CREDISENSE: AI DRIVEN SANCTIONS

A MINI PROJECT REPORT

Submitted by

ARULKUMARAN P
(221801004)

HARISH RAGHAVENDRA R
(221801015)

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

RAJALAKSHMI ENGINEERING COLLEGE
DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

ANNA UNIVERSITY, CHENNAI

NOV 2024

BONAFIDE CERTIFICATE

Certified that this Report titled **"CrediSense: AI Driven Sanctions"** is the Bonafide work of **Arul Kumaran P (2116221801004), Harish Raghavendra R (2116221801015)** who carried out the work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. J. M. GNANASEKAR, M.E., Ph.D., Professor and Head, Department of Artificial Intelligence and Data Science, Rajalakshmi Engineering College, Chennai – 602 105.	**Dr. J. M. GNANASEKAR, M.E., Ph.D.,** Professor and Head, Department of Artificial Intelligence and Data Science, Rajalakshmi Engineering College, Chennai – 602 105.

Submitted for the project viva-voice examination held on………………………………..

INTERNAL EXAMINER EXTERNAL EXAMINER

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavor to put forth this report. Our sincere thanks to our Chairman Mr. S. MEGANATHAN, B.E, F.I.E., our Vice Chairman Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S., and our respected Chairperson Dr. (Mrs.) THANGAM MEGANATHAN, Ph.D., for providing us with the requisite infrastructure and sincere endeavoring in educating us in their premier institution.

Our sincere thanks to Dr. S.N. MURUGESAN, M.E., Ph.D., our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to Dr. J. M. GNANASEKAR., M.E., Ph.D., Head of the Department, Professor and Head of the Department of Artificial Intelligence and Data Science for his guidance and encouragement throughout the project work. We are glad to express our sincere thanks to our supervisor Dr. J. M. GNANASEKAR., M.E., Ph.D., Head of the Department, Dr. P. INDIRA PRIYA, M.E., Ph.D., Professor, Department of Artificial Intelligence and Data Science, Rajalakshmi Engineering College for his valuable guidance throughout the course of the project.

Finally we express our thanks for all teaching, non-teaching, faculty and our parents for helping us with the necessary guidance during the time of our project.

ABSTRACT

In the mortgage lending sector, traditional loan approval processes remain largely manual and time-consuming, leading to significant inefficiencies and prolonged decision-making times. These conventional methods require extensive human involvement, increasing the potential for human error and inconsistencies in loan evaluation. As the demand for more efficient and reliable systems grows, there is a need for technological advancements that can streamline the process and improve accuracy.

An existing automated system attempted to address these challenges by implementing a machine learning-based approach to automate loan approvals. However, this system faced critical issues with accuracy and scalability, limiting its effectiveness and applicability in real-world settings. The low predictive accuracy hindered its reliability, and the lack of scalability prevented it from handling diverse and large volumes of data. These limitations necessitated the development of a more robust solution that could meet the requirements of the mortgage lending industry.

To overcome these challenges, the proposed system employs a stacking classifier—a powerful ensemble method that combines multiple machine learning models. By analyzing financial statement data and credit scores, this model aims to minimize human error and improve decision-making accuracy. Future enhancements will focus on incorporating real-time data analysis and additional financial ratios to increase predictive precision further. These developments are expected to make the loan approval process more efficient, scalable, and adaptable to varying market conditions, offering a promising solution for mortgage lenders.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 GENERAL

In the financial sector, especially in banking, loan defaulting and credit risk are major challenges that can significantly impact profitability and overall financial health. Financial institutions rely heavily on accurate risk assessment to determine whether an applicant is eligible for a loan. Traditional methods for evaluating loan applicants often rely on manual review processes, which are not only time-consuming but also subject to human biases and inconsistencies. With the availability of extensive data and advancements in machine learning, automating this process can lead to better decision-making and more reliable risk assessments.

Loan prediction and risk assessment involve determining whether a borrower is likely to repay the loan based on various financial, demographic, and behavioral factors. Key indicators such as credit score, debt-to-income ratio (DTI), annual income, and credit utilization offer insights into a borrower’s creditworthiness. However, identifying complex patterns across these features to predict loan approval or rejection is challenging. Misjudging an applicant's risk level can lead to granting loans to high-risk individuals, resulting in defaults and financial losses for the institution. Conversely, overly stringent criteria may deny creditworthy applicants, potentially reducing customer satisfaction and market share.

This project addresses the need for a robust, data-driven solution to predict loan approval and assess risk levels for applicants in a more objective, efficient, and scalable way. By leveraging machine learning models such as decision trees, ensemble methods, and stacking classifiers, this system automates the loan evaluation process, aiming to improve both accuracy and efficiency. This project not only forecasts loan approval but also provides a comprehensive risk assessment for applicants, helping financial institutions to make more informed, balanced decisions.

1.2 NEED FOR THE STUDY

The need for a comprehensive study on loan prediction and risk assessment arises from the increasing complexity and volume of data associated with loan applicants and the growing challenges in accurately assessing credit risk. Financial institutions are tasked with making fast, accurate lending decisions, as errors in loan approval processes can have significant financial repercussions. Traditional approaches often rely on manual assessments and basic credit scoring methods that, while useful, may not capture the full risk profile of an applicant. These methods can be limited in predictive power and are subject to human biases, leading to inconsistent results that can impact an institution’s risk exposure.

This study is essential to improve risk management practices by developing more sophisticated loan prediction and risk assessment models. Accurate risk prediction helps lenders identify potentially high-risk applicants, which is crucial to reducing loan defaults and safeguarding financial health. With a robust model in place, financial institutions can make better-informed lending decisions that minimize losses and maximize returns. The study's focus on automating decision-making through machine learning addresses the need for operational efficiency in loan processing. Automation allows institutions to significantly reduce processing times and operational costs, providing applicants with faster, more consistent responses. This efficiency not only enhances the customer experience but also streamlines internal operations, allowing lenders to allocate resources more effectively.

Moreover, this study addresses the issue of financial inclusion. Traditional credit scoring methods often overlook individuals without substantial credit histories, effectively barring them from obtaining loans. By utilizing more sophisticated algorithms and alternative data points, this study aims to improve the assessment of creditworthiness for these underserved groups. This approach could open up lending opportunities for a broader demographic, fostering economic growth and allowing institutions to serve a more diverse customer base.

1.3 OBJECTIVES OF THE STUDY

The primary objective of this study is to develop an advanced, data-driven loan prediction and risk assessment model that can improve decision-making processes in financial institutions. This model aims to leverage machine learning techniques to enhance the accuracy, efficiency, and fairness of loan approval processes. Specifically, the objectives are as follows:

1. **Predict Loan Approval Outcomes**: To build a machine learning model capable of accurately predicting whether a loan application will be approved or rejected based on various applicant features, such as credit score, annual income, debt-to-income ratio, and credit utilization.

2. **Assess and Quantify Risk**: To implement a risk assessment component that evaluates the likelihood of loan default and categorizes applicants by risk level (e.g., low, medium, high). This will help financial institutions better understand the potential risk associated with each applicant.

3. **Optimize Lending Decisions**: To design a recommendation system that can calculate an optimal loan amount for approved applicants based on their financial profile, thereby supporting financial institutions in making balanced and informed lending decisions that maximize profitability while minimizing risk.

4. **Enhance Operational Efficiency**: To automate the loan evaluation process, reducing the time and resources required for manual loan assessments. This objective aligns with the goal of increasing processing speed and improving customer experience through quick decision-making.

5. **Support Financial Inclusion**: To assess creditworthiness in a way that considers applicants who may lack a traditional credit history, thereby expanding lending opportunities for underserved populations. This objective aligns with the broader goal of making credit more accessible.

6. **Ensure Compliance and Transparency**: To develop a model that aligns with regulatory standards, ensuring that the loan approval and risk assessment

processes are transparent, fair, and consistent. This objective supports financial institutions' need to meet compliance requirements while maintaining customer trust.

In summary, this study aims to create a reliable, scalable, and ethically sound loan prediction and risk assessment system that addresses key challenges in credit evaluation, risk management, and regulatory compliance. Through the integration of machine learning, the study seeks to contribute to a more robust and inclusive financial system that benefits both lenders and borrowers.

