Chittagong University of Engineering & Technology (CUET) Chattogram-4349, Bangladesh

Project Report on

Bank Loan Prediction System

Submitted by

Asrarul Hoque Eusha(1804081)
MD. Al-Amin Bhuiyan(1804084)
Ahammed Zayed Uddin Rahat(1804088)
Mirza Asif Razoan(1804093)
Raiyan Bashir Mahin(1804097)

Course Information

Course Name : Software Development Project (Sessional)
Course Code : CSE-300

Course Teacher

Animesh Chandra Roy
Assistant Professor
Department of Computer Science & Engineering,
Chittagong University Engineering & Technology, Bangladesh.

Md. Atiqul Islam Rizvi
Assistant Professor
Department of Computer Science & Engineering,
Chittagong University Engineering & Technology, Bangladesh.

February, 2023

Table of Contents

List of Figures

1 Introduction

A classifier is a particular kind of machine learning algorithm used to categorize data input. When the classifier has had enough training, it may take in unlabeled images and output classification labels for each image. To make predictions about the likelihood of a data input being classified in a particular way, classifier algorithms use complex mathematical and statistical techniques.A machine learning classifier is a type of algorithm that uses statistical and computational methods to analyze data and make predictions based on that analysis. Classifiers can be used for binary classification (predicting one of two outcomes) or multi-class classification (predicting one of many possible outcomes). Examples of classifiers include decision trees, k-nearest neighbors (KNN), support vector machines (SVM), and Naive Bayes. The performance of a classifier depends on the quality and structure of the training data, the choice of features, and the choice of algorithm.We are going to predictions customers are eligible for the loan and check whether what are the missing criteria to know why customer not getting loan.

2 Motivation

The motivation behind making a bank loan prediction using a machine learning classifier is to automate and improve the loan approval process. This would help the bank make more informed decisions, reduce the risk of loan default, and ultimately increase the overall efficiency and profitability of the loan department. By using machine learning algorithms, the bank can analyze a large amount of historical data and identify patterns and trends that can be used to make more accurate loan predictions. This approach is more data-driven and objective compared to traditional methods and can lead to better decision-making in a shorter amount of time. By using a machine learning model, the bank can automate the loan evaluation process, reducing the manual effort and time involved in making a decision.

3 Objectives

- To reduce time wastage during conventional loan approval procedure
- To utilize latest trends in machine vision to implement a feasible solution for loan system
- To automate the whole process so that we have digital environment
- To encourage the use of technology in daily lives

.

4 Related Works

[1] Predicting credit defaulters is a difficult task for the banking industry. The system approved or rejects the loan applications.In this paper three machine learning algorithms, Logistic Regression (LR), Decision Tree (DT) and Random Forest (RF) are applied to predict the loan approval of customers.

[2]A bank's profit or a loss depends to a large extent on loans i.e. whether the customers are paying back the loan or defaulting. By predicting the loan defaulters, the bank can reduce its Non-performing Assets. This makes the study of this phenomenon very important

[3]The Loan Prediction System can automatically calculate the weight of each features taking part in loan processing and on new test data same features are processed with respect to their associated weight .A time limit can be set for the applicant to check whether his/her loan can be sanctioned or not..

5 Methodology

5.1 Loan Dataset

Loan Dataset is very useful in our system for prediction of more accurate result. Using the loan Dataset the system will automatically predict which costumer's loan it should approve and which to reject. System will accept loan application

form as an input. Justified format of application form should be given as an input to get processed.

5.2 Determine the training and testing data

Typically , Here the system separate a data set into a training set and testing set ,most of the data use for training ,and a smaller portions of data is use for testing. after a system has been processed by using the training set, it makes the prediction against the test set.

