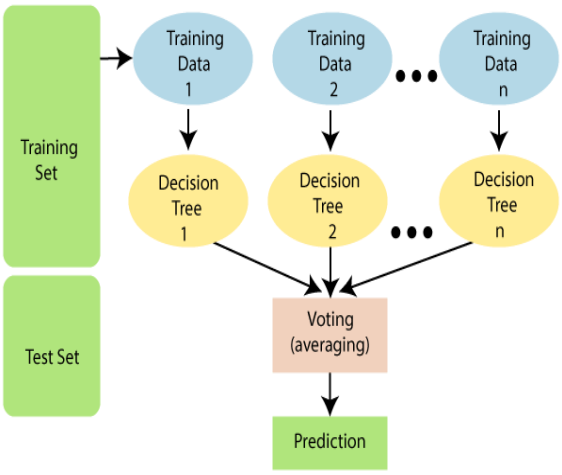


Figure 3.2: Flowchart of Random Forest Algorithm

3.1.2. Naive Bayes

The Naive Bayes algorithm (NB) is a supervised learning system primarily used for classification problems, based on Bayes' theorem. Figure 3.3 illustrates the workflow of the Naive Bayes algorithm, which relies on a large training dataset to perform accurate categorization. Known for its simplicity and effectiveness, Naive Bayes Classifier is widely used in machine learning applications, as depicted in Figure 3.3. It leverages probabilistic reasoning to generate predictions, estimating the likelihood that an object belongs to a particular category. Some common applications of Naive Bayes algorithms include sentiment analysis, document classification, and spam filtering, demonstrating its versatility and utility in various domains.

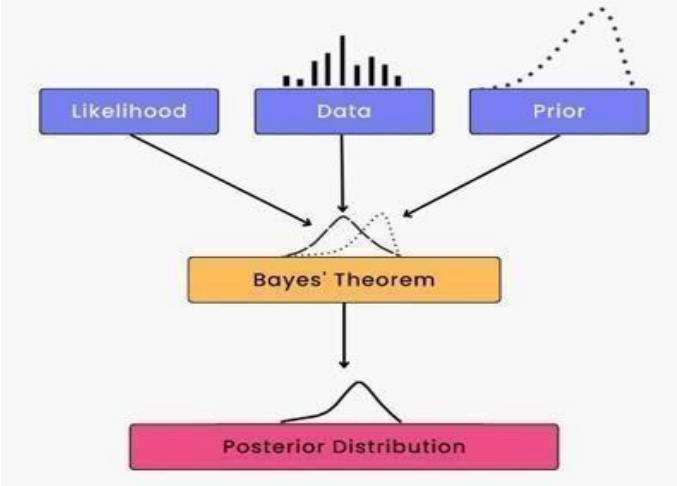


Figure 3.3:
Flowchart for Naive Bayes Algo.

3.1.3. Decision Tree

The decision tree (DT) prediction model utilizes a flowchart-like structure to make decisions based on incoming data. This structure involves branching data paths, with outcomes placed at the leaf nodes of the tree. Decision trees are commonly employed for both regression and classification problems, providing models that are easily interpretable. In decision support systems, decision trees represent hierarchical models that capture various decision paths and their associated factors, such as probability events, resource costs, and outcomes. This algorithmic approach, which is nonparametric and supervised learning, utilizes conditional control statements to construct the tree structure. The tree consists of a root node, branches, internal nodes, and leaf nodes, resembling a hierarchical tree structure.