1.4 OVERVIEW OF THE PROJECT

The system is designed to streamline loan processing for financial institutions. It utilizes machine learning to evaluate loan applications based on various financial metrics and provide insights into the approval likelihood, risk level, and recommended loan amount for applicants.

Key Features:

1. **Loan Approval Prediction:**

- **Objective**: To determine if a loan application should be approved based on key financial indicators.
- **Methodology**: The project employs a **stacking ensemble model**, combining Decision Tree and XGBoost classifiers, with Logistic Regression as the final estimator. This ensemble approach enhances prediction accuracy by leveraging the strengths of different algorithms.
- **Outcome**: The system classifies loan applications as either "Approved" or "Rejected," providing a quick and data-driven initial assessment.

2. **Risk Assessment:**

- **Objective**: To assess the risk level associated with a loan applicant, helping loan officers make informed decisions.

- **Methodology**: The project calculates a **risk score** based on the applicant's credit history, income stability, DTI ratio, and other relevant features. These scores are categorized into three risk levels: **Low, Medium, and High**.
- **Outcome**: This component aids lenders in identifying high-risk applicants, minimizing default rates by allowing careful evaluation of risk-prone applicants.

3. **Loan Recommendation System:**

- **Objective**: To provide a recommended loan amount tailored to the applicant's financial profile, thereby aligning the loan offering with the applicant's repayment capability.
- **Methodology**: This feature computes a loan amount based on the applicant's annual income, credit score, and outstanding loan amount, adjusting recommendations according to risk factors.
- **Outcome**: By setting loan amounts aligned with the applicant's capacity, the system helps reduce financial strain on borrowers and mitigates risk for lenders.

Technical Components:

- **Data Preprocessing**: The project standardizes and encodes input features to ensure compatibility with machine learning algorithms.
- **Feature Engineering**: Key features (e.g., DTI ratio, FICO score, credit utilization) are selected based on their relevance to loan approval and risk assessment.
- **Model Training and Evaluation**: The stacking model is trained on historical loan data, with metrics such as confusion matrix, classification report, and ROC AUC score used to evaluate performance.
- **User Input and Interface**: Functions are provided for users to input their details, enabling real-time prediction, risk evaluation, and loan recommendations.

CHAPTER 2

REVIEW OF LITERATURE

2.1 INTRODUCTION

The review of literature for this project delves into existing studies and methodologies surrounding automated loan processing, risk assessment, and recommendation systems within financial services. The financial industry has increasingly adopted machine learning and predictive analytics to improve decision-making and reduce risk. Research shows that traditional rule-based systems, though widely used, are often limited in handling complex, non-linear relationships within financial data, leading to potential inaccuracies in loan approval and risk evaluation.

Several studies have explored the use of machine learning models, such as decision trees, gradient boosting algorithms, and logistic regression, in predicting loan approval outcomes. These models have demonstrated their capability to analyze extensive datasets and detect patterns in borrowers' credit history, income, debt levels, and other key metrics, which are critical in evaluating creditworthiness. Furthermore, ensemble methods, particularly stacking models, have been shown to improve predictive accuracy by combining multiple algorithms, thereby overcoming the limitations of individual classifiers.

This literature review underpins the system– 's approach of using a stacking ensemble model for loan approval prediction, a risk assessment system for default probability categorization, and a recommendation engine for personalized loan amounts. Collectively, these insights validate the proposed solution, emphasizing the value of machine learning in optimizing loan processing efficiency and accuracy in the financial industry

S. No	Author Name	Paper Title	Description	Journal	Volume/ Year
1	Y. Diwate, P. Rana, and P. Chavan	Loan Approval Prediction Using Machine Learning	This paper discusses methods for predicting loan approval outcomes using machine learning, highlighting key financial indicators and model accuracy improvements.	IRJET	Vol.8,2021
2	N. Uddin, M. K. Uddin Ahamed, M. A. Uddin, M. M. Islam, M. A. Talukder, and S. Aryal	An Ensemble Machine Learning Based Bank Loan Approval Predictions System with a Smart Application	This study proposes an ensemble machine learning approach for bank loan approval prediction, incorporating a smart application to streamline decision-making.	International Journal of Cognitive Computing in Engineering	Vol.4,2022
3	Amruta S. Aphale	Predict Loan Approval in Banking System Machine Learning	Focuses on a machine learning approach tailored to cooperative banks, aiming to improve the accuracy and	International Journal of Engineering Research & Technology	Vol. 9,2020

		Approach for Cooperative Banks Loan Approval	efficiency of loan approval processes.		

Table no 1 Literature Review

2.2 LITERATURE REVIEW

The literature on loan approval prediction and risk assessment highlights the transformative impact of machine learning in the financial sector. Traditional lending models, which often rely on manual evaluations of financial indicators, are increasingly being replaced by automated, data-driven approaches that improve both efficiency and accuracy. Diwate, Rana, and Chavan (2021) showcase how models like Decision Trees and Logistic Regression are employed to predict loan approvals by analyzing critical financial metrics such as credit score, income, and debt-to-income (DTI) ratio, emphasizing the importance of preprocessing and feature selection. Similarly, Uddin et al. (2023) demonstrate the advantages of ensemble learning models in bank loan predictions, particularly in reducing misclassification rates and enhancing decision-making.

Amruta S. Aphale (2020) further explores this field by focusing on cooperative banks, where her study highlights the challenges and benefits of using machine learning models specifically tailored to smaller financial institutions. Her research suggests that these models not only improve the accuracy of loan approvals but also align loan recommendations with applicants' financial stability, enhancing lending fairness and reducing default rates. Collectively, the literature underscores a strong shift towards data-driven loan processing, advocating for machine learning solutions that balance prediction accuracy, speed, and scalability, which are essential for modern banking environments.

CHAPTER 3

SYSTEM OVERVIEW

3.1 EXISTING SYSTEM

Existing loan approval systems in many financial institutions often rely on traditional, rule-based processes combined with manual evaluations of an applicant's financial history. These systems primarily assess eligibility based on fixed thresholds, such as minimum credit scores, income levels, and debt-to-income (DTI) ratios. While these criteria provide a basic measure of creditworthiness, the rigid nature of rule-based assessments lacks flexibility in adapting to complex financial profiles. Additionally, manual evaluations are prone to subjectivity and inconsistencies, leading to potential biases and varying outcomes across loan officers. Another drawback is that these systems are often time-consuming, as they require multiple layers of review and verification, which can delay loan processing and decision-making.

Moreover, traditional methods tend to overlook deeper patterns within an applicant's data, limiting the system's ability to accurately predict default risk. Since these systems typically lack sophisticated predictive capabilities, they may either approve risky loans or reject qualified applicants, impacting both the institution's profitability and customer satisfaction. These limitations underscore the need for a more advanced, data-driven approach that leverages machine learning to improve accuracy, consistency, and efficiency in loan approvals.

Furthermore, existing systems often struggle with scalability and adaptability to changing economic conditions or borrower profiles. With the rapid growth of customer data and the dynamic nature of financial markets, rule-based systems are ill-equipped to handle large volumes of data or incorporate new variables that may emerge as important indicators of credit risk. This rigidity can result in outdated risk assessments that fail to capture trends like shifts in consumer behavior or the impact of economic downturns on borrowers' repayment capacities. In addition, these systems rarely provide personalized recommendations for loan amounts, which can lead to offering loans that are either too high, increasing the risk of default, or too low, affecting customer satisfaction.

Another significant drawback is the lack of an integrated risk assessment mechanism. While credit scores provide a general sense of risk, they do not always offer a comprehensive picture of an applicant's overall financial health. Traditional systems rarely combine multiple risk factors or allow for dynamic risk scoring, which limits their effectiveness in preemptively identifying high-risk borrowers. These limitations create an urgent need for modernized solutions that can enhance predictive accuracy, incorporate real-time data, and provide individualized loan and risk assessments to support smarter, more reliable lending practices.

3.2 PROPSED SYSTEM

The proposed system is a robust and integrated **Loan Approval and Risk Assessment Application**, specifically designed to streamline the loan application process and provide an accurate evaluation of financial risks. This system leverages the power of advanced machine learning algorithms to ensure reliable predictions while offering a user-friendly interface for seamless interaction.

