A Deep Learning Hybrid Model for Enhanced Credit Score Prediction

A Thesis submitted in partial fulfillment for the
Degree of MSc in Information Technology from Kirinyaga
University

GRACE WANJIKU KIMANI

AUGUST, 2024

DECLARATION

"I declare that thesis is my original work and has not been presented for a degree or any other award at any other university".

Signed: ________________ Date: ....……

Kimani Grace Wanjiku

PA201/S/17780/22

"This research thesis has been submitted for examination with our approval as university supervisors".

Signed: _________________ Date: ....…..........

Dr. Josphat Karani, PhD.

Signed: _________________ Date: ....

Dr. Ephantus Mwangi, PhD.

COPYRIGHT

DEDICATION

To Samuel, Sheila, Patience, and Rick—your unwavering support, love, and inspiration have been the foundation of my academic journey. This work is dedicated to you.

ACKNOWLEDGMENT

First and foremost, I praise and thank the Almighty God for providing me with direction and strength throughout my research. I extend my heartfelt gratitude to my supervisors, Dr Josphat Karani and Dr Ephantus Mwangi, for their enthusiasm, vision, sincerity, and invaluable advice throughout this journey. I am deeply grateful to my spouse and children for their inspiration, prayers, and significant sacrifices. I also express my appreciation to Dr. Wanjohi, Dean of the School of Pure and Applied Science, and Dr. Kinyanjui, Chairman of the Department of Pure and Applied Science, for their encouragement, especially during times when my morale was low. My fellow students and friends have been incredibly supportive and encouraging, and I am thankful for their assistance with my studies. Finally, I thank the lecturers and staff at the School of Pure and Applied Sciences for their willingness to help and guide me.

ABSTRACT

Determining someone's creditworthiness accurately is still challenging, especially if they have a short credit history or mostly use cash. In these situations, traditional credit scoring techniques frequently fall short, which could cause misclassification and financial losses for lenders. To improve credit score prediction, the research focuses on creating a hybrid deep learning model that blends Recurrent Neural Networks (RNNs) and Deep Neural Networks (DNNs). This study aims to evaluate the role of behavioral and traditional data in the existing credit scoring models, develop a hybrid deep learning model that integrates both data types for predicting credit scores, validate the developed model, and create a web-based tool to visualize the model. Data preparation techniques include feature engineering and feature selection to find complex patterns in the data. Design Science Research (DSR) is the research design used for developing artifacts in this study. The hybrid RNN+DNN model outperforms solo RNN and DNN models, as shown by performance evaluation measures like accuracy, precision, recall, F1-score, AUC-ROC, confusion matrices, sensitivity, specificity, MSE, and RMSE. With an AUC-ROC score of 0.7971, it attains balanced and dependable credit score predictions, with the lowest RMSE (0.723) and MSE (0.523) and sensitivity of 0.8372 for class 2 and specificity of 0.8790 for class 0. By offering consumers a useful interface, the web-based tool designed to display the hybrid credit scoring model increases the usefulness of the research. This application makes it easier to input financial data, analyze it so that the hybrid RNN+DNN model can use it, and then display the anticipated credit scores ('Good,' 'Standard,' or 'Poor'). Streamlit's features guarantee a smooth user experience, confirming the proposed model's efficacy in practical situations. However, difficulties with interpretability, computing requirements, dataset quality, and ethical issues are mentioned. More extensive and more varied datasets should be obtained, hyperparameters should be optimized, computational efficiency should be increased, interpretability should be strengthened, and the model should be validated against actual credit scoring systems in the real world. By addressing these issues, hybrid deep learning models will be further improved, guaranteeing their ethical use in credit evaluation as well as their scalability, comprehensibility, and reliability. This will also help marginalized communities become more financially included.

ABBREVIATIONS AND ACRONYMS

AB	Adaptive Boosting
AE	Auto Encoders
AEs	Autoencoders
AI	Artificial Intelligence
ANNs	Artificial Neural Networks
AUC-PR	Area Under the Precision-Recall Curve
AUC-ROC	Area Under the ROC Curve
CDS	Credit Default Swaps
CNN	Convolutional Neural Networks
CSV	Comma Separated Value
DBNs'	Deep Belief Networks
DL	Deep Learning
DNN	Deep Neural Networks
DSR	Design science research
DT	Decision Trees
EBITDA	Earnings Before Interest, Tax, Depreciation, And Amortization
EV	Enterprise Value
FS	Feature Selection
GB	Gradient Boosting
GDPR	General Data Protection Regulation
IVs	Information Values
LDA	Linear Discriminant Analysis
LIME	Local Interpretable Model-Agnostic Explanations

LR	Logistic Regression
ML	Machine Learning
MLP	Multilayer Perceptron
NB	Naïve Bayes
NB	Naïve Bayes
P/B	Price-to-Book Ratio
P/E	Price-Earnings Ratio
P2P	Peer to Peer
PA	Passive Aggressive
RF	Random Forest
RFR	Random Forest Regression
RNN	Recurrent Neural Networks
ROC	Receiver Operating Characteristic
SHAP	Shapley Additive explanations
SNNs	Simulated Neural Networks
SPNs	Sum-Product Networks
SVM	Support Vector Machine

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

List of Equations

List of Formulas

CHAPTER ONE: INTRODUCTION

This chapter address the back ground of the study, statement of the problem, purpose, main and specific objective, justification, importance and scope of the study.

1.1 Background of the Study

A credit is established when both parties agree to advance the borrower a specific sum of money. (Wallstreetmojo Team, 2023). Credit is built on the belief that a lender may entrust a borrower with resources or money, and the debtor will be able to repay the loan within the specified time frame. Both debtors' and lenders' financial performance depends on credit scoring. Information about borrowers is gathered by lenders, who are typically financial institutions, to assess their creditworthiness (Mhina et al., 2021).

A statistical technique known as credit scoring is used by lenders and financial institutions to assess borrowers' creditworthiness. According to Sujaini (2023), credit scoring models are statistical tools that assess creditworthiness and ascertain the probability of defaulting on credit obligation. Credit bureaus and lenders use these models to evaluate the risk of making loans or extending credit to people or companies. Lenders consult credit scores when determining whether to approve or deny credit requests. They utilize credit scoring as part of risk-based pricing, which determines a loan's terms depending on the likelihood that it will be repaid, including the interest rate (Sujaini, 2023). An automated credit scoring process can gather all the necessary information, evaluate the loan application, and decide whether or not to approve it.

Prediction algorithms for credit scores have now taken a significant role in business. Credit scores are indicators that enable a financial institution to determine a customer's dependability to pay back the debt on schedule. Therefore, the credit score is extremely important for determining how risky a person or asset is (Standard Chartered, 2022). Traditional data, including history of debt repayment, current debt, new credit, duration of credit history, and credit mix, may be used to create a financial credit score. (Fico, 2020).

Traditional data in credit scoring refers to the utilization of a person's personal information and past financial data to evaluate their creditworthiness. This data is typically obtained from a range of sources, such as credit bureaus, banks, and other relevant establishments. This information is examined as part of the typical credit score procedure to assess if a person would likely return their debts or credit obligations on time (FICO, 2020).

For many years, traditional credit scoring methods have been in use. These early models were constructed using manual procedures and simple statistical methods. They largely leaned on traditional data sources such as payment histories, credit utilization, and credit bureau records. (Ampountolas et al., 2021; Kumar& Bhattacharya,2021; Khatir & Bee, 2022; Aniceto et al., 2022).

The lender may suffer damages when a high-risk borrower is mistakenly classified as low-risk based on traditional data. The lender may suffer immediate financial losses if the borrower fails to take out the loan because they will not be able to receive their money back. Furthermore, it could lead to higher default rates for a portfolio of loans. If high-risk borrowers are mistakenly classified as low-risk, the portfolio may be more exposed to credit risk, leading to increased default rates and potential losses for financial institutions (Aniceto et al., 2022). Although typical credit scoring techniques have been in use for decades, there are situations when they may not effectively determine a person's creditworthiness, especially young adults, those who exclusively use cash, or those who have not yet opened traditional credit accounts (FICO, 2020).

In developed countries like the European Union, the law was enacted in 2016; the General Data Protection Regulation (GDPR) presents a list of data storage rules that restrict businesses' ability to store sensitive consumer data. Companies that rely on credit scores have other issues besides the problem of limiting data usage. (Hamberg& Bouvin,2022). According to Hamberg and Bouvin (2022), developing economies suffer the most because of a lack of proper credit institutions and a largely unbanked population. Lenders are turning to alternative data, such as behaviour data, to address these limitations and make credit decisions more inclusive and accurate.

Behavioural data in credit scoring refers to using non-traditional data sources and alternative data points to determine a person's creditworthiness. Behavioural data considers a person's activities and Behaviour in addition to their credit history (Bright Technologies, 2023). The banking transaction data behaviour made available by open banking is a potent supplement to internal credit scores and credit reporting data (Illion, 2022). This strategy tries to offer a more thorough and comprehensive appraisal of a person's credit risk, especially for those with little to no traditional credit history (Chopra, 2020). According to Balduini et al. (2017), a company's or a person's purchasing and payment habits are examples of behavioural data.

In addition to standard credit data sources, behavioural data sources offer insights into a person's financial activities and habits that go beyond what is recorded in their credit history. Behavioural information used to determine credit scores includes alternative credit information such as utility bills and rent payments, employment and income information, patterns of financial activity, and shopping behaviour. (ICCR,2018).

A decision support system assists in determining the combination of classifiers that will produce decisions that are more precise and wiser in a given situation (Kumar, 2021). According to Bhatia et al. (2017), combining various deep learning techniques and methodologies can be helpful when integrating both traditional and behavioural data for credit score prediction. Bhatia et al. (2017) emphasized that predictions from various deep learning (DL) models can be combined using Ensemble approaches like Stacking, Boosting, or Bagging. Additionally, Bhatia et al.. (2017) suggested that an ensemble may produce more reliable and accurate forecasts of credit scores by combining predictions from numerous models trained on distinct subsets of the data. Well-designed features that incorporate both traditional and behavioural data are advantageous for DL models. Making sure the model receives the most informative input requires careful Feature engineering (Bhatia et al.,2017) With young people and those with short credit histories as the target demographic, the study aims to develop a deep learning-based hybrid model for credit score prediction that integrates traditional data with behavioural data. The term "hybrid" describes combining two distinct procedures or methodologies. This study refers to integrating two different neural networks, Recurrent Neural Networks (RNNs) and Deep Neural Networks (DNNs), to produce a more complete model for predicting credit Scoring.