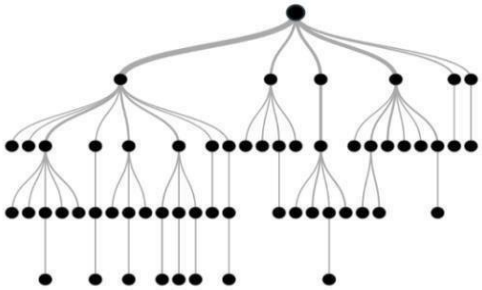
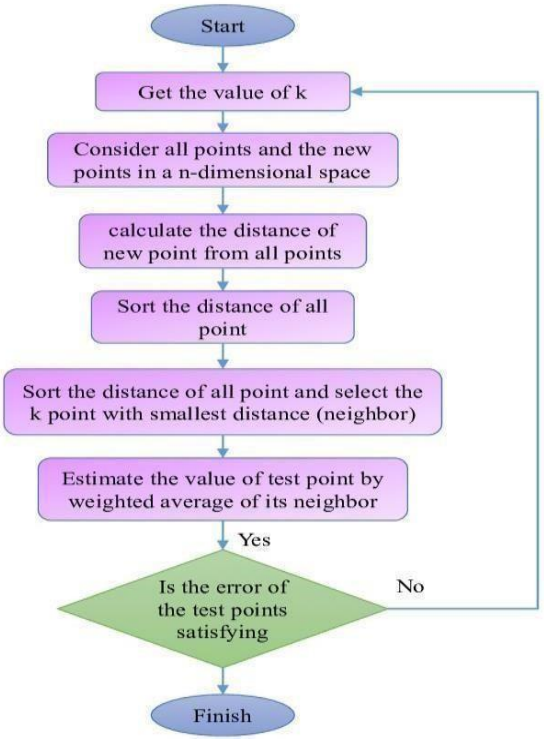


Figure 3.4: Flowchart for Decision Tree (DT) Algorithm

3.1.4. KNN Algorithm

K- Nearest Neighbor, one of the introductory supervised literacy- grounded machine literacy algorithms. The K- NN algorithm places good case, in a order that resembles the current orders the most, presuming that new case, and the former cases are similar. After storing all the former data, a new data point is distributed using the K- NN algorithm grounded on similarity. This indicates that new data can be reliably and snappily distributed using the K-NN approach. Although the K- NN fashion is most constantly worked to break bracket problems, it can also be used for working retrogression, difficulties. KNN is an on-parametric system that makes no hypotheticals about the underpinning data is as shown in the Figure.3.5. As a result of saving dataset of training rather than incontinently learning from it, the system, also known, to as a lazy learner. rather, it performs an action while classifying data by using the dataset. The KNN approach simply stores the data during phase of training and categorizes fresh data into a order that's veritably same for training data.



4. DATASET DESCRIPTION AND PRE-PROCESSING

The dataset for the bank loan prediction system originates from a Kaggle competition and comprises applicants from diverse backgrounds, spanning different periods and genders. It consists of twenty-three attributes, including education, marital status, income, and assets, as depicted in Table 2.1. With a total of [number] applicant records, containing both categorical and numerical data attributes, the dataset undergoes preprocessing to address missing values and standardize the data through feature engineering. Following this preprocessing step, the dataset is partitioned into two sections: training and testing sets. Subsequently, the model is trained using various machine learning techniques, and its predictions are evaluated using the test data, as outlined in the subsequent sections.