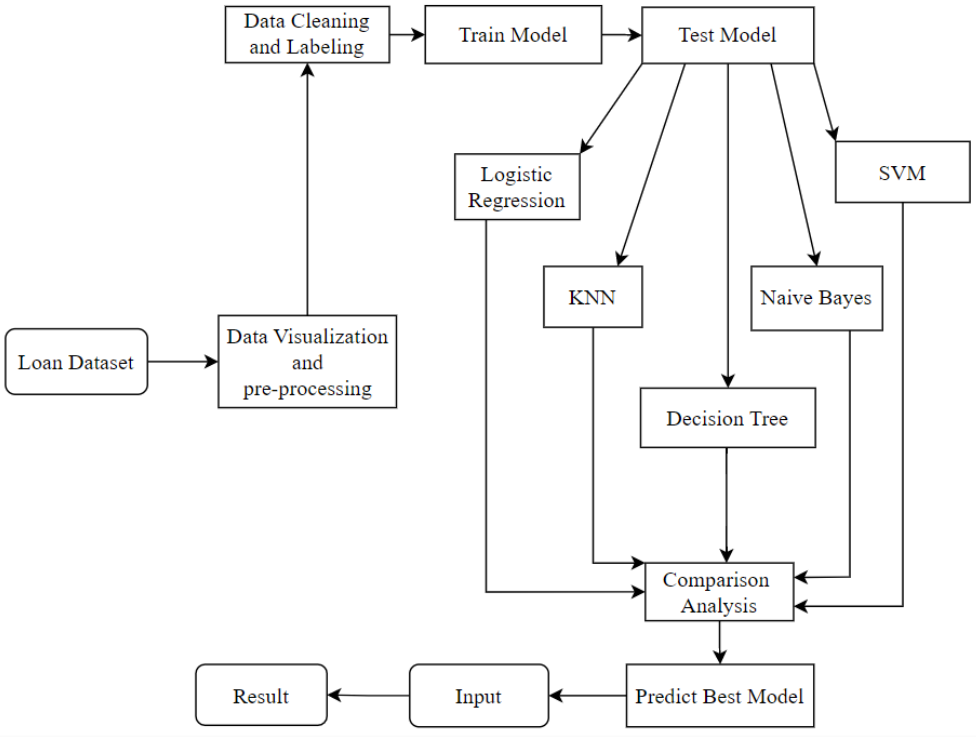

Figure 5.1: Methodology of the proposed system

5.3 Data cleaning and processing

In Data cleaning the system detect and correct corrupt or inaccurate records from database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing , modifying or detecting the dirty or coarse data. In Data processing the system convert data from a given form to a much more usable and desired form i.e. make it more meaningful and informative.

5.4 Models used

Typically , Here the system separate a dataset into a training set and testing set ,most of the data use for training ,and a smaller portions of data is use for testing. after a system has been processed by using the training set, it makes the prediction against the test set.

6 Required Technology

- Anaconda: Anaconda is a distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment.
- Colab : The Basics. Colaboratory, or "Colab" for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education.
- Visual Studio Code : Visual Studio Code (VSCode) is a free and open-source code editor developed by Microsoft. VSCode has become popular among developers due to its speed, versatility, and ease of use, and it's widely used for tasks such as web development, cloud development, and data science.
- Steamlit : Streamlit is an open-source Python library for building and deploying modern, interactive web-based applications for machine learning and data science.Streamlit is widely used by data scientists, machine learning engineers, and data analysts for tasks such as data exploration, prototyping, and model deployment.

.

7 Description

- Load Dataset: To load dataset, we use a specific library or function that is

compatible with the format of the dataset.The dataset is in a CSV file, We use the pandas library in Python to load it into a Pandas DataFrame.

```
[ ] #  Create New Variable and stores the dataset values as Data Frame
    loan_train = pd.read_csv('/content/drive/MyDrive/Software/loan-train.csv')

[ ] loan_train
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0	Urban	Y
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0	Rural	N
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0	Urban	Y
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	Urban	Y
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0	Urban	Y
...	...	...	...	...	...	...	...	...	...	...	...	...	...
609	LP002978	Female	No	0	Graduate	No	2900	0.0	71.0	360.0	1.0	Rural	Y
610	LP002979	Male	Yes	3+	Graduate	No	4106	0.0	40.0	180.0	1.0	Rural	Y
611	LP002983	Male	Yes	1	Graduate	No	8072	240.0	253.0	360.0	1.0	Urban	Y
612	LP002984	Male	Yes	2	Graduate	No	7583	0.0	187.0	360.0	1.0	Urban	Y
613	LP002990	Female	No	0	Graduate	Yes	4583	0.0	133.0	360.0	0.0	Semiurban	N

614 rows × 13 columns

Figure 7.2: Load Dataset

- Data Pre-processing: Data pre-processing is an important step in the machine learning process. It involves removing or correcting inaccuracies, inconsistencies, and missing values in the dataset. This is done to ensure that the data is in a suitable format for training a machine learning model.