Deep Neural Networks (DNNs) are Feed Forward Networks (FFNNs) where data goes from the input to the output layer without travelling backward. The links connecting the layers are one-way, forward-moving, and never come into contact with another node. DNNs are solid tools for large data and complicated tasks because these layers can train to represent data at ever-higher degrees of abstraction; however, due to their high capacity to learn complex patterns, they suffer overfitting, and they don't have a memory to remember what they started. This problem is handled with a Recurrent Neural Network (RNN), which is an FFNN with a temporal twist and can process input sequences by utilizing their internal state or memory, making it suitable for this research.

The hybrid model combines two deep learning neural networks (RNNs and DNNs) to improve the precision and efficacy of credit score prediction. It seeks to develop a more comprehensive and rigorous model for evaluating creditworthiness by combining behavioral data's distinctive insights with traditional data's well-established metrics. This integration offers a more extensive and knowledgeable evaluation of a person's credit risk, which may lead to more precise lending decisions. One benefit of merging behavioral data and traditional credit data is the ability to evaluate creditworthiness more thoroughly, especially for those with brief credit histories. Behavioral data combined with traditional credit data can offer a real-time perspective of a person's financial activity to lenders to give them more up-to-date information (Sedliacikova et al., 2021)

1.2 Problem Statement

Credit scoring models are essential for assessing borrower creditworthiness and assisting financial institutions in making well-informed lending decisions (Shi et al., 2022). However, to improve their efficacy, the shortcomings of the existing credit scoring models must be addressed (Shi et al., 2022). According to a thorough analysis of credit risk analysis, deep learning models perform statistically and machine learning better than standard methods in evaluating credit risk

(Shi et al., 2022). As a result, deep learning techniques are suggested as advantageous means for creating effective credit scoring models.

According to Benavides et al., 2020 it is crucial to concentrate on the constraints of Recurrent Neural Networks (RNNs) and Deep Neural Networks (DNNs) in this field in order to overcome the difficulties in credit scoring. Because DNNs can represent data at higher levels of abstraction, they are good at handling large datasets and complex tasks. However, they are prone to overfitting because they need more memory to store and use past information (Benavides et al., 2020). Still, while being built with memory in mind, RNNs need help withneed help with problems such as vanishing gradients and restricted long-term dependencies, which can make it difficult for them to identify complex patterns in credit scoring data (Sherstinsky, 2020). DNN identifies complex patterns in credit scoring data, making it a better match. The recurrent neural network (RNN) has gained significance because of its robust capacity for sequence data analysis and self-learning (Xiao &Zhou, 2020). Recent empirical research has shown that when DNNs and RNNs are utilized separately for credit risk assessment, these issues lead to inferior performance. This emphasizes the necessity of a hybrid model that addresses the inadequacies of each architecture separately and enhances credit scoring performance overall by utilizing the strengths of both systems.

Since much empirical research on credit analysis employs biased samples and relies primarily on data from successfully issued loans, it can be challenging to predict creditworthiness (Aniceto et al., 2020) accurately. According to Ampountolas et al. (2021), in order to stop the exploitation of people who utilize microcredit, reasonable lending rates must be determined in these settings. As such, information on people denied loans is frequently included in estimations of creditworthiness. Aniceto et al. (2020) contend that misclassifying a high-risk borrower as low-risk is more expensive than the opposite, highlighting the significance of examining the costs related to misclassification in credit scoring models. If the borrower defaults, this misclassification may result in losses for the lender since the loan amount may not be recouped. It may also result in higher default rates. Furthermore, it may raise the default rates in a lending portfolio, putting financial institutions at greater risk of losses and credit risk.

Because there are so many factors to consider to improve the accuracy and predictive capacity of the model, it can take time to predict credit scores accurately. Sum et al. (2022) pointed out that while creating credit scoring models, several characteristics should be included. Aniceto et al. (2020) stressed the importance of investigating the vast array of potential variables. To be more precise, they proposed applying various machine learning methods to determine the importance of these variables in elucidating credit risk, which might advance the theoretical knowledge of credit risk and emphasize important default drivers.

Even though credit scoring models are essential for assessing creditworthiness, their limitations limit their ability to help lenders make wise loan decisions. Applications based on deep learning have the potential to improve credit risk estimation; they frequently outperform statistical and machine learning techniques (Shi et al., 2022). However, the approaches of risk assessment based on machine learning that are now in use need more transparency and are unable to detect biases in pricing and credit choices, which can result in expensive misclassifications (Aniceto et al., 2020). Credit scoring models can become even more accurate and predictive by adding non-traditional sources like behavioural data, banking transaction behaviours, mobile phone usage, and spending and bill payments. These sources, taken together, can paint a more complete picture of a borrower's creditworthiness (Fintech, 2023; Illion, 2022).

To overcome these constraints and improve credit judgments, there is rising interest in using behavioural data, such as past payment histories, purchasing patterns, and banking transaction patterns. Combining behavioural and traditional data reduces bias and misclassification while providing a more thorough knowledge of credit risk (Balduini et al., 2017). In order to improve credit score prediction, this research focuses on creating a hybrid model based on deep learning that combines Recurrent Neural Networks (RNNs) and Deep Neural Networks (DNNs).

1.3Purpose of the study

The research aims to explore the benefits and challenges of integrating traditional credit data with behavioural data in credit scoring. It aims to solve the limits of traditional credit scoring in properly determining creditworthiness for people with little credit history or the unbanked population. By combining behavioural data like spending, banking transaction data behaviour, and payment histories, the study aims to improve credit risk understanding and reduce misclassifications of high-risk borrowers to low-risk ones. Ultimately, the goal is to develop a robust credit scoring model that overcomes these challenges and provides a more accurate evaluation of creditworthiness, promoting fairness, inclusivity, and accuracy in credit decisions for diverse populations and financial institutions (Fintech, 2023).

1.4 Main objective

The major goal of this study is to develop a deep learning-based hybrid model that combines standard credit data with behavioral data for credit score prediction.

1.5 Specific Objectives

By the end of the study, the researcher was able to:

i. To evaluate the role of behavioral and traditional data in the existing credit scoring model

ii. To develop a hybrid deep learning model that integrates behavioral and traditional data for predicting credit scores.

iii. To Validate the developed model

iv. To develop a web-based tool to visualize the model developed

1.6 Research Questions

i. What is the role of behavioral and traditional data in a credit scoring model?

ii. How can a credit score prediction model be developed using deep learning, integrating traditional and behavioral data?

iii. How can a credit score prediction model be validated?

iv. How can a web-based tool integrating behavioral and traditional data for enhanced credit scoring be developed?

1.7 Justification of the Study

This research has solved the shortcomings of traditional credit scoring techniques; there will be fairness in lending, particularly when evaluating those with scant credit histories or who mostly make cash transactions and unbanked customers. Losses in financial institutions will also be reduced since the model will enable lenders to make sound decisions. The study suggests a deep learning-based hybrid model that combines traditional and behavioural data utilizing Ensemble techniques and Feature engineering to address these issues.

The study focuses on young people and those with limited credit histories and aims to provide a more thorough credit risk assessment by integrating expenditure and payment histories. The ultimate goal is to create a sophisticated credit-scoring model that improves credit choices by making them more precise and inclusive of varied populations, therefore advancing credit-scoring methodology and ensuring fairer and reliable credit assessments in the financial industry.

1.8 Limitations of the Study

The limitations have to be overcome to construct the hybrid RNN+DNN model for credit scoring. The quality and representativeness of the training and validation datasets had a big impact on the model's performance. Data preprocessing and feature selection posed the biggest challenge in this research.

1.9 Significance of the Study

The study benefits the following groups;

People with Limited Credit History: Those who primarily use cash for purchases or have limited credit histories will benefit from a more accurate credit evaluation. The hybrid model gets beyond the drawbacks of traditional approaches, allowing for a full assessment of credit risk and giving them access to better loan alternatives.

Financial Institutions: The research helps financial institutions make educated lending decisions by offering a more complex credit rating model. This helps reduce costs brought on by erroneous evaluations and risky loans, improving the general health of the economy and profitability.

Decision-Makers in the Credit Industry: Lenders and credit decision-makers are given a potent tool by the suggested deep learning-based hybrid model to more precisely assess creditworthiness. This gives them the freedom to choose wisely when lending, lowering the risk of default and enhancing the performance of the loan portfolio.

Policymakers and Regulatory Organizations: The study's focus on ethical lending practices is consistent with regulatory objectives. The research's findings help guide policy choices and promote the adoption of more inclusive and unbiased credit evaluation techniques for the banking industry.

Academia and other scholars: The creation of robust credit scoring models advances research in data science, machine learning, and finance by providing fresh approaches and perspectives on credit risk rating. It allows academics to investigate the role of technology in finance, tackling issues such as interpretability and fairness. These models promote innovation, act as standards for assessing new technologies, and help create more trustworthy and open financial systems.

1.10 Scope of the Study

The researcher concentrates on the developing a hybrid credit scoring model based on deep learning that combines traditional and behavioral data which will be used in the financial sector. The model aims to solve the drawbacks of traditional credit scoring methods and offer more accurate evaluations of credit risk for young people as well as those with short credit histories. The study test and validate the developed model.

The feature added to the hybrid model to make it different from others is payment behavior.

1.11 Operational Definition of Terms

i **Hybrid-** In this study, it refers to the integration of two different neural networks (RNN +DNN) to develop a more advanced model.

ii **Behavioral data -**in credit scoring refers to the use of non-traditional data sources and alternative data points to determine a person's creditworthiness.

iii **Traditional data-** in credit scoring refers to the utilization of a person's personal information and past financial data to evaluate their creditworthiness.

CHAPTER TWO: LITERATURE REVIEW

2.0 INTRODUCTION

In this chapter, the study carefully examines earlier research done by other researchers on credit scoring. The research is guided by the study's objectives: to evaluate the role of behavioral and traditional data in the existing credit scoring model, to develop a deep learning model that integrates behavioral and traditional data for predicting credit scores, and to validate the developed model. In addition, this chapter discusses the conceptual framework and the operationalization of the variables.

2.1 Behavioral and traditional data in credit scoring

The first objective of the study was to evaluate the role of behavioural and traditional data in the existing credit scoring model. Therefore, this section reviews literature related to the role of behavioural and traditional data in the credit scoring model

According to Bright (2023), behavioral data refers to the use of alternative data points, banking transaction data, and non-traditional data sources to assess a person's creditworthiness. Traditional data in credit scoring is used to assess a person's creditworthiness by using their personal information and historical financial information (Braight Technologies, 2023).