The frontend interface, built using HTML and CSS, provides an intuitive and responsive design tailored for both loan applicants and financial institutions. It features two main sections: one dedicated to loan approval prediction and another for risk assessment. The loan approval section allows users to enter essential details such as FICO score, debt-to-income (DTI) ratio, annual income, revolving utilization, and other relevant financial parameters. The risk assessment section focuses on evaluating the financial standing and potential risks associated with the applicant. The interface ensures a professional and engaging user experience, with clearly structured forms, dynamic result displays, and aesthetically pleasing visuals.

The backend is powered by Flask, a lightweight yet powerful Python web framework, designed to handle requests efficiently. It includes APIs for loan approval prediction and dynamic risk evaluation. The backend processes user inputs by applying pre-trained machine learning models, including a stacking classifier ensemble that combines Decision Tree, XGBoost, and Logistic Regression. The data is preprocessed using techniques like feature scaling and encoding to ensure compatibility with the model. Predictions are generated in real-time, classifying applicants into categories such as approved or rejected for loans and low, medium, or high-risk for financial evaluations. Error handling mechanisms ensure robustness and provide meaningful feedback for invalid or incomplete inputs.

Overall, this system addresses key challenges in the loan processing domain by automating approval predictions and risk assessments, minimizing manual effort, and improving accuracy. Its modular architecture ensures scalability, allowing additional features or datasets to be integrated in the future. By combining cutting-edge machine learning with a professional, user-centric interface, the proposed system aims to enhance decision-making processes for financial institutions and provide transparency and confidence to applicants.

3.3 FEASIBILITY STUDY

The system leverages existing technologies and tools, making it technically feasible to develop and implement. The backend is built using Flask, a lightweight Python web framework known for its efficiency in handling web APIs. The machine learning models, including the stacking classifier, utilize robust libraries such as scikit-learn and XGBoost, which are widely used for predictive analytics. For the frontend, HTML and CSS are employed to create an intuitive interface, ensuring compatibility with modern web browsers. The system's infrastructure is adaptable to cloud hosting solutions, providing scalability for handling large volumes of data and user requests.

1. Technical Feasibility: The system leverages existing technologies and tools, making it technically feasible to develop and implement. The backend is built using Flask, a lightweight Python web framework known for its efficiency in handling web APIs. The machine learning models, including the stacking classifier, utilize robust libraries such as scikit-learn and XGBoost, which are widely used for predictive analytics. For the frontend, HTML and CSS are employed to create an intuitive interface, ensuring compatibility with modern web browsers. The system's infrastructure is adaptable to cloud hosting solutions, providing scalability for handling large volumes of data and user requests.

2. Operational Feasibility: The proposed system simplifies loan processing and risk evaluation, making it operationally efficient for financial institutions and applicants. By automating the approval and risk assessment processes, it reduces manual effort, increases accuracy, and minimizes decision-making time. The user-friendly interface ensures that both technical and non-technical users can interact with the system effortlessly. Furthermore, the system's modular design allows easy integration with existing financial workflows, enhancing its practical utility.

3. Economic Feasibility: The cost of development is justified by the benefits the system provides. Open-source tools such as Flask, scikit-learn, and HTML/CSS minimize development costs, while the integration of machine learning reduces reliance on manual evaluations, lowering operational costs for financial institutions. The system’s predictive accuracy and risk management capabilities help mitigate potential loan defaults, saving significant amounts in the long term. Additionally, the system can be monetized by offering it as a service to banks or financial organizations, ensuring a favourable return on investment.

CHAPTER 4

SYSTEM REQUIREMENTS

4.1 SOFTWARE REQUIREMENT

1. Operating System: Windows 10/11

2. Programming Languages:

Python3: Python is required for developing the core functionality, including machine learning models and video processing. Python libraries such as TensorFlow, Keras, and Sklearn will be used.

3. Web Development:

Flask: Flask, a lightweight Python web framework, is used for building the backend of the web application. It handles routing, form submissions, and communication between the frontend and backend.

HTML5, CSS3, and JavaScript: HTML is essential for structuring the web page, while CSS provides styling to ensure a professional user interface. JavaScript for client-side interactivity and API request handling.

4. Machine Learning Libraries:

TensorFlow/Keras: These deep learning/machine learning libraries are essential for training and running the human action recognition model that detects student engagement and behavior in the videos.

Google Colab: For interactive data exploration, preprocessing, and model training.

5. Data Processing: NumPy and Pandas These libraries are essential for handling and processing numerical data, such as managing predictions and preparing data for visualizations.

6. Visualization: Matplotlib these libraries are used for generating visualizations like pie charts, bar graphs, or other analytics to represent student engagement patterns.

CHAPTER 5

SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE

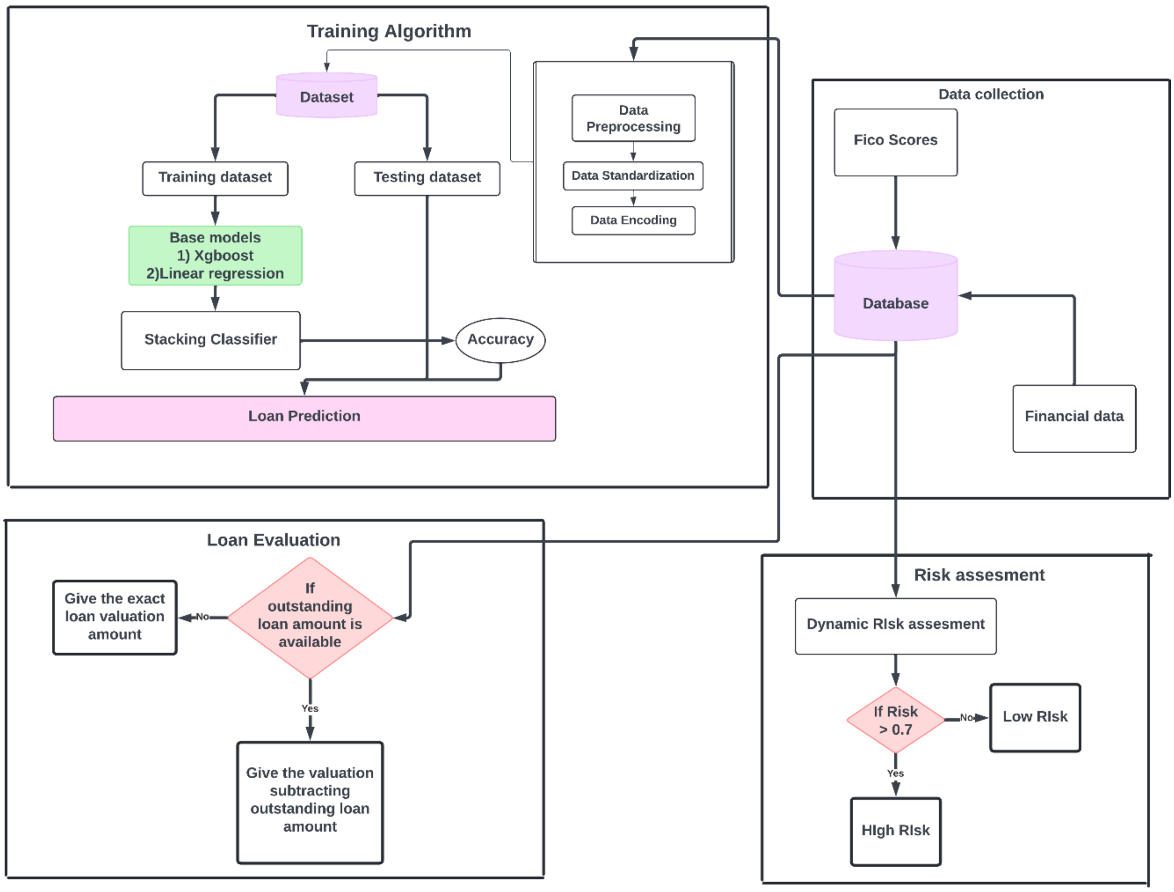

Fig 5.1 System Architecture

The system architecture for the loan prediction and risk assessment project comprises three primary sections: Training Algorithm, Loan Evaluation, and Risk Assessment. It starts with a dataset divided into a training dataset and a testing dataset, followed by data preprocessing which involves standardization and encoding. Two base models, Xgboost and Linear regression, are then utilized. These base models feed into a stacking classifier, which is evaluated for accuracy, resulting in loan predictions. For loan evaluation, a decision node determines if the outstanding loan amount is available; if so, the system provides the valuation after subtracting the outstanding loan amount; otherwise, it gives the exact loan valuation amount. The risk assessment involves a database containing FICO scores and financial data, with a dynamic risk assessment that continuously evaluates the risk level. A decision node classifies the risk, deeming it high risk if the level is greater than 0.7, and low risk if it is not. This structured approach ensures accurate loan predictions and effective risk management, essential for financial decision-making systems.

