**Abstract**

Finance sector in India comprises mainly of commercial banks that generate most of their business through the interest received on loans disbursed to people and organisations. However, credit lending comes with risk of consumer defaulting the loan i.e., inability to pay back the due sum of money agreed upon by both the parties. Ensuring that the borrower will be able to pay back the loan on the proposed terms is the biggest challenge faced by banks in today’s ever-changing world. This paper proposes to leverage the use of Machine Learning (ML) algorithms to predict which clients are more likely to default on their using historical financial data like credit and debit card data, bank account transactions, previous credit history, current bank loans and income of the client.

This study makes use of the Credit Risk Model Stability data available on Kaggle provided by the Home Credit finance provider, founded in 1997. The dataset consists of 32 csv training files and 36 testing files comprising of financial records of over 1.5 million clients, collected from internal and external sources. The research is a comparative analysis between Bayesian learning, Lazy learning, ensemble learning and gradient boosting machine learning algorithms. Gradient boosting algorithms Lightgbm, XGBoost and Adaboost outperform other algorithms and predict with upto 99% accuracy whether a client will default a loan or not. Results of this study can be scaled and applied to a real world dataset and holds the immense potential to revolutionise the financial industry.

*Keywords – credit; loan default; Lightgbm; XGBoost; Adaboost*

**Introduction**

Commercial banks are the main players in Indian financial sector. Most of their revenue is earned from the interest on loans that is extended to individuals and corporates. However, granting loans has its own pros and cons. One of the cons include default risk. Default risk means when someone takes a loan but due to some reason is unable to pay back the agreed amount. To minimize this risk and ensure that the loan is repaid according to the accepted terms and conditions, gave us the motivation to work on this project.

To address this challenge, this paper leverages eight different Machine Learning (ML) algorithms to predict the likelihood of loan default. These algorithms include Random Forest, Decision Tree, K- Nearest Neighbours, Gaussian Naïve Bayes, XG-Boost, Light Gradient Boosting Machine (LGBM) , Ada Boost and Long Short Term Memory(LSTM).

**Literature Survey**

This section discusses in brief about some of the work that has already been done on creating Machine Learning models using various algorithms to help decrease the credit card default and help the banking authorities and financial firms select an eligible candidate with low credit risk.

Bhoomi Patel et al. in [1] compared four algorithms named Logistic Regression, Gradient Boosting, CatBoost Classifier and Random Forest to predict loan default. In [1] CatBoost Classifier outperformed other algorithms. It has an accuracy of 84.045%, whereas the other algorithms were 14.963%, 84.035%, and 83.514% respectively. Mehul Madaan et al. In [2], compared Random Forest and Decision Tree algorithms to predict loan default. The conclusion of [2] is that Random Forest with an accuracy of 80% outperformed Decision Tree algorithm that gave an accuracy of 73%. The dataset that they used had biased data. Huannan Zhang et al. in [3] compared Random Forest, Decision Tree and Logistic Regression algorithms to showcase the application of Random Forest Classifier in Loan default forecast. The conclusion of [3] is that the Random Forest Algorithm (≈ 86%) exceeds the decision tree (≈80%) and logistic regression classification (≈80%). Alžbeta Bačová and František Babič in their [5] have used Random Forest, AdaBoost and XGBoost for predictive analysis for credit card default. The results of [5] showed that the performance of these algorithms was very similar. Lili Lai in [7] has compared AdaBoost, XGBoost, Random Forest, KNN and Multi-Layer Perceptron algorithms to predict loan default. The conclusion of [7] is that AdaBoost outperformed all the other algorithms followed by XGBoost. Theoneste Ndayisenga in his [8] has mentioned the use of Logistic Regression, Decision Tree, Support Vector Machines, Random Forest, KNN, Gausian Naive Bayes, Gradient Boosting and XG Boost. The result of the analysis of these algorithms shows that Gradient Boosting (≈ 81%) is the best model to predict bank default followed by XG Boost (≈ 80%).