According to Illion (2022), the availability of banking transaction data to a wide range of consumer profiles and the strong correlation between credit behavior and consumer spending, budgeting, and payment behavior make it valuable. The power of bank transaction data confirms the relationship between people's inclination to prioritize their obligations and manage debts and the broader financial and consumption decisions they make. Better credit risk consumers, according to transaction data, consistently pay their bills on time, maintain a positive bank balance And avoid going overdrawn, earn a consistent income, use direct debits to pay their bills, and make their spending transparent by opting for electronic payments over cash.

To evaluate the role of behavioural and traditional data in the existing credit scoring model, there is a need to understand the variables that are commonly used in credit scoring. The key variables mainly revolve around the borrower's characteristics. These include their household income, Age, gender, educational background, occupation, place of residence tenancy, housing situation, geographical location, number of dependents, and marital status (Peprah et al., 2018)

Subsequently, the model utilizes traditional factors such as Payment history, Credit Utilization(percentage of credit that is presently being used, computed by dividing the total sums owed on all of your credit cards by the sum of all of your credit limits), Applications of New Credits, Length of credit history(your accounts' average Age, your oldest account ages or the Age of your newest accounts and whether or not the accounts have been utilized recently) and Credit Mix(Your credit mix accounts for the various credit accounts you have, including credit cards, would miss a payment) (Kumar& Joshi, 2023).

2.1.1 Behavioral data

Behavioral data refers to the use of alternative data points and non-traditional data sources to assess a person's creditworthiness. It also considers the person's actions, behaviour, and credit history (World Bank, 2019).

According to a model by Björkegren and Grissen (2020), mobile phone usage behaviour is used to predict borrowers' credit scores. Borrowers lack financial records in developing

Nations, which makes it difficult to obtain credit. However, the ubiquitous use of cell phones provides useful behavioural information for lending. The study shows how call records tied to loan repayments can predict defaults better than traditional credit data for those with thin or no credit histories. Mobile phone usage-related individual traits have slightly stronger correlations, up to 0.16 (Björkegren& Grissen, 2020).

Influence scores are produced by using sophisticated social network analytics tools to spread the influence of previous defaulters throughout the network. Call networks are built using call-detail data. The results show that call-detail data dramatically improves the effectiveness of traditional credit scoring algorithms. Remarkably, the model that takes into account characteristics linked to calling behavior produces the largest profit (Óskarsdóttir et al.,2020).

Fintech enterprises have tried to provide the market with effective financial services that fulfill their requirements and demands due to the nature of traditional banking, which does not offer credit facilities to most people regarded as unsafe and unbanked (Huang, 2019). It was determined that only 3% of people in the Philippines borrowed money from banks, while close to 39% of those who used credit facilities did so from unofficial sources such as mobile money loans. Fintech companies must, therefore, find efficient ways to create credit scoring processes by looking beyond the traditional technique and using new sources like mobile devices. It shows that mobile phone data are readily available, easily accessed, and packed with crucial data that Fintech companies need for credit scoring. Cellular data is effective in revealing the way of life and economic activity of a borrower (Huang, 2019). Ultimately, the way people arrange their contacts using first and last names and how they construct their brief messages using proper syntax and punctuation might be helpful as data points in the credit scoring model.

According to Suthanthiradevi et al. (2021), the risk assessment process in the model behavioral scoring system for loans using Twitter involves generating a credit score while taking certain financial aspects into account. They recommend creating a behavioral score with information from social media. The danger of using traditional evaluation models is reduced when a person's behavioral and credit scores are combined. The behavioral score is influenced by the tweet score, profile score, and financial attitude. The information that was obtained from Twitter is used to calculate a score for the whole profile. The usefulness, regularity, and truthfulness of a person's tweets are only a few factors that go into calculating their Twitter score (Suthanthiradevi et al., 2021).

Pritchard (2019) notes that typical bank credit scores frequently need to account for expenses for things like power, rent, shopping, and insurance. However, information from these transactions is useful for Fintech and startup financial companies, mainly when it is collected from phone records. Pritchard claims that Fintech companies use this data to measure an individual's credit risk and have an impact on loan choices. Families and individuals make many monthly payments to various entities. Utility bill default might affect credit scores, making getting loans from Fintech companies more challenging. Phone-based data was discovered to be a significant predictor of an adverse outcome.

2.1.2 Traditional data

In markets where credit scoring models based on traditional data sets are employed, a potential borrower must have access to sufficient historical credit data to be deemed score-able. Without this data, constructing a credit score is impossible, and an applicant who may have good credit is frequently denied access to loans with reasonable terms (World Bank, 2019).

According to Ampountolas et al. (2021), it is challenging to assess an individual borrower's creditworthiness in the absence of a credit history. With the aid of actual micro-lending data, this study demonstrates that Age, Log Amount, Annualized Rate, and Number of Repayments are numerical properties using freely available consumer data. Categorical features like gender, marital status, and frequency are classified and they produced an accurate estimate of each applicant's creditworthiness by factoring in their Age, income, monthly expenses, sector, and ability to pay in installments. Payments, income, and Age all provided low features' ability to predict outcomes in a dataset - information value (IVs) during the development phase, whereas monthly expenses, the sector, and installments all produced high IVs. Low IV variables were permitted since they significantly affected the state of the account. In the absence of credit histories or centralized credit databases, this provides low-cost and reliable ways for micro-lending groups in developing countries to assess borrower creditworthiness (Ampountolas et al., 2021).

Luo (2020) lays out the foundation for a strong decision-support strategy in credit scoring. The researcher explores a novel method for improving prediction accuracy that includes both textual and numerical data elements. The study's main emphasis areas include the country of headquarters, industry, stock returns, changes in Credit Default Swaps (CDS) spreads, price-earnings (P/E) ratio, price-to-book ratio (P/B), Enterprise Value/ Earnings Before Interest, Tax, Depreciation, And Amortization (EV/EBITDA) ratio, and dividend yield ratio. After careful investigation of the data, interesting conclusions are drawn. The use of a group of classifiers routinely surpasses the use of a single classifier, resulting in greater and more reliable classification accuracy in a variety of contexts. This collaborative strategy shows its superiority in improving prediction quality, mainly when used in the majority of instances (Luo, 2020).

Sum et al. (2022) introduce a cutting-edge and effective personal loan-specific credit rating model. Incorporating major elements that definitively establish an applicant's creditworthiness is essential to this model's efficacy. These include elements like installment kind, Age, regular expenses, employment industry, mode of payment, and the essential income-to-finance ratio. The inclusion of these variables guarantees that the model's assessment of the applicant's creditworthiness is quite accurate. A summary of variables used in the existing credit scoring model with their strengths and limitations is shown in Table 2.1.

Table 2. 1 Summary of variables used with their strengths and limitations.

S/N	Title	Authors& Year	Variable Used	Strengths	Limitations
1	Modeling a Behavioral scoring system for lending loans using Twitter	Suthanthiradevi et al.(2021)	Twit score	Reduces risk by combining behavioral and credit scores; Uses social media data to assess financial attitudes	May not represent overall financial behavior; Social media data can be manipulated or misrepresentative
2	Behavior revealed in mobile phone usage predicts credit Repayment	Björkegren &Grissen(2020)	Mobile phone transaction records	Provides insights into lifestyle and financial behavior; Readily available and easily accessible data	Limited to mobile phone users; May overlook traditional financial indicators
3	The Value of Big Data for Credit Scoring: Enhancing Financial Inclusion	Óskarsdóttir et al(2020)	Call-detail records	Significantly improves traditional credit scoring models; Generates higher profits	Reliant on social network data, which may not be available for all users;

	using Mobile Phone Data and Social Network Analytics			by including calling behavior data	Privacy concerns
4	A machine learning approach for micro-credit scoring	Ampountolas et al.(2020)	Numerical features like Age, Annualized rate, Log Amount, and No of repayments. Categorical data that include Marital Status, Gender and Frequency.	Provides accurate creditworthiness estimates even without credit history; Low-cost and reliable method for micro-lending groups	Low Information Value (IV) for some features; Dependent on the availability of demographic data
5	A comprehensive decision support approach for credit scoring	Luo(2020)	Country Of Headquarter Industry Stock Returns CDS Spread Changes Price–Earnings (P/E) Price-To-Book Ratio (P/B) EV/EBITDA ratio Dividend Yield Ratio	Improves prediction accuracy by using a collaborative classifier approach; Effective in various contexts	Complexity in integrating diverse data sources; Requires sophisticated data processing capabilities
6	A New Efficient Credit Scoring	Sum et al.(2022)	Account number, Age, Amount, dependent Education,	Highly accurate assessment of creditworthiness;	May require extensive data collection; Specific to personal loans,

	Model For Personal Loan Using Data Mining Technique for Sustainability Management		Borrower’s, education level Gender, Sex of borrowers, Income salary or income, Monthly expenses	Incorporates essential financial and demographic variables	limiting generalizability to other loan types
7	Retail credit scoring using fine-grained payment data	Tobback and Martens (2019)	Payment data	It reflects real financial behavior and is accessible to a broad population, including those without traditional credit histories.	challenges such as data availability, privacy concerns, and the potential for misinterpretation must be addressed to maximize its effectiveness.
8	Does Paying Utility Bills Affect Your Credit Score	Pritchard (2019)	Power bill, rent payment, shopping, and insurance bill	Captures financial behavior through utility payments; Provides significant insights for individuals without traditional credit scores	Utility payment data may not be comprehensive; Phone-based data may not fully represent financial behavior
9	Alternative credit scoring through mobile phone data	Huang,(2019)	Mobile phone data	Provides insights into lifestyle and financial behavior; Readily available and easily accessible data	Limited to mobile phone users; May overlook traditional financial indicators

The existing credit scoring model evaluated demonstrate the strengths and limitations of credit scoring based on behavioral or traditional data. Even if traditional models are trustworthy, they frequently miss behavioral indicators that are essential for estimating creditworthiness, particularly for people without credit histories. On the other hand, while behavioral data-based models such as those derived from social media or mobile phone usage can provide insightful information about a person's financial behavior, they may not be entirely accurate, and they may be vulnerable to manipulation or privacy issues. These results guided the creation of a hybrid model that combines the best features of both methods by integrating traditional data (such as income, credit history, and demographic information) with behavioral data (such as payment behavior, credit mix, and social media activity).