5.2 MODULE DESCRIPTION

5.2.1 DATA COLLECTION AND PREPROCESSING MODULE:

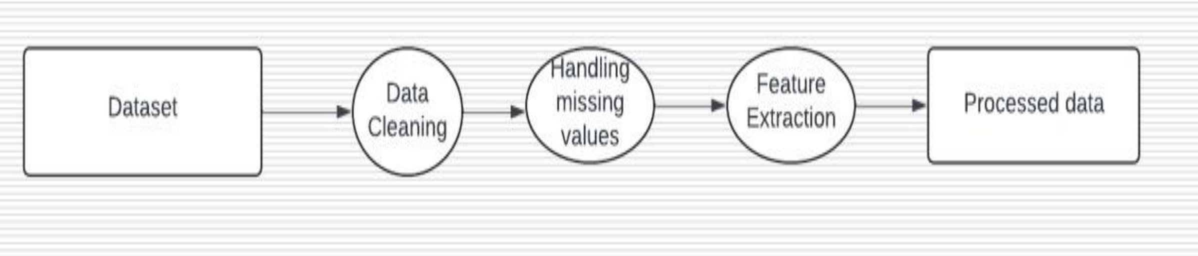

Fig 5.2 Data collection and Pre-processing

Data Collection:

Data collection involves gathering relevant financial information about loan applicants. This includes structured data, often sourced from historical loan datasets, financial records, or other trusted repositories.

- **Source of Data**:
    1. Historical loan datasets containing details like FICO scores, debt-to-income ratios, annual income, and loan outcomes (approved/rejected).
    2. Data from publicly available financial datasets or private financial institution records.
    3. Simulated data for training and testing purposes if real-world data is unavailable.
- **Data Attributes**: Key features extracted from the dataset include:

1. **Numerical Features**: Debt-to-income ratio (DTI), annual income, revolving balance, revolving utilization, days with credit line, FICO score.
2. **Categorical Features**: Loan purpose, credit policy, public records.
3. **Target Variable**: Loan approval status (binary: 0 for rejection, 1 for approval).

- **Tools for Collection**:
    1. pandas: To load and explore datasets, usually in .csv or database formats.

Data Preprocessing:

Data preprocessing involves transforming raw data into a format suitable for analysis. This step eliminates inconsistencies, handles missing values, and prepares features for machine learning models.

Steps in Preprocessing:

1. **Handling Missing Data**:
    a) Imputation techniques are applied to fill missing values.
    b) For numerical features, the mean or median value is used.
    c) For categorical features, the mode or a placeholder value is substituted.
2. **Encoding Categorical Variables**:
    a) LabelEncoder is used to convert categorical values (e.g., loan purpose) into numerical representations for model compatibility.
3. **Feature Scaling**:
    a) StandardScaler is employed to standardize numerical features like DTI and annual income to a uniform scale, improving model convergence and performance.

4. **Feature Selection**:

    a) Correlation analysis is conducted to identify the most impactful features.

    b) Features with low importance or high multicollinearity are excluded.

5. **Data Splitting**:

    a) The dataset is split into training and testing sets (e.g., 70% for training and 30% for testing) using train_test_split to evaluate model performance on unseen data.

6. **Outlier Detection and Removal**:

    a) Statistical methods (e.g., Z-scores) or visualizations (e.g., boxplots) are used to detect and handle outliers that could skew model results.

5.2.2 TRAINING AND TESTING MODULE:

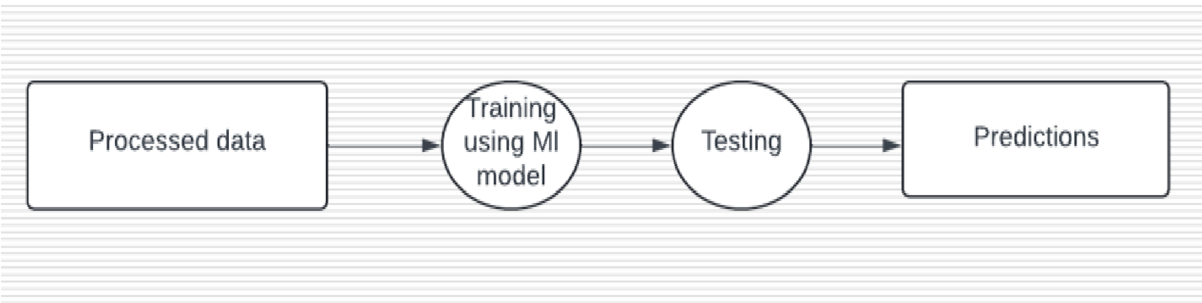

Fig 5.3 Training and Testing Module

The **Training and Testing Module** is a pivotal part of the Loan Approval and Risk Assessment system. This module is responsible for building, optimizing, and evaluating machine learning models to predict loan approval and assess financial risks. It ensures the system's accuracy, reliability, and robustness by training on historical data and validating against unseen data.

Training Phase:

1. **Dataset Preparation:**

    a) Preprocessed data is split into training and testing datasets, typically in a ratio of 70:30.

    b) The training dataset is used to train the machine learning models, while the testing dataset evaluates the model's performance.

    c) Feature scaling (using StandardScaler) ensures numerical stability and faster convergence during training.

2. **Model Selection:**

    a) A stacking classifier is employed, combining multiple base models for improved predictive accuracy.

    b) **Base Models:**

        1. Decision Tree Classifier: Captures simple decision-making rules.

        2. XGBoost Classifier: An efficient gradient boosting algorithm for handling non-linear relationships.

    c) **Meta Model:**

        1. Logistic Regression: Aggregates predictions from the base models to make the final decision.

3. **Training Process:**

    a) Each base model is trained on the training dataset to learn specific patterns in the data.

    b) Predictions from the base models are passed to the meta model for final training.

    c) Hyperparameter tuning (using techniques like grid search or randomized search) is conducted to optimize model performance.

4. **Tools and Libraries:**
    a) scikit-learn: For building and training the stacking classifier.
    b) XGBoost: For gradient boosting implementation.
    c) pandas and numpy: For data manipulation during the training process.

Testing Phase:

1. **Evaluation Metrics**: The model's performance is assessed using the following metrics:
    a) **Confusion Matrix**: To evaluate the distribution of true positives, true negatives, false positives, and false negatives.
    b) **Classification Report**: Provides precision, recall, and F1-score for each class (approved/rejected).
    c) **ROC AUC Score**: Measures the model's ability to distinguish between the positive and negative classes.
2. **Validation Process**:
    a) The trained models are applied to the testing dataset (data unseen during training) to validate their generalization ability.
    b) Predictions are compared against actual outcomes to compute the evaluation metrics.
3. **Overfitting Prevention**:
    a) Techniques like cross-validation are used to ensure the model does not overfit the training data.
    b) Feature selection and regularization are applied to improve model robustness.

Output of Model:

1. **Trained Model**:
    a) The stacking classifier is saved as a serialized file (e.g., .pkl) for deployment.

b) The scaling and encoding objects (e.g., StandardScaler, LabelEncoder) are also saved to ensure consistency during prediction.

2. **Model Performance Summary**:

a) The confusion matrix and classification report provide insights into areas where the model performs well and where it needs improvement.

b) The ROC AUC score indicates the model's reliability and predictive power.