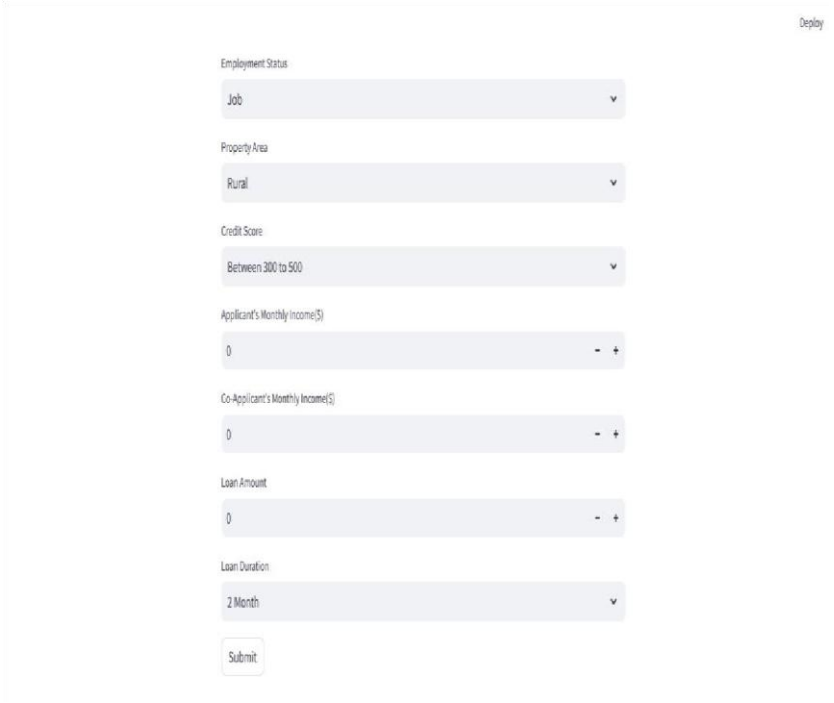


Figure 3.5: Flowchart of KNN Algorithm

Table 2.1 Some of dataset attribute names and information

Variable Name	**Description of Variable**	**Data Type**
Loan ID CLIENTNUM	Unique Loan ID	Integer
Customer Age	Age of Customer	Integer
Gender	Male/ Female	Charact er
Dependents	Number of dependents	Integer
Married	Applicant married (Y/N)	Charact er
Education	Graduate/ Under Graduate	String
Income Category	Income type	String
Card Category	Card type	String
Self Employed	Self Employed (Y/N)	Charact er
Applicant Income	Applicant income	Integer
Co-applicant Income	Co-applicant income	Integer
Loan Amount	Loan amount in thousands	Integer
Loan_Amount_Te rm	Term of loan in months	Integer

Credit History	credit history meets guidelines	Integer
Property Area	Urban/ Semi Urban/ Rural	String
Loan Status	Loan Approved(Y/ N)	String

5. CONCLUSION AND FUTURE SCOPE

In this study, we conducted an in-depth analysis of machine learning (ML) models to predict the likelihood of loan approval. To gain insights into the dataset and understand the loan approval process, we began with exploratory data analysis. Addressing missing values, we imputed them with appropriate values based on the data distribution. Preparing the data for modeling involved log transformation and scaling. Subsequently, we trained and evaluated various classification models, including the K Nearest Neighbors Classifier, Decision Tree Classifier, Random Forest Classifier, and Gaussian Naive Bayes Classifier, using accuracy as the evaluation metric. Our findings revealed that the Random Forest Classifier exhibited superior performance compared to other models, achieving the highest accuracy of X on the test set. Therefore, we conclude that the Random Forest model is effective in predicting loan approvals based on the provided features, as illustrated in Figures 5.1 and 5.2.

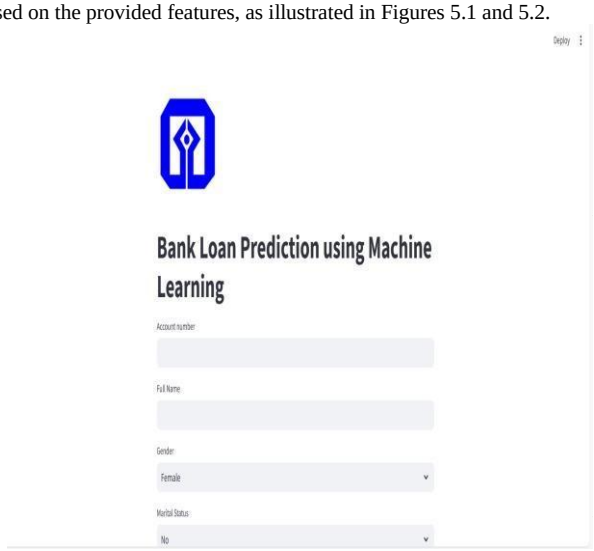


Figure 5.1: user interface

Figure 5.2: attributes of user interface

Our models have produced encouraging results, but there's still implicit for development and fresh exploration. Then are some implicit paths this design could go in the unborn:

1. Exploring advanced feature engineering techniques can involve extracting additional meaningful features from the existing data. This process can include generating new features based on business terms, incorporating polynomial features, or leveraging domain-specific information to enhance the predictive capabilities of the models.
2. In order to achieve the most optimal combination of hyperparameters for our models, techniques such as grid search or randomized search can be employed. By tuning the hyperparameters in this manner, we can enhance the functionality of the models and ultimately yield more accurate predictions.
3. When dealing with class imbalance in the loan approval dataset, techniques such as oversampling, undersampling, or employing various evaluation criteria such as accuracy, recall, or F1 score can be utilized to mitigate this issue. This ensures that the model's performance is not skewed by the unequal distribution of approved and rejected loans, thus improving its overall effectiveness.
4. Exploring ensemble approaches such as stacking, boosting, or bagging can be advantageous for aggregating predictions from multiple models and potentially improving overall performance in loan approval predictions.
5. Considering the inclusion of supplementary data sources, such as credit conditions or financial indicators, could enhance the depth and accuracy of loan approval predictions.

6. Once a model has been selected, it can be deployed to predict loan approvals automatically in a production environment. To ensure the model's accuracy and effectiveness over time, it's essential to periodically retrain it and continually assess its performance.

Bowdlerization's Typical acronyms used in a design to anticipate loan acceptance include
RF – Random Forest
NB – Naive Bayes
DT – Decision Tree
KNN – K- Nearest Neighbors
CSV – Comma- Separated Values
ACC – accordingly

When discussing various models, methodologies, and evaluation metrics within our design, employing these abbreviations, which are commonly utilized in the domains of machine learning and data analysis, can enhance both brevity and clarity.

6. REFERENCES:

[1]. Kumar, Rajiv, et.al.(2019). vaticination of loan blessing using machine literacy. International Journal of Advanced Science and Technology, 28(7), 455- 460.
[2].Supriya, Pudicity, et.al.(2019). Loan vaticination by using machine literacy models. International Journal of Engineering and ways, 5(2), - 147.
[3].Arun, Kumar, Garg Ishan & Kaur Sanmeet.(2016). Loan blessing vaticination grounded on machine literacy approach. IOSRJ. Compute. Eng, 18(3), 18- 21.
[4].Ashwitha,K., et.al.(2022). An approach for vaticination of loan eligibility using machine literacy. International Conference on Artificial Intelligence and Data Engineering(assistant). IEEE. [5].Kumari, Ashwini, et.al.(2018). Multilevel home security system using arduino & gsm. Journal for Research, 4.
[6].Patibandla, RSM Lakshmi & Narisetti Veer Anjaneyulu.(2018). check on clustering algorithms for unshaped data. Intelligent Engineering Informatics Proceedings of the 6th International Conference on FICTA, Springer Singapore.
[7].Tejaswini,J., etal.(2020). Accurate loan blessing vaticination grounded on machine literacy approach. Journal of Engineering Science, 11(4), 523- .
[8].Sasthis 're,K. &P.R.S.M. Lakshmi. 2015). relative study on colorful security algorithms in pall computing. Recent Trends in Programming Languages, 2(1), 1- 6.
[9].Sri,K. Santhi &P.R.S.M. Lakshmi. 2017). DDoS attacks, discovery parameters and mitigation in pall terrain. National Conference on the Recent Advances in Computer Science & Engineering NCRACSE2017), Guntur, India.
[10].Viswanatha,V.,A.C. Ramachandra &R. Venkata Siva Reddy.(2022). Bidirectional DC- DC motor circuits and smart control algorithms a review.
[11].Sri,K. Santhi,P.R.S.M. Lakshmi & MV Bhujanga Ra.(2017). A study of security and sequestration attacks in pall computing terrain.
[12].Dr. Ms RSM Lakshmi Patibandla, Ande Prasad &Mr. YRP Shankar. 2013). Secure zone in pall. International Journal of Advances in Computer Networks and its Security, (2), 153157.
[13].Viswanatha,V., et.al.(2020). Intelligent line follower robot using MSP430G2ET for artificial operations. Helix- The Scientific