The accuracy of creditworthiness evaluations is increased by this hybrid model, which offers a broader view of a person's financial behavior, especially for those with a short credit history. It also improves financial inclusion by providing credit to people who might not have been eligible for it under traditional credit scoring techniques. Moreover, because it can be applied to a variety of financial situations, it can be used with a range of loans and financial products. With its more inclusive and accurate assessments, this model is a significant breakthrough in credit scoring that can benefit financial institutions as well as borrowers.

2.2 Credit scoring model development techniques

The second goal of the study was to develop a deep learning model that integrates behavioural and traditional data for predicting credit scores. Therefore, this section reviews literature related to the development of the credit scoring model.

In recent years, significant advancements in credit scoring development have been driven by factors such as increased data availability, technological advancements, and evolving consumer behaviour, as highlighted by Hussain et al. (2019). These improvements are supported by enhanced access to a broader spectrum of data, more excellent processing capabilities, growing demand for efficiency gains, and economic expansion. Consequently, both the adoption and diversity of credit scoring have seen substantial growth (Hussain et al., 2019).

Moreover, credit scoring has evolved beyond traditional decision-making processes like approving or denying credit applications to encompass additional aspects of the credit cycle. This includes pricing financial services based on customers' or businesses' risk profiles and establishing appropriate credit limits (World Bank Group, 2019).

According to Hussain et al. (2019), when determining a credit applicant's creditworthiness, many credit scoring techniques are used, where every loan customer's credit score is created using the information they submitted, and these scores are used to differentiate between bad loans and good loans. Any lending organization typically divides credit ratings into statistical and judgmental categories. By aggregating a few critical characteristics of a loan applicant, a credit scoring model may assess a customer's creditworthiness (Hussain et al., 2019). According to Hussain et al. 2019, two approaches, the statistical approach and the judgment approach, can be employed to obtain the outcomes or scores generated by the scoring systems.

2.2.1 Judgment Approach for Credit Scoring

Using a judgmental credit analysis, lending decisions are made based on the lender's expertise rather than a specific credit score methodology. It comprises assessing the borrower's application and regulating credit approval based on previous dealings with applicants who meet comparable criteria. This methodology does not base approval decisions on any rules or empirical methods. A judgmental scoring methodology is based on traditional credit analysis practices. To create an overall credit score, a number of variables are assessed and weighted, including payment history, bank and trade references, Age, business size and kind, place of origin, and financial statements (Sujaini, 2023).

In judgmental scoring systems, according to Husain, 2019, the borrower is given points or weights depending on particular qualities; they are then weighted and translated into a score, which determines whether or not to provide a loan. The credit officer's final conclusions are based on his knowledge, common sense, and unambiguous numerical support.

Sujaini (2023) asserts that the well-known five Cs are beneficial in figuring out a borrower's creditworthiness. They consist of Character (history and reputation of the borrower), Capital (Contribution of the Borrower to the Investment), Collateral (Assurances to support the loan in the event of default), Capacity (financial capacity of the borrower to repay the loan) and Condition (The overall financial status of the borrower) are useful in determining a borrower's creditworthiness.

According to Hussain et al. (2019), loan decisions are still made using evaluating techniques that are based on both human experience and scant or unstructured data. According to Munguti & Ngali (2020), Lending institutions' reluctance to use credit-scoring systems might be due to three factors: a desire to keep highly skilled credit officers, possible model problems, and a need for quantitative credit management skills. Credit-granting procedures do not make use of statistical systems. Due to the expansion of financial institutions and large volumes of data generated Judgmental approaches are replaced by statistical methodologies that have evolved (Munguti & Ngali, 2020)

2.2.2 Statistical Approach in Credit Scoring

Banks and other financial institutions have been forced to dramatically improve their credit policies as a result of the explosive expansion of consumer lending and technological improvements in recent years. This development entails changing how creditworthiness is assessed and using more cutting-edge techniques to solve the shortcomings of their traditional methodologies. This change signaled the introduction of statistical approaches for credit evaluation (World Bank,2019).

According to Dastile et al. (2020), using statistical credit scoring has many advantages. He showed how lenders are becoming more aware of how statistical credit scoring might be a superior option to subjective, inconsistent, and unreliable procedures. Before the advent of the necessary computer technology in the early 1960s, quantitative and statistical methods were rarely applied. The development of the credit-scoring system, an impartial framework for choosing credit, alleviated the economic pressures that had made this worse. The statistical or operational methods used in today's credit-scoring include decision trees, linear discriminant analysis (LDA), Naïve Bayes (NB) and Logistic Regression (LR) (Dastile et al., 2020)

The main difference between machine learning (ML) and statistical techniques is that ML techniques focus on creating systems that can learn directly from the data that is already present, whereas statistical techniques focus on analyzing already-existing data and studying their relationships while making assumptions in order to predict an outcome (Aniceto, 2020).

2.2.3 Machine learning algorithms

Machine Learning (ML) is a branch of Artificial Intelligence (AI) that focuses on generating algorithms and statistical models that let computers learn from data and enhance their performance on a given job without being explicitly programmed. It is a strong and adaptable tool with several uses in many different industries (World Bank,2019). In general, there are two types of ML algorithms: supervised and unsupervised learning algorithms. Unsupervised learning methods use unlabeled datasets to train models, as opposed to supervised learning algorithms, which use labelled datasets. Dastile et al. (2020) claim that credit scoring is a supervised learning issue that focuses on binary classification with the goal of separating good and bad borrowers. The main objective of developing a credit-scoring model is to determine the best classification techniques that can differentiate between good and bad credit and, consequently, predict new loan applications. Credit-scoring models are widely employed in the financial sector, and more significantly in the banking sector.

2.2.4 Supervised Machine Learning Algorithms

When classifying data or making predictions, supervised machine learning is typically employed, whereas unsupervised learning is typically used to identify patterns within datasets. The requirement for labelled data makes supervised machine learning substantially more resource-intensive (World Bank ,2019).

Classification and regression models are the two primary categories for supervised learning algorithms.

Classification models are utilized when a variable or output is categorical, or when it can be divided into distinct classes or categories. The aim of classification is to assign each input data point to one of these established groups. Regression models are utilized when a goal variable or output is continuous, or when it might have a variety of numerical values. Regression aims to predict a numerical value based on input data and discover the link between these features and the target variable (Johnson,2023).

When it comes to credit scoring, classification models are frequently employed to group applicants into distinct risk groups. For instance, applicants could be categorized as defaulters or not. Classification models produce outcomes that are simple to grasp, evaluate, and employ in decision-making. When categorizing and assessing risks is the main focus, they are accommodating. Examples of algorithms that may be used for credit scoring classification tasks include LR, Decision Trees (DT), Random Forest (RF), Support Vector Machines (SVM), and Ensembled machine learning models.

2.2.4.1 Logistic Regression

Logistic regression (LR) is a popular and efficient classification model that is often applied to solve binary and multiclassification problems. LR frequently limits the output to a specified range using the sigmoid function based on linear regression. The logistic regression strategy has obvious advantages when dealing with large volumes of data, and the gradient descent method drastically cuts down on computation time. The model may be produced with accurate parameters after training with data, and it makes sense and is reliable. (Wu and Pan, 2021).

2.2.4.2 Decision Trees

DT trees classify instances by arranging them in ascending order based on the feature values. Every node in a decision tree represents a feature in an instance that has to be classified, and every branch of the tree shows a value that a node in the tree can assume. Instances are grouped and classified based on the values of their features, starting at the root node. In decision tree learning, a predictive model is used in data mining and machine learning, where observations about an object are mapped to conclusions about the item's target value. More descriptive terminology like regression trees or classification trees are occasionally used to refer to these tree models. Most of the time, decision tree classifiers evaluate the decision trees' efficacy using post-pruning techniques following their pruning using a validation set. Any node can be removed, and its most common class can be assigned by the sorted training instances. Problems involving regression and classification can both be solved using this method. The decision trees readily capture the non-linear correlations between the selected features and the objective variable. Although there is a benefit to this model in that it requires minimal preparation of the data, it is also prone to overfitting, meaning that small changes in the data can have a big effect on the entire tree (Sharma, 2021).

2.2.4.3 Support Vector Machines

SVM is a standard model for binary classification. The foundation of the SVM model is a linear classifier with the largest interval in feature space. The essential idea of the SVM method is to solve a hyperplane that can split the training dataset accurately and maximize the geometric separation between the data. The input needs to be converted into a linear classification problem in a particular feature space using nonlinear techniques before the linearly separable support vector machine model for the nonlinear classification problem can be solved. This section uses a nonlinear support vector machine approach due to the large volume of credit data and the nonlinear relationship between all of the data. To compute a nonlinear support vector, two key ideas are the penalty coefficient C and the kernel function (Wu & Pan, 2021).

A hyperplane is used by SVM to divide data. If the separation fails, a kernel trick is used to raise the dimension to the point where the data points can be split by a hyperplane (Sharma, 2021). This algorithm's strengths include its better prediction accuracy and resistance to outliers. This model's drawback is that it performs poorly when dealing with big datasets or noisy data (Sharma, 2021).

2.2.4.4 Random Forest

RF is a flexible ensemble learning technique that can be applied to both regression and classification. Experts in regression call it "Random Forest Regression (RFR)." RFR is an extension of the RF approach that is meant to predict a continuous numerical output (Sharma, 2021). Sayjadah et al. (2018) found that while assessing the variable in forecasting credit default, RF outperformed decision trees and LR regarding accuracy and area under the curve. This result shows that RF best describes the factors that should be considered when assessing the credit risk of credit card customers, with an accuracy of 82% and an Area under Curve of 77%. Because of sufficient sampling, the RF approach performs effectively when categorizing large-scale credit data. The division of DT improves the model's ability to avoid overfitting. The technique may be applied to continuous and discrete data and has better reliability (Wu & Pan, 2021).

2.2.4.5 Ensembled model

The ensemble model is a machine-learning approach combining many models' predictions to improve forecasting accuracy and robustness. Utilizing the ensemble's collective intelligence seeks to reduce any mistakes or biases that may be present in individual models. Typically, supervised learning environments use ensemble machine learning models, incorporating strategies like bagging (Bootstrap Aggregating), boosting, and stacking. When a model is trained on labelled data, supervised learning is used, and the input characteristics are linked to matching target labels or outcomes (Singh, 2023)

2.2.4.5.1 Bagging

Bagging uses the training data's distinct bootstrap samples (random subsets with replacement) to train numerous instances of the same essential learning algorithm. Each base model picks up information from a slightly different angle on the data. Random sampling adds variety, which helps prevent overfitting and recognizes various patterns in the data. In most cases, the predictions of several models are averaged (for regression) or decided by a majority vote (for classification) to get the final forecast. As an illustration, the well-known ensemble technique RF employs Bagging with DT as its primary model (Singh, 2023).