```
for featureName in loan_train_columns:
    if loan_train[featureName].dtype == 'object':
        print('\n"' + str(featureName) + '\'s" Values with count are :')
        explore_object_type(loan_train, str(featureName))
```

```
"Loan_ID's" Values with count are :
LP001002    1
LP002328    1
LP002305    1
LP002308    1
LP002314    1
           ..
LP001692    1
LP001693    1
LP001698    1
LP001699    1
LP002990    1
Name: Loan_ID, Length: 614, dtype: int64

"Gender's" Values with count are :
Male      489
Female    112
Name: Gender, dtype: int64

"Married's" Values with count are :
Yes    398
No     213
Name: Married, dtype: int64
```

Figure 7.3: Data Pre-processing

- Data Labeling: Data leveling refers to the process of transforming variables in the dataset so that they are on the same scale. This can be important because some machine learning algorithms are sensitive to the scale of the variables and can produce better results when the variables are transformed to a common scale.
- Train Dataset: A train dataset is a subset of data used to train a machine learning model. The model uses this data to learn patterns and relationships in the data, and make predictions based on that knowledge. The accuracy of the model's predictions can be improved by using a larger and more diverse train dataset.

Figure 7.4: Data Labeling

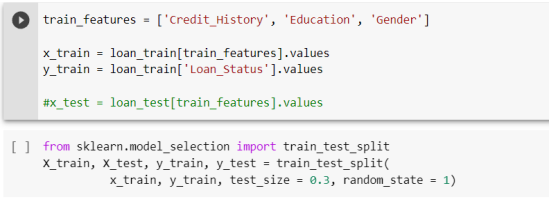

Figure 7.5: Train Dataset

.

To accurately predict our result in the project we used five different classifiers. Here is a description of each classification:

7.1 Classifiers

- Logistic Regression: Logistic Regression is a statistical method for analyzing a dataset in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes). It is used to predict a binary outcome (1/0,Yes/No,True/False) given a set of independent variables. The logistic regression model is an extension of linear regression and is used to model the relationship between the dependent variable and one or more independent variables by fitting a logistic curve to the observed data.

- K-Nearest Neighbors(KNN): KNN is a simple, supervised machine learning algorithm that can be used for both classification and regression tasks. In the KNN algorithm, an object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among

```
[ ] logistic_model.fit(X_train, y_train)

    LogisticRegression()

[ ] accuracy=metrics.accuracy_score(y_test, predicted)
    accuracy

    0.7891891891891892

[ ] conf_matrix = metrics.confusion_matrix(y_test, predicted)
    sns.heatmap(conf_matrix, annot=True,cmap='Greens')

    <matplotlib.axes._subplots.AxesSubplot at 0x7f7f083a78e0>
```

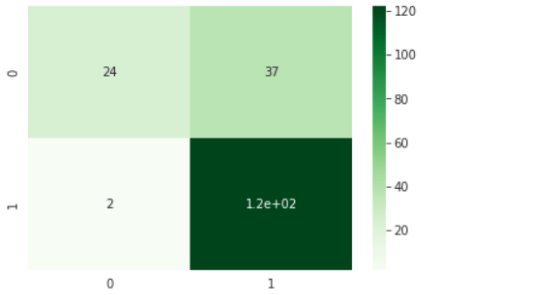

Figure 7.6: Logistic Regression

its k nearest neighbors (k is a positive integer, typically small). The distance between the object and the data points is calculated using a distance metric such as Euclidean distance.