5.2.3 LOAN VALUATION OR RECOMMENDATION MODULE:

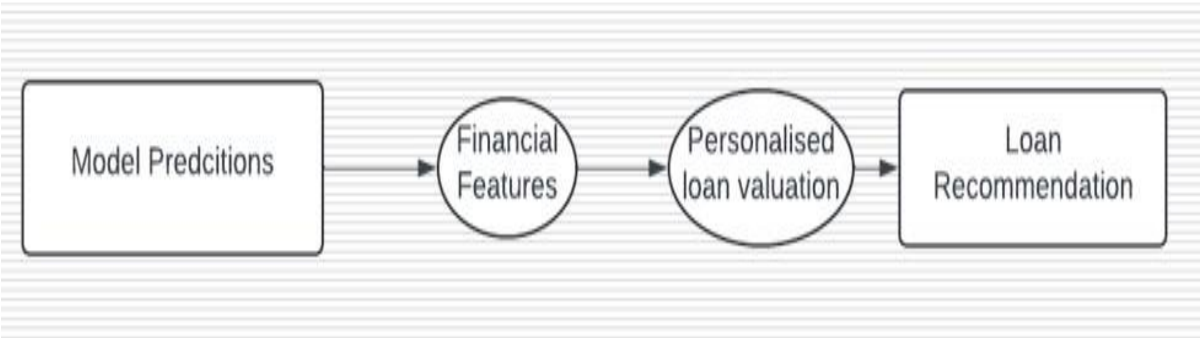

Fig 5.4 Loan Valuation or Recommendation Module

The **Loan Valuation Model** is an integral component of the Loan Approval and Risk Assessment system, designed to evaluate and estimate the potential loan amount that can be approved for an applicant. It provides insights into financial eligibility based on key factors such as income, credit score, and outstanding debts. This model ensures that loans are sanctioned responsibly, minimizing financial risks for both lenders and borrowers.

1. **Purpose of the Model:**

The Loan Valuation Model aims to:

- Determine the maximum loan amount that an applicant can responsibly repay.
- Incorporate financial parameters to assess creditworthiness.
- Provide lenders with a data-driven approach to loan allocation.

2. **Key Features of the Model:**

Financial Assessment: Considers income, credit score, and existing debts to make recommendations.

Dynamic Loan Adjustment: Adjusts the recommended loan amount based on changing factors like credit history or additional income.

Risk Mitigation: Reduces the chances of defaults by evaluating loan affordability rigorously.

3. **Inputs to the Model**

The model requires the following inputs to calculate the recommended loan amount:

1. **Annual Income**: The applicant's total income over a year, which serves as a primary indicator of repayment capacity.
2. **Credit Score**: A numerical value reflecting the applicant's creditworthiness. Higher scores generally allow for larger loan amounts.
3. **Outstanding Loan Amount**: Existing debts that need to be accounted for to avoid over-leveraging.
4. **Debt-to-Income Ratio (DTI)**: A ratio indicating the percentage of income already allocated to debt payments.
5. **Revolving Utilization**: The proportion of available credit currently being used by the applicant.

4. **Calculation of the Model:**

The Loan Valuation Model employs a systematic approach to compute the recommended loan amount:

1. **Base Loan Amount Calculation**:
    - Derived as a percentage of the applicant's annual income. Typically, 30-40% of the income is considered as a safe loan amount.
    - Formula: Base Loan Amount = Annual Income × Income Factor

2. **Credit Score Adjustment**:
    - Adjusts the base loan amount based on the applicant's credit score. For example:
        - Scores below 600: Reduction factor of 0.5.
        - Scores between 600–700: Adjustment factor of 0.8.
        - Scores between 700–800: Adjustment factor of 1.0.
        - Scores above 800: Boost factor of 1.2.
3. **Debt Adjustment**:
    - Subtracts any outstanding loan amounts from the adjusted loan amount to ensure repayment feasibility.
    - Formula: Final Loan Amount = (Base loan amount x Credit Factor) – Outstanding loan amount.
    - Ensures the final loan amount is non-negative, setting it to zero if debts exceed the adjusted eligibility.
4. **Recommendation**:
    - The calculated amount is presented to lenders as the recommended loan cap for the applicant.

5.2.4 RISK ASSESSMENT MODULE:

The Risk Assessment Module is a crucial component of the Loan Approval and Risk Assessment system. It is designed to evaluate the financial risk associated with each loan application. By analysing applicant data and predicting potential default risks, the module enables financial institutions to make informed lending decisions while minimizing exposure to bad loans.

1. Purpose of the Module

The primary objectives of the Risk Assessment Module are:

- To classify applicants into risk categories (e.g., low, medium, high).
- To provide a quantitative score that reflects the likelihood of default.

- To support lenders in assessing the financial stability and repayment capability of applicants.

2. Features of the Module

Dynamic Risk Scoring: Calculates a risk score based on real-time input data and trained machine learning models.

Categorical Risk Levels: Classifies applicants into risk bands (e.g., low risk for scores below 0.3, medium risk for scores between 0.3 and 0.7, and high risk for scores above 0.7).

Holistic Evaluation: Considers multiple factors, including financial stability, credit history, and current liabilities, for comprehensive risk analysis.

3. Inputs to the Module

The module evaluates risk using the following applicant data:

1. **Credit Policy Compliance:** Indicates whether the applicant meets the financial institution's predefined credit standards.
2. **Loan Purpose:** Categorical data reflecting the intended use of the loan (e.g., debt consolidation, credit card, or other purposes).
3. **Interest Rate:** The annual percentage rate charged on the loan, impacting affordability.
4. **Installment Amount:** The periodic repayment amount, used to assess financial burden.
5. **Log Annual Income:** The logarithm of the applicant’s annual income, reflecting financial stability.
6. **Debt-to-Income Ratio (DTI):** Indicates the percentage of income already allocated to debt payments.
7. **FICO Score:** Measures the applicant’s creditworthiness.
8. **Days with Credit Line:** Reflects the duration of the applicant’s credit history.

9. **Revolving Balance:** Represents the total credit card balance owed.
10. **Revolving Utilization:** The percentage of available credit currently in use.
11. **Inquiries in Last 6 Months:** Shows the number of recent credit checks, potentially indicating financial strain.
12. **Public Records:** Details any derogatory marks such as bankruptcies or liens.

CHAPTER 6

RESULT AND DISCUSSION

6.1 Result and Discussion

The system uses machine learning models to predict loan approval and assess the associated risk. The system leverages a Stacking Classifier, which combines the strengths of multiple machine learning algorithms (including DecisionTreeClassifier, XGBClassifier, and LogisticRegression) to provide accurate predictions.

The key outcomes of the project include:

1. **Loan Valuation:** The system calculates a recommended loan amount based on the applicant's annual salary, credit score, and outstanding loan amount. This recommendation helps ensure that the loan amount is both reasonable and manageable for the applicant, reducing the risk for lenders.

2. **Loan Prediction:** The system predicts whether an applicant will likely default on the loan, based on financial factors such as debt-to-income ratio (DTI), revolving balance, credit score, inquiries in the last 6 months, and other features. The model provides a prediction result indicating whether the loan should be approved or rejected.

3. **Risk Assessment:** The system also assesses the risk associated with a loan by evaluating factors like credit policy, interest rate, installment amount, FICO score, and revolving utilization. Based on these factors, the system generates a risk score and categorizes the risk into different levels, such as low, medium, or high. This helps lenders make informed decisions on loan approval.

The system provides both visual feedback (such as the recommended loan amount or risk score) and clear recommendations, making it easier for users to understand the decisions behind their loan predictions and assessments.

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENT

7.1 CONCLUSION

The **Credisense** system offers a comprehensive solution for loan prediction, valuation, and risk assessment. By utilizing a **stacking classifier** with models like **XGBClassifier**, **DecisionTreeClassifier**, and **LogisticRegression**, the system successfully predicts loan outcomes and assesses the associated risk based on key financial indicators such as **credit score**, **debt-to-income ratio**, and **revolving balance**.

The system enables lenders to make informed decisions by providing an accurate prediction of loan approval chances and an evaluation of potential risk. The loan valuation feature offers personalized recommendations for the loan amount, while the risk assessment categorizes borrowers based on their financial behavior, helping to identify high-risk applicants.

Overall, the system demonstrates the effectiveness of using machine learning for financial decision-making, with the potential to streamline the loan approval process and improve risk management in the banking sector.

Despite its promising capabilities, the system's success depends on the quality of the data used for training and its periodic updates to adapt to changing market conditions. As the financial landscape evolves, continuous improvements to the model will be essential to maintain its relevance and accuracy.