2.2.4.5.2 Boosting

Boosting involves training many base models one after the other, with each new model aiming to correct the errors made by the previous one. Data items that earlier models misidentified are given more weight in subsequent training cycles. Boosting highlights the value of improperly categorized data items for improving overall performance by assigning different weights to individual data points. Boosting seeks to turn a weak learner (a model that is just marginally more accurate than random guessing) into a strong learner (a highly accurate model). Popular boosting algorithms include Gradient Boosting and AdaBoost (Adaptive Boosting) (Singh, 2023).

2.2.4.5.3 Stacking

By using an ML technique such as linear regression or another, stacking combines many base models by learning a meta-model from the predictions of the base models. The meta-model develops the most effective way to meld the forecasts. Stacking usually takes place on two layers. The base models generate predictions based on the training data and make up the first level. The meta-model at the second level predicts the result using these predictions as input characteristics. Stacking is a very versatile method since it allows the selection of several base models and meta-models. The choice of base and meta-models depends on the particular job and may be applied to various machine-learning issues (Singh, 2023).

2.2.5 Deep Learning Algorithms

Deep learning is a subfield of artificial intelligence and machine learning that uses multi-layered artificial neural networks to process and analyze large volumes of data and create patterns that can be used to make decisions. These networks function and are set up much like the brain. In essence, three or more neural networks are used in deep learning. These networks cannot "learn" from massive datasets in the same way that the human brain does, despite their best efforts to mimic the structure and functions of the brain. A single-layer neural network can approximate predictions, but by fine-tuning the output, more hidden layers increase accuracy (Team IBM Data and AI, 2023).

Neural networks, a subset of machine learning (ML), also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are the foundation of deep learning approaches, according to IBM Data and AI Team (2023). Figure 2.1 shows that these neural networks comprise node layers consisting of an input layer, one or more hidden layers, and an output layer. In deep learning, "deep" refers to the quantity of layers in the neural network. A neural network is called a deep-learning algorithm if it contains more than three levels, including the input and output layer

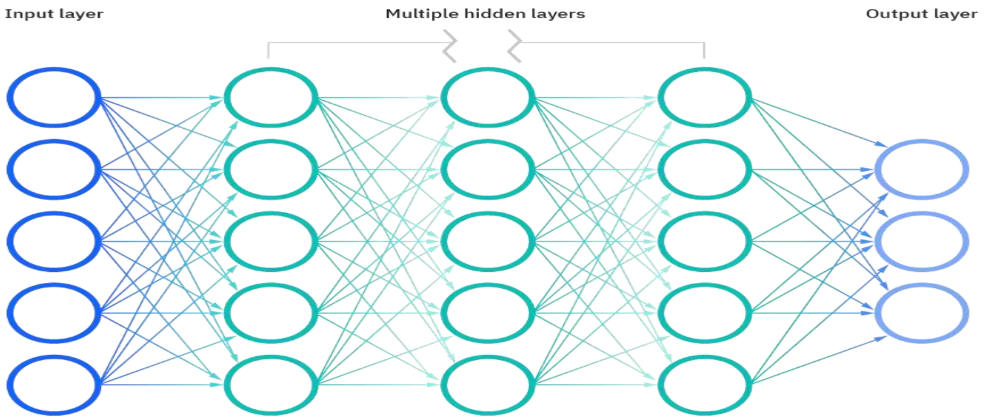


Figure 2. 1 Structure of Deep Neural Network

Deep neural networks comprise several layers of nodes, each building upon the one before it to enhance and optimize prediction or categorization. Forward propagation refers to the way calculations go across a network. A deep neural network's input and output layers are referred to as visible layers. The final prediction or classification is generated at the output layer after the deep learning model has processed the data in the input layer (IBM et al., 2023).

Backpropagation is a technique for modifying the weights and biases within a function by iteratively going backward through the layers to train the model, according to IBM Data and AI Team (2023). In backpropagation, methods like gradient descent are employed to quantify prediction mistakes. A neural network can make predictions and quickly fix mistakes thanks to the combined effects of forward propagation and backpropagation, which gradually improves the algorithm's accuracy. Deep learning algorithms can be categorized as unsupervised, supervised, or hybrid based on whether they are trained to meet particular goals, as shown in Figure 2.2 (Benavides et al., 2020)

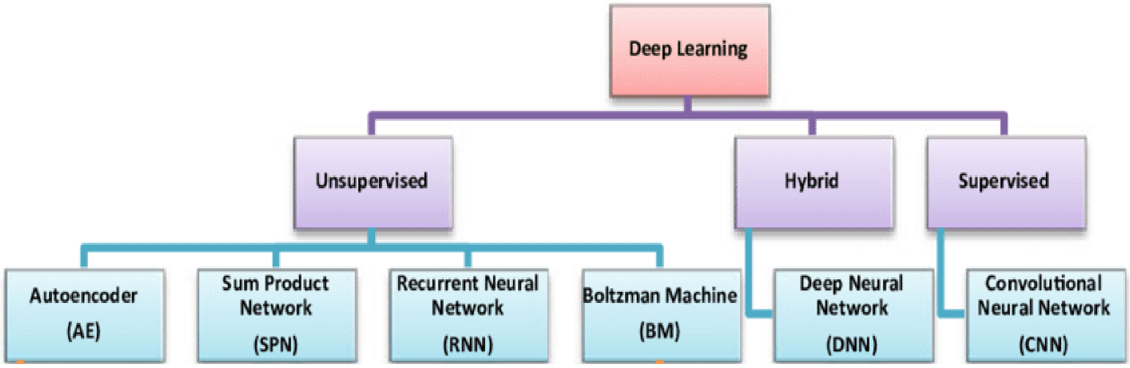

Deep Learning
Unsupervised
Hybrid
Supervised
Autoencoder (AE)
Sum Product Network (SPN)
Recurrent Neural Network (RNN)
Boltzman Machine (BM)
Deep Neural Network (DNN)
Convolutional Neural Network (CNN)

Figure 2. 2 Deep learning approach classification

2.2.5.1 Supervised deep learning models

In supervised deep learning, neural networks are trained with labelled data, a subset of machine learning. In supervised learning, the algorithm gains the ability to anticipate outcomes or categorize data based on input-output pairings supplied throughout the training phase. They mainly tackle classification and regression-related problems (Delua,2021). Convolutional neural networks (CNN), which utilize many input-output pairings, are a successful supervised deep architecture.

2.2.5.1.1 Convolutional neural networks

Convolutional Neural Networks (CNNs) are a subset of deep neural networks (DNNs) created primarily for processing structured grid data, including pictures and video. They have transformed computer vision and are often used for picture categorization, object identification, facial recognition, and more. CNNs are renowned for their ability to automatically deduce hierarchical features from input data, making them particularly useful for jobs requiring visual patterns (Selvaganapathy et al.,2018).

2.2.5.2 Unsupervised deep learning models

Unsupervised deep learning is a kind of machine learning in which neural networks are trained on unlabeled data without input-output pairings serving as explicit supervision. Unsupervised learning is concerned with identifying patterns, connections, or representations within the data, unlike supervised learning, which trains the model to predict or categorize using labelled data. Unsupervised deep-learning approaches are very beneficial for applications like data clustering, dimensionality reduction, and generative modelling (Delua,2021). The unsupervised models include;

2.25.2.1 Autoencoders

Autoencoders are neural networks that frequently reconstruct input data while learning a lower-dimensional representation. The standard training strategy, in which a network learns to anticipate particular output values for given inputs, is diverged by autoencoders (AEs). An autoencoder's goal is to rebuild its inputs, and it does this by using a two-part structure that consists of an encoder and a decoder. The encoder transforms the input vector, which creates a compressed code by passing it through several smaller hidden layers. The decoder then tries to recreate the original input vector from this compressed code while keeping the sizes of the input and output vectors. Reducing the reconstruction error is the goal of the AE optimization process (Selvaganapathy,2018).

2.2.5.2.2 Boltzmann Machines

A Boltzmann Machine is an artificial neural network that can learn and express complicated probability distributions across its collection of binary-valued patterns. This kind of probabilistic graphical model is capable of capturing intricate data relationships. They may be applied to feature learning, dimensionality reduction, and collaborative filtering (Karagiannakos,2020).

2.2.5.2.3 Recurrent Neural Networks

Recurrent neural networks (RNN) are a potent deep generative architecture for modeling and producing sequential data. The RNN's depth may be calculated based on the length of the incoming data sequence. The "vanishing gradient" issue compromised the network's appropriate training. Nowadays, the "vanishing gradient" issue is resolved by training

RNNs using improved optimization approaches that alter stochastic gradient descent (Karagiannakos,2020).

Recurrent Neural Networks (RNNs) provide several significant benefits for the development of credit-scoring models. When it comes to processing and evaluating sequential data, such as payment transactions or transaction sequences, RNNs are especially good. Their proficiency lies in their ability to discern the temporal correlations and trends within data throughout time, an essential skill for comprehending an individual's financial conduct and forecasting their creditworthiness in the future (Sherstinsky, 2020).

RNNs can also benefit from historical financial behavior and long-term patterns because of their ability to retain knowledge across time steps. By adding previous data into predictions, this capacity to manage long-term dependencies helps to create credit scoring models that are more accurate (Sherstinsky, 2020). Additionally, RNNs can adjust to shifting patterns in data over time, which is critical in financial contexts where risk variables and credit behavior might alter. Their dynamic learning capability allows them to adjust predictions based on the most recent data, enhancing their relevance and accuracy (Xiao & Zhou, 2020).

Further, RNNs can store and exploit historical data by using their internal state, which makes them appropriate for applications requiring historical context knowledge. This characteristic makes it possible for RNNs to take into account the order and timing of financial events, resulting in a more sophisticated evaluation of credit risk (Sherstinsky, 2020). Lastly, RNNs are excellent at predicting future events based on past sequences, which improves the forecasting of future financial behavior on the part of borrowers. This skill increases overall credit score accuracy and facilitates better-informed credit judgments (Xiao & Zhou, 2020).RNNs improve credit scoring models overall by efficiently identifying and utilizing historical data and temporal trends, leading to assessments of credit risk that are more precise and contextually relevant (Sherstinsky, 2020; Xiao & Zhou, 2020).