```
[ ] from sklearn.neighbors import KNeighborsClassifier
    knn = KNeighborsClassifier()
    knn.fit(X_train, y_train)
    predicted = knn.predict(X_test)

[ ] accuracy=metrics.accuracy_score(y_test, predicted)
    accuracy

    0.7891891891891892

[ ] conf_matrix = metrics.confusion_matrix(y_test, predicted)
    sns.heatmap(conf_matrix, annot=True,cmap='Greens')

    <matplotlib.axes._subplots.AxesSubplot at 0x7f7f07537610>
```

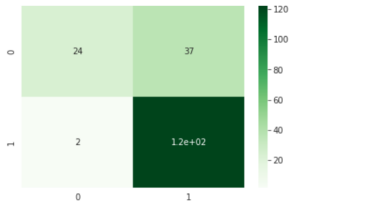

Figure 7.7: K-Nearest Neighbors

- Decision Tree: A Decision Tree is a tree-based model used for problem-solving and decision-making. It is a type of supervised machine learning algorithm that is mostly used for classification and regression tasks. Each internal node in the tree represents a test on an attribute, each branch represents the outcome of a test, and each leaf node represents a class label or a predicted value. The decision tree is built by starting from the root

and adding internal nodes, branches, and leaves until a stopping criterion is met, such as all the instances belong to the same class or the tree has reached a maximum depth.

Figure 7.8: Decision Tree

- Naive Bayes: Naive Bayes is a probabilistic machine learning algorithm based on Bayes' theorem, which states that the probability of a class given some features is equal to the probability of the features given the class, multiplied by the prior probability of the class. It is called "naive" because it makes a strong assumption that all the features are independent of each other, which is rarely true in real-world problems. Naive Bayes is a fast and simple algorithm that works well for large datasets and high-dimensional problems, and is often used for text classification, spam filtering, and sentiment analysis.

- Support Vector Machine (SVM): SVM is a type of supervised machine learning algorithm that is mainly used for classification, but can also be used for regression. SVM tries to find a hyperplane that best separates the data into different classes. A hyperplane is a decision boundary that separates the data into different classes. In two-dimensional space, a hyperplane is a line, in three-dimensional space it is a plane, and in higher dimensions, it is a hyperplane.SVM is a powerful and versatile algorithm that can handle both

Figure 7.9: Naive Bayes

linear and non-linear data and is often used for text classification, image classification, and bioinformatics problems.

Figure 7.10: Support Vector Machine

- Pipelining and Comparison Analysis : By using a pipeline, we reduce the risk of introducing errors and inconsistencies into the process, and make it easier to compare different models and parameters. Pipelines can also be useful for large and complex datasets, as we can help to manage the computational and memory resources required to process the data.

Figure 7.11: Pipelining and Comparison Analysis

- Deploy model: Deploying a machine learning model means making it available for use in a real-world scenario, such as a web application or a production environment.It's important to note that deploying a machine learning model is not just about making it available for use, but also about ensuring that it is robust, secure, and scalable. The deployment process must take into account various operational and security concerns, such as data privacy, security, and compliance.

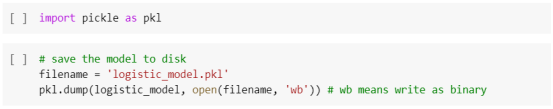

Figure 7.12: Deploy Model

.

7.2 Import Packages

Importing all the packages needed like Streamlit for building interactive machine learning and data science applications, NumPy(Numerical Python) as it's widely used for scientific computing, data analysis, and machine learning, Pandas, PIL (Python Imaging Library)for opening, manipulating, and saving many different image file formats and Pickle to save our machine learning model, trained on a large dataset , to disk and later load it for making predictions on new data shown in Figure 7.13

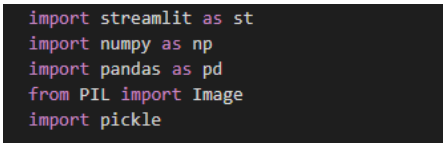

Figure 7.13: Importing Packages

7.3 Loading the Model

Loading our model using pickle which provides a simple and convenient way to save Python objects to a file and later retrieve them, preserving their structure and information shown in Figure 7.14.

```
model = pickle.load(open('C:/Users/HP/Desktop/SDP Final/logistic_model.pkl','rb'))
```

Figure 7.14: Loading the Model

7.4 Image and Title

Making our interface creating run() function shown in Figure 7.15.