7.2 FUTURE ENHANCEMENT:

1. **Improved Model Performance**:

a) Incorporating additional machine learning models (e.g., **Random Forest**, **Neural Networks**) and exploring advanced techniques like **ensemble learning** can enhance the accuracy of predictions and risk assessments.

b) Experimenting with **deep learning** models could uncover more complex patterns in the data and improve performance on large-scale datasets.

2. **Incorporating More Features**:

a) Expanding the dataset to include more relevant features like **employment history**, **geographic location**, and **loan history** would make the predictions and recommendations more accurate.

b) Integration of **alternative data** (e.g., utility bills, social behavior patterns) can help in assessing applicants with limited credit histories.

3. **Real-time Data Integration**:

a) Enabling the system to process **real-time data** would make predictions more dynamic and timely. For example, updating the system with real-time changes in credit scores or loan repayment histories could improve the system's adaptability.

4. **Enhanced User Interface**:

a) The current system can be enhanced with a **more user-friendly interface**, providing interactive features like dashboards for applicants and lenders, real-time loan tracking, and easy-to-read reports.

b) A **mobile app** or a **web-based platform** with notifications and updates would improve user experience and accessibility.

5. **Integration with Credit Scoring Agencies**:

a) Integrating with credit scoring agencies and other financial institutions could allow the system to automatically update an applicant's financial data, ensuring accuracy and improving the speed of loan approval.

6. **Ethical and Fair Lending**:

a) Ensuring **bias-free predictions** and transparent decision-making processes is crucial. Future updates should include regular audits to detect any biases related to gender, race, or socioeconomic status in the models.

b) Implementing **explainable AI** techniques could help applicants understand the reasoning behind loan decisions, fostering trust in automated financial systems.

7. **Data Privacy and Security**:

a) Enhancing security measures to protect sensitive financial and personal data is essential. The system should comply with privacy regulations (e.g., **GDPR**, **CCPA**) and ensure data encryption, especially when dealing with applicant data.

APPENDIX

A1.1 Sample code of model building:

```
# Split the dataset
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=42)


# Standardize the features
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)


# Define base models
decision_tree = DecisionTreeClassifier()
xgboost = XGBClassifier(use_label_encoder=False, eval_metric='logloss')


# Create stacking classifier
stacking_clf = StackingClassifier(estimators=[
    ('decision_tree', decision_tree),
    ('xgboost', xgboost)
    ], final_estimator=LogisticRegression())


# Train the stacking classifier
stacking_clf.fit(X_train, y_train)
```

Output of the Model Building:

```
Epoch 11/30
335/335 ━━━━━━━━━━━━━━━━━━━━ 476s 1s/step - accuracy: 0.2394 - loss: 2.3384 - val_accuracy: 0.2344 - val_loss: 2.3403 - learning_rate: 1.0000e-04
Epoch 12/30
335/335 ━━━━━━━━━━━━━━━━━━━━ 468s 1s/step - accuracy: 0.2432 - loss: 2.3389 - val_accuracy: 0.2439 - val_loss: 2.3205 - learning_rate: 1.0000e-04
Epoch 13/30
335/335 ━━━━━━━━━━━━━━━━━━━━ 474s 1s/step - accuracy: 0.2482 - loss: 2.3195 - val_accuracy: 0.2545 - val_loss: 2.3064 - learning_rate: 1.0000e-04
...
Epoch 29/30
335/335 ━━━━━━━━━━━━━━━━━━━━ 416s 1s/step - accuracy: 0.2954 - loss: 2.1881 - val_accuracy: 0.3037 - val_loss: 2.1817 - learning_rate: 2.0000e-05
Epoch 30/30
335/335 ━━━━━━━━━━━━━━━━━━━━ 416s 1s/step - accuracy: 0.2971 - loss: 2.1810 - val_accuracy: 0.3011 - val_loss: 2.1793 - learning_rate: 2.0000e-05
```

Fig A.1 Model Building

A1.2 Stacking Classifier Sample code:

```
y_pred = stacking_clf.predict(X_test)


print("Confusion Matrix:")

print(confusion_matrix(y_test, y_pred))


print("\nClassification Report:")

print(classification_report(y_test, y_pred))


print("\nROC AUC Score:")

print(roc_auc_score(y_test, y_pred))
```

Output of Stacking Classifier:

```
Confusion Matrix:
[[   0  466]
 [   0 2408]]

Classification Report:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00       466
           1       0.84      1.00      0.91      2408

    accuracy                           0.84      2874
   macro avg       0.42      0.50      0.46      2874
weighted avg       0.70      0.84      0.76      2874


ROC AUC Score:
0.5
```

Fig A.2 Classification Report

A1.3 Data collection and Pre-processing sample code:

 # Load the dataset

file_path = '/clean.csv' # Adjust the path to your CSV file

data = pd.read_csv(file_path)

 # Encode categorical variables

le = LabelEncoder()

data['purpose'] = le.fit_transform(data['purpose'])

important_features = [

'dti', 'annual income', 'revol.bal', 'revol.util', 'days.with.cr.line', 'fico', 'pub.rec', 'inq.last.6mths'

```
]

X = data[important_features]

y = data['Approval']
```

Output of the Data analysis:

	credit.policy	int_rate	installment	log.annual.inc	dti	fico	days.with.cr.line	revol.bal	revol.util	inq.last.6mths	delinq.2yrs
count	9578.000000	9578.000000	9578.000000	9578.000000	9578.000000	9578.000000	9578.000000	9.578000e+03	9578.000000	9578.000000	9578.000000
mean	0.804970	0.122640	319.089413	10.932117	12.606679	710.846314	4560.767197	1.691396e+04	46.799236	1.577469	0.163708
std	0.396245	0.026847	207.071301	0.614813	6.883970	37.970537	2496.930377	3.375619e+04	29.014417	2.200245	0.546215
min	0.000000	0.060000	15.670000	7.547502	0.000000	612.000000	178.958333	0.000000e+00	0.000000	0.000000	0.000000
25%	1.000000	0.103900	163.770000	10.558414	7.212500	682.000000	2820.000000	3.187000e+03	22.600000	0.000000	0.000000
50%	1.000000	0.122100	268.950000	10.928884	12.665000	707.000000	4139.958333	8.596000e+03	46.300000	1.000000	0.000000
75%	1.000000	0.140700	432.762500	11.291293	17.950000	737.000000	5730.000000	1.824950e+04	70.900000	2.000000	0.000000
max	1.000000	0.216400	940.140000	14.528354	29.960000	827.000000	17639.958330	1.207359e+06	119.000000	33.000000	13.000000

Fig A.3 EDA

A1.4 Web Design sample code:

HTML:

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Loan Application</title>

    <link rel="stylesheet" type="text/css" href="{{ url_for('static',
filename='styles.css') }}">

</head>
```

```
<body style="font-family: Arial, sans-serif; margin: 0; background-color: #f4f4f4;
color: #333;">
    <header style="background-color: #004d40; color: #ffffff; padding: 20px; text-
align: center;">
        <h1>CrediSense: AI Driven Sanctions</h1>
        <nav>
            <ul style="list-style: none; padding: 0; display: flex; justify-content: center;">
                <li style="margin: 0 15px;"><a href="{{ url_for('loan_prediction') }}"
style="color: #ffffff; text-decoration: none; font-weight: bold;">Loan
Prediction</a></li>
                <li style="margin: 0 15px;"><a href="{{ url_for('loan_valuation') }}"
style="color: #ffffff; text-decoration: none; font-weight: bold;">Loan
Valuation</a></li>
                <li style="margin: 0 15px;"><a href="{{ url_for('risk_assessment') }}"
style="color: #ffffff; text-decoration: none; font-weight: bold;">Risk
Assessment</a></li>
            </ul>
        </nav>
    </header>
    <main style="max-width: 1200px; margin: 20px auto; padding: 20px; background-
color: #ffffff; border-radius: 8px; box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1); text-align:
center;">
        <h2>Welcome to the Loan Application System</h2>
        <p>Select a feature from the menu to get started.</p>


        <div style="display: flex; justify-content: space-around; margin-top: 20px;">
```