2.2.5.2.4 Sum-Product Network

Sum-product networks (SPNs), a particular category of deep architectures, are directed acyclic networks used to compute the partition and marginal functions of intricate, high-dimensional probability distributions. The leaves of the graph in SPNs are the underlying data, while the nodes in SPNs are internal components of both sum and product processes. SPNs are used for various practical purposes, with image completion only one of them (Karagiannakos,2020).

2.2.5.3 Hybrid deep learning models

Hybrid deep learning architectures or hybrid deep learning architectures integrate aspects of several deep learning architectures and Machine learning models. These models frequently outperform single-model techniques by combining the advantages of several different strategies to handle complicated problems (Qaid,2021).

Credit risk prediction is crucial for banks and financial institutions to minimize lending risks and prevent financial losses. Recent advancements in artificial intelligence (AI) have led to the development of hybrid prediction models that combine traditional statistical methods with modern AI techniques, enhancing predictive capabilities. Chi et al. (2019) explored various hybrid models that integrate logistic regression (LR) with different neural networks, such as Radial Basis Function Networks (RBFs), Deep Neural Networks

(DNNs), Adaptive Neuro-Fuzzy Inference Systems (ANFISs), and Multilayer Perceptrons (MLPs). They also examined combinations of Discriminant Analysis (DA) and Decision Trees (DT) with these neural networks, assessing the performance of 16 hybrid model combinations. The study found that these hybrid models consistently outperformed traditional models across ten performance parameters, validated using five real-world credit-score datasets. By combining the strengths of traditional statistical methods with AI, these hybrid models offer more accurate and reliable credit risk predictions, helping financial institutions make better-informed lending decisions and reducing potential financial losses. This innovative approach also addresses common issues like overfitting and data sparsity, making it a significant advancement in the field of credit risk assessment (Chi et al., 2019).

Uthayakumar et al. (2020) present a two-stage cluster-based classification model for Financial Crisis Prediction (FCP) designed to improve classification performance and adaptability across diverse datasets. In the first stage, the model employs an improved K-means clustering algorithm to refine the data by eliminating incorrectly clustered instances. This step ensures that the data is more accurately categorized before further analysis. Following this, a rule-based model is constructed to fit the refined dataset, enhancing its suitability for specific classification tasks. In the second stage, the Fitness-Scaling Chaotic Genetic Ant Colony Algorithm (FSCGACA) is applied to optimize the parameters of the rule-based model. This hybrid approach aims to combine the strengths of clustering with the optimization capabilities of FSCGACA to achieve superior performance. The proposed model was tested on three benchmark datasets—the qualitative bankruptcy dataset, Weislaw dataset, and Polish dataset—and demonstrated better classification accuracy and adaptability compared to other models, making it more appropriate for a variety of datasets (Uthayakumar et al., 2020).

The hybrid model presented by Nalić et al. (2020) is designed to enhance credit scoring prediction by combining advanced feature selection techniques with ensemble learning methods. The model begins by preprocessing the dataset and applying five different feature selection algorithms to identify the most relevant features. These algorithms' results are then aggregated using various voting methods, including a novel "if_any" voting technique, which was found to outperform traditional methods. Following feature selection, four different classification algorithms generalized linear models (GLM), support vector machines (SVM), naïve Bayes, and decision trees (DT) are used on the refined dataset. These classifiers are subsequently combined into eight distinct ensemble models using a soft voting approach. The experimental results demonstrate that the hybrid model, particularly the combination of features selected by the "if_any" voting method and the GLM + DT ensemble, offers superior performance in credit scoring prediction, outperforming both single classifier models and other ensemble approaches. This hybrid data mining model not only improves predictive accuracy but also provides a more robust and reliable tool for decision-making in credit risk assessment (Nalić et al., 2020).

Zhang et al. (2021) propose a new hybrid ensemble model for credit scoring that enhances predictive performance through a series of innovative steps. The model begins with a voting-based outlier detection method that enhances traditional outlier detection algorithms by integrating outlier scores using a weighted voting mechanism, thereby creating an outlier-adapted training set. This step ensures that noise-filled data does not mislead classifier training. To address the issue of imbalanced data, the model introduces a bagging-based balanced sampling method, which enhances traditional under-sampling by dividing the dataset into parallel subsets, performing random under-sampling, and producing a balanced training set. To further optimize the model, a stacking-based ensemble approach is employed, where the parameters of base classifiers are adaptively optimized, and these optimized classifiers are then combined into a multi-stage ensemble model. Finally, the ensemble model is used to predict the test set outcomes, with the results aggregated through a soft voting mechanism. The proposed model's effectiveness and robustness are validated using five datasets from the UC Irvine machine learning repository, demonstrating its superior predictive power in credit scoring (Zhang et al., 2021).

The hybrid model proposed by Roy and Shaw (2023) for credit scoring of Small and Medium Enterprises (SMEs) is developed through a multi-stage process that integrates the Analytic Hierarchy Process (AHP) and the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS). The development of this model begins with the identification of relevant credit rating criteria and sub-criteria, which are determined through an extensive literature review and consultations with industry experts. In the second stage, AHP is employed to calculate the weights of these criteria and sub-criteria, reflecting their relative importance in the credit scoring process. Finally, in the third stage, these AHP-derived weights are applied within the TOPSIS framework to determine the credit scores of SMEs. The use of TOPSIS enables the ranking of SMEs based on their proximity to an ideal solution, effectively distinguishing between potential creditworthy candidates and those with higher risks. This hybrid AHP-TOPSIS model is presented as a low-cost, customizable solution that can aid financial institutions in making informed credit decisions for SMEs, particularly in scenarios where financial data is unorganized or limited (Roy & Shaw, 2023).

Machado and Karray (2022) develop a hybrid credit risk model for commercial customers by combining unsupervised and supervised machine learning (ML) algorithms. The process begins with the application of unsupervised learning techniques, specifically k-Means and DBSCAN, to cluster customers based on various features, which helps mitigate overfitting issues commonly associated with supervised algorithms. Following this clustering, several supervised ML algorithms such as Adaboost, gradient boosting (GB), support vector machines (SVM), decision trees (DT), random forests (RF), and artificial neural networks (ANN) are applied to predict the final credit scores. The performance of these hybrid models is evaluated using metrics like mean absolute error (MAE), explained variance (EV), and mean squared error (MSE). A key innovation in this approach is the inclusion of past credit scores as additional features, which enhances the predictive accuracy of both individual and hybrid models. This hybrid model not only demonstrates superior predictive performance compared to traditional individual models but also offers a more nuanced approach to credit risk assessment, particularly for North American commercial customers (Machado & Karray, 2022).

Liu et al. (2022) propose a two-stage hybrid model to enhance the accuracy of credit risk prediction by leveraging advanced machine learning techniques. In the first stage, XGBoost is utilized to linearize and transform the complex, nonlinear features in the credit data into a high-dimensional sparse feature matrix. This transformation helps in making the classified information within the data more accessible and manageable. In the second stage, the transformed high-dimensional data is processed using a graph-based neural network called forgeNet, which is particularly effective in handling high-dimensional data and uncovering relationships between features. The model's robustness was validated using real-world credit data from Lending Club, spanning various economic cycles between 2007 and 2016. The results demonstrate that the proposed model outperforms other models, achieving average prediction results of 87.52% in accuracy, 93.13% in F1-score, and 85.59% in G-mean. This indicates the model's superior performance in credit risk prediction, particularly when integrating feature transformation with feature graph mining, and highlights its resilience across different economic conditions (Liu et al., 2022).

After reviewing the literature on various hybrid credit risk prediction models, it is evident that combining multiple machine learning techniques enhances predictive accuracy, adaptability, and robustness compared to single-model approaches. The models analyzed integrate traditional statistical methods with advanced AI techniques, such as logistic regression with neural networks or clustering with genetic algorithms, to address issues like data imbalance, overfitting, and the complexity of credit data. These hybrid models have demonstrated superior performance across diverse datasets and economic conditions, highlighting their effectiveness in credit risk assessment.

Given these advancements, developing a hybrid model that integrates Recurrent Neural Networks (RNNs) with Deep Neural Networks (DNNs) is a worthy endeavor, particularly when incorporating both traditional financial data and behavioral data. RNNs are adept at capturing temporal dependencies and patterns within sequential data, making them well-suited for analyzing behavioral data that unfolds over time. DNNs, on the other hand, excel in handling complex, high-dimensional data, making them ideal for processing traditional financial indicators. By combining these two architectures, the hybrid model can leverage the strengths of both RNNs and DNNs, offering a more comprehensive and accurate prediction of credit risk. This integration not only enhances the model's predictive power but also addresses limitations observed in previous hybrid models, such as the need for better handling of temporal and non-linear relationships within the data. Thus, the proposed hybrid RNN-DNN model represents a significant advancement in credit risk prediction, offering a more nuanced and reliable tool for financial institutions to make informed lending decisions.

2.2.5.3.1 A Deep Neural Network

A Deep Neural Network (DNN) is a multi-layered network that, during pre-training, makes use of the Deep Belief Networks' (DBNs') generative model. Backpropagation is used for categorization and fine-tuning. DL methods are skilled in spotting complicated inborn patterns in large amounts of complex data. DL can successfully detect gentle and dangerous traits within the provided dataset because of its ability to abstract information (Qaid,2021).

According to Benavides et al., (2020) Deep Neural Networks (DNNs) are increasingly used in credit scoring due to their ability to handle complex and high-dimensional data, making them particularly suited for the intricate nature of financial datasets. These networks can identify non-linear relationships and patterns that traditional models might overlook, enhancing the accuracy and reliability of credit risk assessments. One of the key advantages of DNNs is their ability to automatically learn feature representations from raw data. This capability allows them to extract and combine relevant features from large datasets, leading to more precise and robust credit-scoring models(Benavides et al., 2020). Moreover, DNNs are very scalable, which makes it possible for them to handle enormous volumes of data, an essential capability in the financial industry, where data volumes are frequently substantial(Xiao & Zhou,2020). This scalability aids in the creation of complete credit scoring models that may incorporate a variety of data sources, including alternative data sources like social media usage and mobile phone usage, as well as more conventional financial information like income and credit history. Because of their adaptability, DNNs may create models that include a variety of data, increasing lending choices' and predictions' accuracy (Xiao & Zhou, 2020). Additionally, by depending on data-driven insights rather than arbitrary judgments, DNNs can lessen the influence of human bias in evaluating credit. More impartial and equitable credit evaluations result from this. DNNs help to improve credit score by identifying intricate patterns and relationships in the data, which helps financial organizations better evaluate risk and make more dependable loan decisions.