```
def run():
    img1 = Image.open('bank.jpg')
    img1 = img1.resize((156,145))
    st.image(img1,use_column_width = False)
    st.title("Bank Loan Prediction Using Machine Learning")
```

Figure 7.15: Image and Title

7.5 Tuples

Tuple are ordered, immutable, and heterogeneous collection of elements. Tuples are similar to lists, but unlike lists, tuples cannot be modified after they are created. We used that kind of data we need to predict the result accurately shown in Figure 7.16.

```
##Account no
account_no = st.text_input('Account Number')

##Full name
fn = st.text_input('Full Name')

##For gender
gen_display = ('Female','Male')
gen_options = list(range(len(gen_display)))
gen = st.selectbox("Gender",gen_options, format_func = lambda x:gen_display[x])

##For marital status
mar_display = ('No','Yes')
mar_options = list(range(len(mar_display)))
mar = st.selectbox("Marital Status",mar_options, format_func = lambda x:mar_display[x])

##No of dependents
dep_display = ('No','One','Two','More than Two')
dep_options = list(range(len(dep_display)))
dep = st.selectbox("Dependents",dep_options, format_func = lambda x:dep_display[x])

#For edu
edu_display = ('Not Graduate','Graduate')
edu_options = list(range(len(edu_display)))
edu = st.selectbox("Education",edu_options, format_func = lambda x:edu_display[x])
```

```
##For emp status
emp_display = ('Job','Business')
emp_options = list(range(len(emp_display)))
emp = st.selectbox("Employment Status",emp_options, format_func = lambda x:emp_display[x])

#For property status
prop_display = ('Rural','Semi-Urban', 'Urban')
prop_options = list(range(len(prop_display)))
prop = st.selectbox("Property Area",prop_options, format_func = lambda x:prop_display[x])

##For credit score
cred_display = ('Between 300 to 500','Above 500')
cred_options = list(range(len(cred_display)))
cred = st.selectbox("Credit Score",cred_options, format_func = lambda x:cred_display[x])

##Applicant monthly income
mon_income = st.number_input("Applicant's Monthly Income($)",value = 0)

##Co-applicant monthly income
co_mon_income = st.number_input("Co-Applicant's Monthly Income($)", value = 0)

##loan amount
loan_amt = st.number_input("Loan Amount", value = 0)

##loan duration
dur_display = ['2 months', '6 months', '8 months','1 Year', '16 Month']
dur_options = range(len(dur_display))
dur = st.selectbox("Loan Duration",dur_options, format_func = lambda x:dur_display[x])
```

Figure 7.16: Tuples

7.6 Prediction

Prediction in machine learning is the process of using a trained model to make a prediction about a new, unseen data sample. The model takes as input a set of features or variables, and outputs a prediction based on the relationships and patterns it has learned from the training data.Conditions for predicting accurate result of getting loan or not shown in Figure 7.17.

```
if st.button("Submit"):
    duration = 0
    if dur == 0:
        duration = 60
    if dur == 1:
        duration = 180
    if dur == 2:
        duration = 240
    if dur == 3:
        duration = 360
    if dur == 4:
        duration = 480
    features = [[cred, gen, edu]]
    print(features)
    prediction = model.predict(features)
    lc = [str(i)for i in prediction]
    ans = int("".join(lc))
```

Figure 7.17: Prediction

7.7 Output

From showing output of the interface's successful and unsuccessful messages needed command given in Figure 7.18.