Abhishek Agarwal et al. in [4] have mentioned about Logistic Regression, Random Forest, Decision Trees, Naïve Bayes and KNN in [4]. The main motive of [4] is to compare measures between the original dataset before and after applying the Principal Component Analysis. The conclusion of [4] is that the accuracy of Logistic Regression was best in both the cases and Decision Tress was not affected much. Mohammad Ahmad Sheikh et al. in [6] have used Principal Component Analysis (PCA) to analyse its importance. The conclusion of [6] Is that the model is marginally better after applying PCA.

**Methodology**

* **Data Collection**

The dataset used in the model is the Home Credit – Credit Risk Model Stability provided by the Home Credit organisation on Kaggle. It consists of 32 training files and 36 testing files. All the files are available in csv and parquet format. The files consists of data collected for a particular case id from two types of sources: internal and external data sources.

The training files are used to train and test the model. These are further divided into two parts in the ratio 75:25 for training and testing purposes respectively.

* **Data mining**

The csv files extracted from the dataset are needed to be merged together and converted into a format that can fed into the machine learning models. To merge all the files together, a “train\_base.csv” file is provided that contains the target variable. The files were merged into a single dataset using the python library, polars. Polars is a python library written in Rust and uses a multi-threaded query engine for fast and effective parallel execution.

* **Preprocessing**

After converting all the csv files into a single polars dataframe, the data consists of null values, categorical string data and dates. This data needs to be preprocessed and converted into a usable format. Firstly, the dataset is transformed back into pandas dataframe for label encoding the columns with string data. Label Encoding is a preprocessing technique which converts string data into numerical data by assigning a unique index value to each unique string value in a column.

Another data preprocessing technique known as standard scaling is used. Standard scaling is also known as Z-score normalisation. The numerical features of the dataset are scaled to have a mean of 0 and standard deviation of 1. The formula for standard scaling is:

z = (x−μ) / σ

where, x = original value of the feature

u = mean of the feature values

o = standard deviation of the feature values

Standard scaling is used to normalise distribution of values, reduces dominance of a single feature or multiple features in the dataset and improves converge rate of the model.

* **Feature Engineering**

The final dataset after cleaning and preprocessing consists of 303 independent features, 1 target variable and over 15 lakh records. The size of the dataset was reduced from 15 lakh records to 1 lakh records so that dataset is balanced and there is equal representation between the two classes. Faster convergence and improved training times can be achieved using an algorithm called Principal Component Analysis (PCA). PCA is a dimensionality reduction technique that is used to reduce the number of features in a dataset while preserving all or most of the essential information. PCA can be lossy or lossless in nature depending on the dataset. Using PCA, top 100 features were extracted to train the machine learning models.

* **Proposed Model**

The architecture of the model proposed is as follows:

<Diagram/flowchart>

Given the nature of the dataset, the machine learning algorithms used for comparative analysis in this study are: ensemble learning, gradient boosting, bayesian learning, lazy learning and deep learning algorithms. The dataset is divided into 75:25 training and testing data respectively.

* **Decision Tree**

Decision tree is a heirarchical structure that can be used for both, regression as well as classification tasks. Each internal node represents a decision taken based on a feature and leaf nodes represent the output of the model. It splits the dataset features recursively into subsets based on the feature that best divides the data into separate classes. The dataset is divided at each step such that it maximises the information gain.

* **Random Forest**

Random forest is an ensemble learning algorithm that builds multiple decision trees during model training. The output of a random forest model for a classification task is based on voting of the various decision trees and for a regression task, the output is the mean of the output of the internal decision trees. It is indifferent to noisy data and generalises the model reducing overfitting.

* **Gradient Boosting**

Gradient boosting is a boosting technique in which a strong model is built by sequential learning of multipe weak learning models. It combines weak learner models and optimises them to minimise the value of a loss function. Gradient boosting algorithms used in this study are: XGBoost, Ada Boost and LightGBM

* **Gaussian Naive Bayes**

Gaussian Naive Bayes is a probabilistic classifier algorithm that is based on the Bayes’ theorem. It takes into consideration that the input features are independent and they follow a Gaussian (normal) distribution curve. It works by calculating the probability for each class for a set of input features and returns the class with maximum probability.

* **K-Nearest Neighbours**

K-nearest neighbours is a lazy learning algorithm that allots a class label to a data point based on the majority of its k-nearest neighbours calculated using a distance metric from the given point. It does not store the dataset in memory and is simple to implement.