```
            <div style="background-color: #e0f7fa; padding: 20px; border-radius: 8px; box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1); width: 30%;">
                <h3>Loan Prediction</h3>
                <p>Predict your loan eligibility based on various parameters.</p>
                <a href="{{ url_for('loan_prediction') }}" style="display: inline-block; padding: 10px 15px; background-color: #00796b; color: #ffffff; border-radius: 5px; text-decoration: none; transition: background-color 0.3s;">Get Started</a>
            </div>
            <div style="background-color: #e0f7fa; padding: 20px; border-radius: 8px; box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1); width: 30%;">
                <h3>Loan Valuation</h3>
                <p>Evaluate the worth of your loan based on financial metrics.</p>
                <a href="{{ url_for('loan_valuation') }}" style="display: inline-block; padding: 10px 15px; background-color: #00796b; color: #ffffff; border-radius: 5px; text-decoration: none; transition: background-color 0.3s;">Get Started</a>
            </div>
            <div style="background-color: #e0f7fa; padding: 20px; border-radius: 8px; box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1); width: 30%;">
                <h3>Risk Assessment</h3>
                <p>Analyze the risks associated with your loan application.</p>
                <a href="{{ url_for('risk_assessment') }}" style="display: inline-block; padding: 10px 15px; background-color: #00796b; color: #ffffff; border-radius: 5px; text-decoration: none; transition: background-color 0.3s;">Get Started</a>
            </div>
        </div>
```

```
    </main>
    <footer style="background-color: #004d40; color: #ffffff; text-align: center; padding: 10px 0; position: absolute; bottom: 0; width: 100%;">
        <p>© 2024 Credisense: AI Driven Sanctions</p>
    </footer>
</body>
</html>
```

Flask:

```
from flask import Flask, render_template, request
import pickle
import numpy as np
import pandas as pd
from sklearn.preprocessing import StandardScaler

app = Flask(__name__)

# Load the trained model
model_path = 'model/model.pkl'
with open(model_path, 'rb') as file:
    stacking_clf = pickle.load(file)

# Load your training data for scaling (ensure the path is correct)
```

```
training_data_path = r'C:\Users\haris\OneDrive\Desktop\Projects\Data Science mini project\flask\clean.csv'  # Adjust this path
training_data = pd.read_csv(training_data_path)


# Extract the features used for training
features = training_data[['dti', 'annual income', 'revol.bal', 'revol.util',
                          'days.with.cr.line', 'fico', 'pub.rec', 'inq.last.6mths']]


# Create a StandardScaler instance and fit it on training data
scaler = StandardScaler()
scaler.fit(features)


@app.route('/')
def home():
    return render_template('index.html')


@app.route('/loan_prediction', methods=['GET', 'POST'])
def loan_prediction():
    if request.method == 'POST':
        user_input = [
            float(request.form['dti']),
            float(request.form['annual_income']),
            float(request.form['revol_bal']),
```

```
            float(request.form['revol_util']),
            float(request.form['days_with_cr_line']),
            float(request.form['fico']),
            int(request.form['pub_rec']),
            int(request.form['inq_last_6mths'])
        ]

        # Scale user input
        user_input_scaled = scaler.transform(np.array([user_input]))

        # Make prediction
        prediction = stacking_clf.predict(user_input_scaled)
        result = "Approved" if prediction[0] == 1 else "Rejected"
        return render_template('loan_prediction.html', result=result)

    return render_template('loan_prediction.html', result=None)


@app.route('/loan_valuation', methods=['GET', 'POST'])
def loan_valuation():
    if request.method == 'POST':
        annual_salary = float(request.form['annual_salary'])
        credit_score = float(request.form['credit_score'])
        outstanding_loan_amount = float(request.form['outstanding_loan_amount'])
```

```
        recommended_loan_amount = calculate_loan_amount(annual_salary,
credit_score, outstanding_loan_amount)
        return render_template('loan_valuation.html',
recommended_loan_amount=recommended_loan_amount)

    return render_template('loan_valuation.html', recommended_loan_amount=None)

@app.route('/risk_assessment', methods=['GET', 'POST'])
def risk_assessment():
    if request.method == 'POST':
        user_data = {
            'credit.policy': int(request.form['credit_policy']),
            'int_rate': float(request.form['int_rate']),
            'installment': float(request.form['installment']),
            'log.annual.inc': float(request.form['log_annual_inc']),
            'dti': float(request.form['dti']),
            'fico': int(request.form['fico']),
            'days.with.cr.line': float(request.form['days_with_cr_line']),
            'revol.bal': float(request.form['revol_bal']),
            'revol.util': float(request.form['revol_util']),
            'inq.last.6mths': int(request.form['inq_last_6mths']),
            'delinq.2yrs': int(request.form['delinq_2yrs']),
```

```
            'pub.rec': int(request.form['pub_rec'])
        }

        risk_score, risk_level = calculate_risk(user_data)
        return render_template('risk_assessment.html', risk_score=risk_score,
risk_level=risk_level)

    return render_template('risk_assessment.html', risk_score=None)

def calculate_loan_amount(annual_salary, credit_score, outstanding_loan_amount):
    base_loan_amount = annual_salary * 0.3
    if 600 <= credit_score < 700:
        credit_score_factor = 0.8
    elif 700 <= credit_score < 800:
        credit_score_factor = 1.0
    elif credit_score >= 800:
        credit_score_factor = 1.2
    else:
        credit_score_factor = 0.5
    adjusted_loan_amount = base_loan_amount * credit_score_factor
    final_loan_amount = adjusted_loan_amount - outstanding_loan_amount
    return max(final_loan_amount, 0)
```

```
def calculate_risk(user_data):
    risk_score = 0


    # Credit Policy
    if user_data['credit.policy'] == 1:
        risk_score -= 1  # Lower risk if credit policy is met
    else:
        risk_score += 1  # Higher risk if credit policy is not met


    # FICO Score
    fico = user_data['fico']
    if fico >= 700:
        risk_score -= 1  # Lower risk
    elif fico >= 650:
        risk_score += 0  # Moderate risk
    else:
        risk_score += 1  # Higher risk


    # Debt-to-Income Ratio (DTI)
    dti = user_data['dti']
    if dti < 30:
        risk_score -= 1  # Lower risk
    elif dti <= 40:
```

```
    risk_score += 0  # Moderate risk
else:
    risk_score += 1  # Higher risk


# Interest Rate
int_rate = user_data['int_rate']
if int_rate < 5:  # Assuming interest rate is in percentage
    risk_score -= 1  # Lower risk
elif int_rate <= 10:
    risk_score += 0  # Moderate risk
else:
    risk_score += 1  # Higher risk


# Days with Credit Line
days_with_cr_line = user_data['days.with.cr.line']
if days_with_cr_line > 365:
    risk_score -= 1  # Lower risk
elif days_with_cr_line >= 180:
    risk_score += 0  # Moderate risk
else:
    risk_score += 1  # Higher risk


# Revolving Utilization
```

```
revol_util = user_data['revol.util']
if revol_util < 30:  # Assuming this is a percentage
    risk_score -= 1  # Lower risk
elif revol_util <= 50:
    risk_score += 0  # Moderate risk
else:
    risk_score += 1  # Higher risk

# Inquiries in Last 6 Months
inquiries = user_data['inq.last.6mths']
if inquiries == 0:
    risk_score -= 1  # Lower risk
elif inquiries <= 2:
    risk_score += 0
else:
    risk_score += 1  # Higher risk