```
if ans == 0:
    st.error(
        "Hello: "+fn+ " || "
        "Account Number: "+account_no +' || '
        'According to our calculations, you will not get the loan from Bank'
    )
else:
    st.success(
        "Hello: "+fn+ " || "
        "Account Number: "+account_no +' || '
        'Congratulations!! You will get the loan from Bank'
    )
```

Figure 7.18: Output

7.8 Streamlit Output

Streamlit is a library for creating interactive web applications in Python. The main focus of Streamlit is to allow developers to create apps with a simple and intuitive syntax, making it easy to turn complex data pipelines and machine learning models into user-friendly and interactive applications.

As we use Streamlit app, which simply specify the elements we want to include

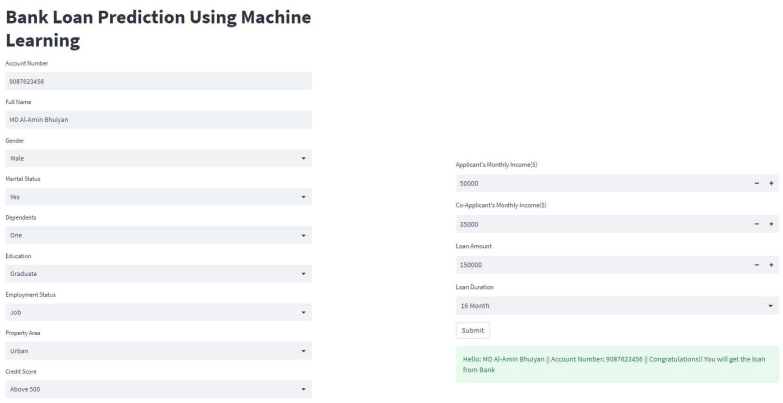

Figure 7.19: Output with getting loan

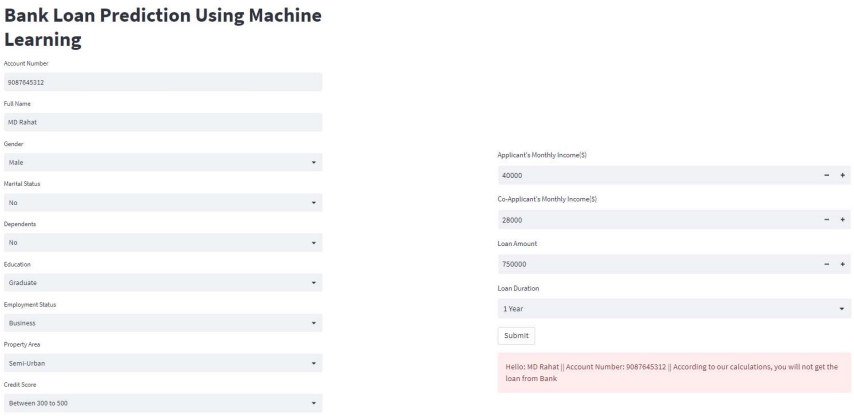

Figure 7.20: Output with not getting loan

on the page and Streamlit takes care of the rest, handling the layout, styling, and user interactions shown in Figure 7.19 and Figure 7.20.

8 Applications

- Loan Approval: The prediction model can be used to approve or reject loan applications based on the calculated risk of loan default. This helps banks to minimize the risk of loan default and improve the loan approval process.
- Loan Amount: The prediction model can also be used to determine the loan amount that a borrower can be approved for. This helps banks to make informed decisions about loan amounts and manage the risk of loan default.
- Loan Terms: The prediction model can be used to determine the terms and conditions of the loan, such as interest rates and repayment schedules, based on the borrower's creditworthiness and ability to repay the loan.

- Customer Segmentation: The prediction model can be used to segment customers into different risk categories, allowing banks to develop targeted marketing and loan products for each group.
- Fraud Detection: The prediction model can be used to detect potential fraud by analyzing anomalies in the loan application data and identifying unusual behavior.

9 Conclusion

Loan prediction includes the lender reviewing numerous background information about the applicant before choosing whether or not to offer the loan. Credit score, loan amount, lifestyle, job, and assets are all determining factors in loan approval.We are going to predictions customers are eligible for the loan and check whether what are the missing criteria to know why customer not getting loan.

References

[1] J. Tejaswini, T. M. Kavya, R. D. N. Ramya, P. S. Triveni and V. R. Maddumala, 'Accurate loan approval prediction based on machine learning approach,' *Journal of Engineering Science*, vol. 11, no. 4, pp. 523–532, 2020 (cit. on p. 2).

[2] A. Kadam, S. Nikam, A. Aher, G. Shelke and A. Chandgude, 'Prediction for loan approval using machine learning algorithm,' *International Research Journal of Engineering and Technology*, vol. 8, no. 04, pp. 4089–4092, 2021 (cit. on p. 2).

[3] S. Bhattad, S. Bawane, S. Agrawal, U. Ramteke and P. Ambhore, 'Loan prediction using machine learning algorithms.,' *International Journal of Computer Science Trends and Technology*, vol. 9, no. 3, pp. 143–146, 2021 (cit. on p. 2).