* **Long-Short Term Memory (LSTM)**

LSTM is a type of Recurrent Neural Network (RNN) architecture that helps to capture long term and sequential data. It consists of memory cell, input gate, output gate and a forget gate to help retain important information throughout the model training phase.

* **Hyperparameter Tuning**

Hyperparameter refers to parameters of a machine learning model that are set prior to model training process. Hyperparameter tuning is the process of finding optimal values for the hyperparameters of the machine learning models.

The hyperparameter selection for machine learning was done using gridsearchcv method in the scikit-learn library in python. This method allows a single machine learning model to train on a different number of combinations of hyperparameters and returns the best possible combination having the highest accuracy. In LSTM model, the hyperparameter selection was done using random search and techniques early stopping and model checkpoints along with dropout layers were used to prevent overfitting during model training.

**Results**

This

**References**

[1] Patel, B., Patil, H., Hembram, J., & Jaswal, S. (2020). Loan Default Forecasting using Data Mining. In 2020 International Conference for Emerging Technology (INCET). 2020 International Conference for Emerging Technology (INCET). IEEE. https://doi.org/10.1109/incet49848.2020.9154100

[2] Madaan, M., Kumar, A., Keshri, C., Jain, R., & Nagrath, P. (2021). Loan default prediction using decision trees and random forest: A comparative study. In IOP Conference Series: Materials Science and Engineering (Vol. 1022, Issue 1, p. 012042). IOP Publishing. https://doi.org/10.1088/1757-899x/1022/1/012042

**[3] Huannan Zhang, Yilin Bi, Wangdong Jiang1, Chuntian Luo, Shengjia Cao, Peng Guo and Jianjun Zhang (year) Application of Random Forest Classifier in Loan Default Forecast**

[4] Agarwal, A., Rana, A., Gupta, K., & Verma, N. (2020). A Comparative Study and enhancement of classification techniques using Principal Component Analysis for credit card dataset. In 2020 International Conference on Intelligent Engineering and Management (ICIEM). 2020 International Conference on Intelligent Engineering and Management (ICIEM). IEEE. https://doi.org/10.1109/iciem48762.2020.9160230

[5] Bacova, A., & Babic, F. (2021). Predictive Analytics for Default of Credit Card Clients. In 2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI). 2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI). IEEE. https://doi.org/10.1109/sami50585.2021.9378671

[6] Sheikh, M. A., Goel, A. K., & Kumar, T. (2020). An Approach for Prediction of Loan Approval using Machine Learning Algorithm. In 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC). 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC). IEEE. https://doi.org/10.1109/icesc48915.2020.9155614

[7] Lai, L. (2020). Loan Default Prediction with Machine Learning Techniques. In 2020 International Conference on Computer Communication and Network Security (CCNS). 2020 International Conference on Computer Communication and Network Security (CCNS). IEEE. https://doi.org/10.1109/ccns50731.2020.00009

**[8] Theoneste Ndayisenga (year) Bank Loan Approval Prediction Using Machine Learning Techniques**

[9] Barbaglia, L., Manzan, S., & Tosetti, E. (2021). Forecasting Loan Default in Europe with Machine Learning. In Journal of Financial Econometrics (Vol. 21, Issue 2, pp. 569–596). Oxford University Press (OUP). <https://doi.org/10.1093/jjfinec/nbab010>

[10]Bagga, S., Goyal, A., Gupta, N., & Goyal, A. (2020). Credit Card Fraud Detection using Pipeling and Ensemble Learning. In Procedia Computer Science (Vol. 173, pp. 104–112). Elsevier BV. <https://doi.org/10.1016/j.procs.2020.06.014>

[11]Kim, H., Cho, H., & Ryu, D. (2020). Corporate Default Predictions Using Machine Learning: Literature Review. In Sustainability (Vol. 12, Issue 16, p. 6325). MDPI AG. <https://doi.org/10.3390/su12166325>

[12]Zhu, L., Qiu, D., Ergu, D., Ying, C., & Liu, K. (2019). A study on predicting loan default based on the random forest algorithm. In Procedia Computer Science (Vol. 162, pp. 503–513). Elsevier BV. <https://doi.org/10.1016/j.procs.2019.12.017>