According to Benavides et al.(2020), DNNs provide sophisticated skills for handling and evaluating intricate financial data, leading to more precise and trustworthy credit scoring models. They are essential tools in today's credit risk assessment because of their capacity to handle massive datasets, pick up feature representations, and combine various data sources.

DNNs have several drawbacks and restrictions. First of all, DNNs are prone to overfitting, mostly due to their inability to store and use historical data efficiently due to a lack of natural memory. A model is said to be overfitted if it performs well on training data but badly on unknown data, which reduces the model's capacity for generalization (Benavides et al., 2020). Because of this, DNNs may have trouble with jobs which includes credit risk assessment where temporal and historical context are crucial.

DNNs' dependency on massive amounts of data for efficient training is another drawback. Although they perform exceptionally well with large datasets, scant or inadequately representative data can severely impair their effectiveness. This may cause problems with the accuracy and dependability of the model.

Furthermore, DNNs need a significant amount of computational power and time for training, which can be problematic in settings with constrained hardware or when quick model updates are required. This computing requirement may make deploying DNN-based solutions more expensive and difficult.

These mentioned limitations underscore the necessity for hybrid models, which integrate the advantages of DNNs with alternative architectures, such as RNNs, to overcome these limitations. It is possible to improve credit scoring performance and minimize some of the limitations associated with standalone DNN models by merging DNNs with RNNs, which are better able to handle sequential data and long-term dependencies (Benavides et al., 2020; Sherstinsky, 2020; Xiao & Zhou, 2020).

2.2.6 Summary

Deep Neural Networks (DNNs) are Feed Forward Networks (FFNNs), where data goes from the input layer to the output layer without ever traveling backward. The links connecting the layers are one-way, forward-moving, and never come into contact with another node. DNNs are strong tools for large data and complicated tasks because these layers can train to represent data at ever-higher degrees of abstraction; however, due to their high capacity to learn complex patterns, they suffer overfitting, and they don't have a memory to remember what they started. This problem is handled with a Recurrent Neural Network (RNN), which is an FFNN has a temporal twist and can process input sequences by utilizing their internal state or memory, making it suitable for this research.

Recurrent Neural Networks (RNNs) and Deep Neural Networks (DNNs) were selected for the creation of credit scoring models because of their complimentary capabilities and special advantages in handling various aspects of credit risk assessment. RNNs are especially good at processing and evaluating sequential data, such payment histories or transaction sequences, they were chosen for this application. Because of the way they're made, they can record patterns and relationships throughout time, which is important for deciphering a person's financial behavior and forecasting their creditworthiness in the future. Because RNNs are skilled at managing long-term dependencies and preserving information over time steps, they can adapt to changing data patterns in financial contexts and incorporate past data into predictions. This makes them highly suitable for tasks where the sequence and timing of financial events are essential for assessing credit risk (Sherstinsky, 2020; Xiao & Zhou, 2020).

DNNs, on the other hand, were selected because to their capacity to process big datasets and spot intricate patterns in high-dimensional data. They can handle several features at once and are very good at describing data at higher levels of abstraction. This feature improves the model's capacity to examine complex relationships found in the credit data, which is necessary for a precise evaluation of risk. But because DNNs don't have memories and need a lot of data to train well, they can overfit (Benavides et al., 2020).

When the data is insufficiently representative or sparse, their performance may be restricted.

Utilizing the advantages of both architectures, a hybrid model that combines RNNs and DNNs is developed. DNNs offer their ability to analyze complicated, high-dimensional data, but RNNs are superior at recognizing temporal patterns and managing sequential data. By combining the strengths of both model types, the hybrid technique overcomes the drawbacks of each model type, RNNs' difficulties with long-term dependencies and DNNs' propensity for overfitting. As a result, the credit scoring model becomes more resilient and offers a thorough and precise evaluation of credit risk, improving the system's overall functionality and adaptability (Benavides et al., 2020; Sherstinsky, 2020; Xiao & Zhou, 2020).

2.3 Model Validation and Testing

Testing and validating the model developed is the third objective of this study. According to Kumar et al. (2021), testing and validating is a crucial phase in the model creation process for all machine learning models, a credit scoring model, and other kinds of models. This study Testing and validation should be done continuously throughout the development process. This assists in identifying faults and problems early, making resolution simpler and less expensive. Using a distinct hold-out dataset, the study can validate the model before deploying it in a real-world setting or applying it to decision-making. This dataset, which is different from the training set, closely resembles the real-world data the model would face. Since the model the study developed is a classification problem, Validation Metrics were employed to show the validity. The study used metrics for classification like accuracy, precision, sensitivity, specificity, recall, or F1 score (Liu, 2022).

2.4 Web-based tool integrating behavioural and traditional data for enhanced credit scoring

The study acknowledged the need to develop a user-friendly web-based application. The credit scoring model that is developed might be seen and interacted with by people and financial institutions using this tool. The development process begins with a thorough investigation and comprehension of credit scoring systems, along with identifying relevant sources of behavioural data. The procedure will smoothly transition into data collection, whereby APIs or scraping methods will be used to acquire traditional credit data and behavioural insights. Subsequently, a high-level feature engineering stage will be conducted to extract and refine features from both data types, guaranteeing strong normalization and standardization for uniformity. The training process will be driven by state-of-the-art machine learning techniques, with ongoing iterative improvements made for peak performance. The web development phase will demonstrate a robust backend infrastructure, a user-friendly interface design, and the implementation of smooth RESTful APIs (Representational State Transfer Application Programming Interfaces). Scalability and dependability during deployment will be ensured by thorough testing.

2.5 Empirical Review

Xia et al. (2022) developed a novel credit scoring model that used deep learning and ensemble classifiers, used Bayesian methods to improve macroeconomic variables, integrated these variables into the model, and added specialized measures for regulatory evaluation. These contributions represented a substantial divergence from earlier credit-scoring studies. The study showed the superiority of their heterogeneous deep forest model, the advantages of including macroeconomic variables, and the importance of regulatory-oriented evaluations in credit scoring.

In order to enhance the assessment of loan default risk in peer-to-peer (P2P) lending, this research suggests a deep neural network-based method that uses an extensive feature set that includes both category and numerical data. One-hot encoding is used to transform categorical features into numerical data. A three-layered multilayer perceptron (MLP) that addresses class imbalance is trained using TensorFlow and the Synthetic Minority Over-Sampling Technique (SMOTE). This MLP obtains a 93% test classification accuracy, which is higher than the 75% accuracy of a single-layered MLP (Duan, 2019).

Client segmentation, risk management, fraud detection, and client retention are some of the issues the banking industry faces. Machine learning (ML) and data analytics present viable answers to these problems. In order to handle credit risk analysis and client retention in banking, this study provides a model. To analyze bank customer data, we used supervised learning techniques such as artificial neural networks (ANN), support vector machines (SVM), and deep neural networks (DNN). We assessed these algorithms on the German credit dataset and obtained recognition accuracies of 72%, 72%, and 76% for the German credit dataset and 98%, 92%, and 97% for bank customer data, respectively. The suggested approach successfully raises credit risk assessment and client retention, which boosts bank profitability (Dharwadkar& Patil, 2018).

Credit risk prediction is essential for banks and other financial institutions to prevent making poor decisions that could result in lost opportunities or financial losses. According to Chi et al. (2019), Hybrid prediction models, which combine conventional and cutting-edge artificial intelligence (AI) techniques, have been established recently and offer better predictive skills than single methodologies. Hybrid models that combine multilayer perceptions (MLPs) and other AI technologies with logistic regression (LR) have been proposed by Chi et al. (2019). We compared 16 combinations of logistic regression (LR), discriminant analysis (DA), and decision trees (DT) with four different types of neural networks (NN): radial basis function networks (RBFs), deep neural networks (DNNs), adaptive neuro-fuzzy inference systems (ANFISs), and multilayer perceptrons (MLPs) in order to assess the viability and efficacy of these hybrid models. The statistical analysis and experimental findings show that the suggested hybrid models outperform other models on ten performance parameters. Five real-world credit-score datasets were used to validate the classifier(Chi et al.,2019).

Ampountolas et al. (2021) tested multiple machine learning algorithms using actual microlending data to evaluate how well they classified borrowers into various credit categories. The study showed that utilizing easily accessible consumer data and commercially available multi-class classifiers, including random forest algorithms, may complete this task. The machine learning ensemble classifiers, such as random forest, XGBoost, and Adaboost, outperformed all other models examined. Notably, the random forest classifier was marginally outperformed by XGBoost and Adaboost. Diagnostic metrics showed that the ensemble classifiers outperformed the other models on our dataset.

According to Kumar et al. (2021), the study established a credit scoring and prediction framework for the banking sector using a mix of deep learning and the K-Means algorithm. The study's suggested system incorporates a prediction model employing DL and feature selection (FS) classification approaches to improve performance. To increase the effectiveness of the study prediction model in determining credit ratings, it uses explicitly an efficient feature selection approach, the ReLU activation function for weighting, and a decision tree classifier for class labelling. The study model distinguishes between default and non-default clients with an estimated accuracy of 87%. To distinguish between default-prone and non-default consumers in the banking sector, the research prediction model combines a deep neural network with a decision tree classifier (Kumar et al., 2021)

Tyagi's (2022) study examines various machine learning models, including sequential neural networks, heterogeneous ensembles like AdaBoost and RF, and single classifiers like LR, DT, LDA, and QDA. According to the study's findings, ensemble classifiers and neural networks perform better than other models in terms of credit rating. Additionally, the study integrates LIME and SHAP, two cutting-edge post-hoc explainability methodologies. These methods are used to assess machine learning-based credit rating models, emphasising their application to open-access datasets made available by Lending Club, a P2P lending platform with headquarters in the US. The study aims to maximize loan decisions to increase investor profit and improve transparency in the selection of loans and assets. The research further investigates LIME and SHAP's suitability for describing Black-Box classifiers like neural networks (Tyagi, 2022**)**.

According to Balduini et al. (2019), credit analysts increasingly include both financial and behavioral data when assessing a borrower's credibility. However, the market hasn't yet developed a standardized method for incorporating these many data types. Our methodology formalizes the integration procedure to fill this gap.