# Clamp the risk score between 0 and 5
risk_score = max(0, min(risk_score, 5))
risk_level = ""

if risk_score <= 2:
    risk_level = "Low Risk"
```

```
    elif risk_score <= 4:
        risk_level = "Medium Risk"
    else:
        risk_level = "High Risk"

    return risk_score, risk_level

if _name___== '_main_':
    app.run(debug=True)
```

Output for web design:

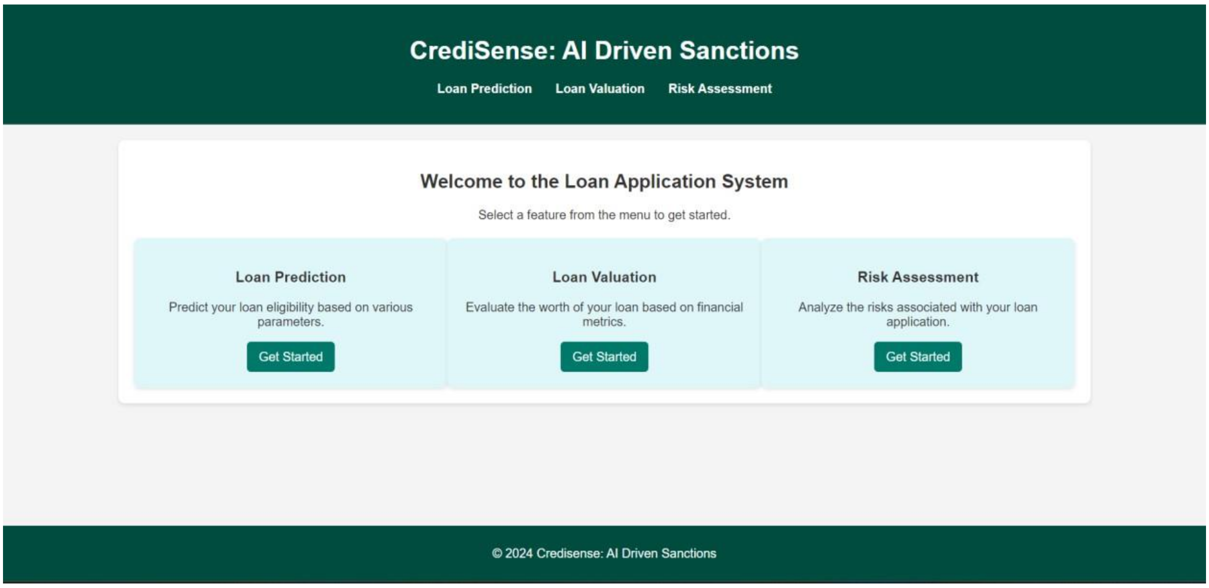

Fig A.4 Web Design

A1.4 Visualization sample code:

Python:

```
# Heatmap for correlation
```

```
import seaborn as sns

import matplotlib.pyplot as plt


correlation_matrix = data.corr()

plt.figure(figsize=(12, 8))

sns.heatmap(correlation_matrix, annot=False, fmt=".2f", cmap='BrBG', cbar=True,
square=True)

plt.title('Heatmap to show correlation b/w features')

plt.show()
```

Output for Visualization:

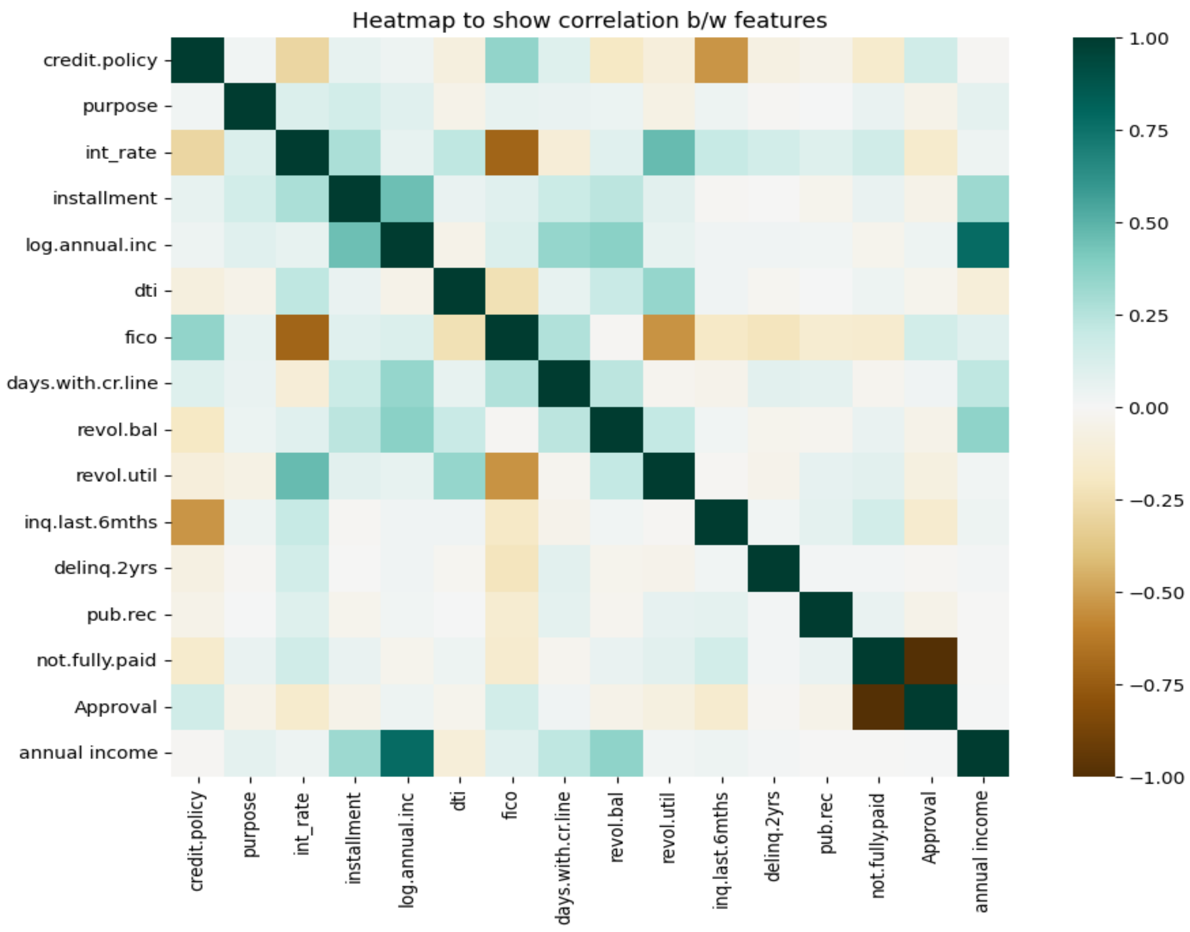

Fig A.5 Heatmap

References:

[1] Y. Diwate, P. Rana, and P. Chavan, "Loan Approval Prediction Using Machine Learning," International Research Journal of Engineering and Technology (IRJET), vol. 8, no. 5, pp. 1741-1745, May 2021.

[2] N. Uddin, M. K. Uddin Ahamed, M. A. Uddin, M. M. Islam, M. A. Talukder, and S. Aryal, "An Ensemble Machine Learning Based Bank Loan Approval Predictions System with a Smart Application," International Journal of Cognitive Computing in Engineering, vol. 4, pp. 327-339, Sep. 2023.

[3] Amruta S. Aphale, " Predict Loan Approval in Bankingm System Machine Learning Approach for Cooperative Banks Loan Approval ", International Journal of Engineering Research & Technology, Vol. 9 Issue 8, August-2020.

[4] M. A. R. Mollah, S. J. S. T. Selvan, and S. A. H. A. Baig, "A Machine Learning Approach for Credit Scoring Prediction and Loan Risk Assessment," *IEEE Access*, vol. 8, pp. 178306–178319, 2020, doi: 10.1109/ACCESS.2020.3023952.

[5] D. M. R. R. K. P. Dinesh, S. Kumar, and S. S. Rao, "Credit Scoring and Risk Assessment Using Machine Learning: A Survey," *IEEE Transactions on Artificial Intelligence*, vol. 1, no. 1, pp. 38-49, Feb. 2020, doi: 10.1109/TAI.2020.2991292.

[6] S. Patel, K. Gohil, and V. Patel, "Loan Default Prediction Using Machine Learning Techniques: A Review," *IEEE International Conference on Machine Learning and Data Science (MLDS)*, Pune, India, 2020, pp. 345–350, doi: 10.1109/MLDS.2020.00063.

[7] S. S. R. Abidi, S. Sarfraz, and M. Ali, "Loan Default Prediction Using XGBoost for Financial Risk Assessment," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 6, pp. 1484-1494, Dec. 2020, doi: 10.1109/TCSS.2020.2978006.

[8] G. H. Goh, J. H. Nguyen, and D. H. Pham, "Financial Fraud Detection and Loan Risk Assessment Using Machine Learning," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 7, pp. 2310-2319, July 2020, doi: 10.1109/TSMC.2019.2955049.

[9] L. P. Li, Z. Wang, and F. Yang, "A Study on the Risk Assessment Model for Loan Default Prediction in Financial Institutions," *IEEE Access*, vol. 9, pp. 50204–50213, 2021, doi: 10.1109/ACCESS.2021.3066144.