In order to combine credit scores obtained from behavioural and financial data, we use a dynamic weighting mechanism. This combined metric performs better regarding ranking precision and prediction than individual scores. Currently, the banking sector mainly relies on two types of information: behavioural data, which includes spending and payment patterns, and financial data, which includes Income Statements, Balance Sheets, Cash Flow, and Financial Ratios (Balduini et al., 2019)

In conclusion, the models discussed above highlight how important it is for credit scoring models to include both traditional financial data and behavioural data. This integration not only conforms to evolving industry norms for banking that increasingly value both data types for establishing borrower creditworthiness but also significantly increases predictive accuracy. The combination of deep learning and ensemble classifiers, as well as the incorporation of macroeconomic variables like interest rate and the unemployment index, emphasize the significance of such integrations to improve the accuracy and relevance of credit scoring in the financial environment.

2.6 Conceptual framework

Based on the results of the literature research, we create our study's organizational framework to reflect the interactions between the independent and dependent variables that have been studied. The borrower attributes that fall into traditional data and behavioural data are the independent variables in this model as shown in figure 2.3. The borrower's creditworthiness is determined by these elements taken as a whole. Moderating variables affect the direction or intensity of the link between independent and dependent variables.

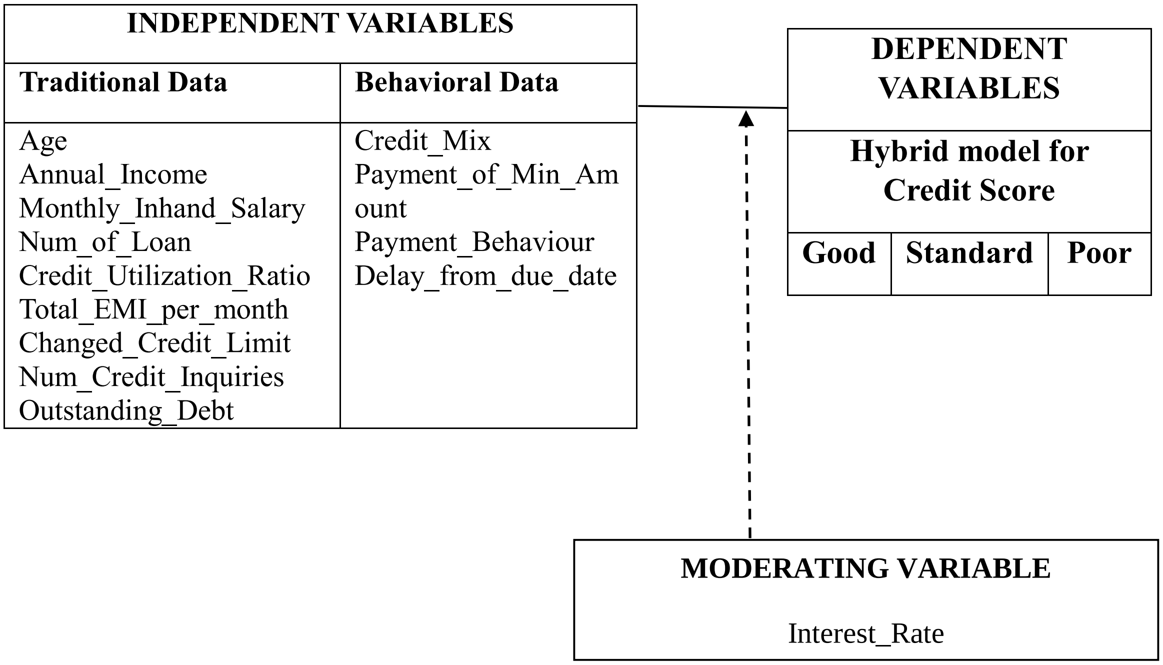


Figure 2. 3 Conceptual framework

2.6.1 Independent variables

These are variables that were used as inputs in the credit scoring model. In this research, independent variables have been categorized into two;

2.6.1.1 Traditional variables

Traditional credit scoring variables include a range of financial and personal metrics that are crucial for evaluating a borrower's creditworthiness. Key variables include Age, which can influence financial stability and borrowing behaviour; Annual Income and Monthly In-hand Salary, which reflect the borrower's income and ability to repay loans; and Interest Rate, which affects the cost of borrowing. Other variables, such as the

Number of Loans and Credit Utilization Ratio, provide insights into the borrower's current credit obligations and how much of their available credit they are using. Additionally, Total EMI per Month and Changed Credit Limit offer information on existing debt repayment commitments and adjustments in credit limits. Number of Credit Inquiries indicates how frequently the borrower applies for new credit, and Outstanding Debt represents the total amount of debt currently owed. Collectively, these variables help in constructing a comprehensive profile of a borrower's financial situation, which is essential for accurate credit risk assessment.

2.6.1.2 Behavioral variables

Behavioural data in credit scoring encompasses several critical indicators of a borrower's financial habits and payment patterns. Credit Mix reflects the variety of credit types a borrower manages, which can impact their creditworthiness. Payment of Minimum Amount indicates whether a borrower meets at least the minimum required payments, highlighting their ability to manage debt obligations. Payment Behavior tracks the consistency and timeliness of payments over time, providing insights into overall financial responsibility. Delay from Due Date measures the frequency and duration of late payments, directly assessing the borrower's reliability in meeting deadlines. These behavioral metrics provide a deeper understanding of a borrower's financial habits and reliability, complementing traditional credit data in the evaluation process.

2.6.2 Moderating variables

The credit-default risk of borrowers is moderated by the monetary authority lending rate in credit prediction models that integrate both traditional and behavioural data. Lower lending rates imply a healthy state of the economy and lower the risk of default by making credit more accessible. Higher rates, on the other hand, result in higher borrowing costs and possibly increased default risk. Furthermore, changes in lending rates reflect the state of the economy, which influences borrower behavior. The monetary authority lending rate coupled with data improves model accuracy by including macroeconomic parameters.

2.6.3 Dependent variables

These are variables that the credit scoring model seeks to predict or evaluate. Regarding credit scoring, the dependent variable is usually a binary result indicating whether a person is deemed creditworthy. It is expected to describe this binary result as High Risk (Indicates situations where, given specific conditions, there is a high chance of default) and Low Risk (Indicates situations where, according to specific standards, there is little chance of default)

2.7 Operationalization of Variables

According to Bhandari (2023), the operationalization of variables in credit scoring is establishing and measuring the crucial elements or variables used to judge a person's creditworthiness. An essential step for the lending industry is the operationalization of credit score variables, which enables lenders to offer credit while successfully managing risk based on consistent and data-driven choices (Bhandari, 2023).In this study, the following variables were used to develop the model: age, income, income, interest, Num_of_Loan, Delay_from_due_date, Changed_Credit_Limit, Num_Credit_Inquiries, credit, Outstanding_Debt, credit, Payment_of_Min_Amount, Total_EMI_per_month, Payment_Behaviour, and Credit_Score as the target variables.

2.8 Gaps identified

This section highlights the gaps found in analyzing the current credit scoring models. Although current models play a crucial role in assessing borrower creditworthiness and directing lending decisions (Shi et al., 2022), they need to be improved by several restrictions. One major obstacle to successful prediction is the existence of unfair lending practices and skewed samples, especially in microcredit situations (Ampountolas et al., 2021). Furthermore, lenders may suffer significant financial losses due to misclassifying borrowers (Aniceto et al., 2020). Additionally, consideration needs to be given to problems with variable selection and transparency in risk evaluations based on machine learning. These discrepancies highlight the need for better approaches to credit scoring systems that eliminate prejudice, increase classification accuracy, and provide more transparency.

2.9 Summary

The literature review provides a thorough exploration of the traditional and behavioral aspects influencing credit ratings, highlighting their significance in the credit scoring process. However, further research is needed to assess the comprehensiveness of these factors, questioning whether other critical aspects remain unexplored and how these elements interact in real-world scenarios.

The inclusion of alternative data sources, such as social media activity and mobile phone usage, introduces significant ethical concerns, particularly regarding the potential for discrimination and privacy violations. These data sources contain sensitive information about individuals' lives, habits, and preferences, which necessitates the development of norms or regulations to ensure their fair and responsible use in credit scoring. Ethical implications must be thoroughly examined through ongoing research to safeguard against misuse.

While both traditional and behavioral data are essential for credit scoring, the literature indicates a gap in the development of comprehensive models that seamlessly integrate these data types. Current research primarily focuses on the effectiveness of traditional and behavioral data individually, neglecting the potential advantages of a hybrid approach. This limitation underscores the need for models that combine these data sources effectively, harnessing the strengths of each to create more accurate and reliable credit scoring systems.

Incorporating payment behavior as a key component of behavioral data in credit scoring models is particularly crucial. Payment data offers a direct reflection of an individual's financial habits, providing insights that are both accessible and indicative of their financial responsibility. Its inclusion can enhance the predictive power of credit scoring models, especially for populations with limited access to traditional financial services. By integrating payment behavior with traditional credit data, a more holistic and accurate assessment of creditworthiness can be achieved, addressing the current gap in hybrid model development. This integration is necessary to create credit scoring systems that are not only more reliable but also more inclusive, ensuring that all relevant aspects of an individual's financial behavior are considered.

CHAPTER THREE: METHODOLOGY

3.0 INTRODUCTION

This chapter describes the approach used in creating a deep learning-based hybrid model for credit score prediction, building on the shortcomings of traditional credit scoring techniques that have been found and the increasing significance of including behavioural data. As was said in the sections above, it can be difficult to determine someone's creditworthiness using typical credit scoring methods, especially if they have a short credit history or engage in unusual financial practices. This study uses a Design Science Research (DSR) design, which stresses developing and accessing new artifacts to solve practical issues in response to these concerns. Using this strategy, we want to create a solid hybrid model that combines behavioural data with the advantages of traditional credit data.

By using this design, we want to get beyond the drawbacks of traditional credit scoring methodologies and give financial institutions access to a more precise and comprehensive tool for determining creditworthiness. The following sections of this chapter have detailed each step of the approach, detailing the precise methods and procedures used to accomplish our study goals.

3.1 DESIGN SCIENCE RESEARCH (DSR)

According to Pello (2018), DSR is a comprehensive research paradigm with the primary goal of creating prescriptive knowledge concerning the design of various artifacts, including software, methods, models, and concepts. This knowledge is instrumental in guiding research and practical applications for systematically and scientifically designing artifacts in future projects. The process of design and application, in turn, contributes to the accumulation of design-oriented knowledge within the DSR knowledge. (Pello, 2018) Because Design Science Research (DSR) focuses on developing and accessing novel solutions to challenging issues, it is especially well suited for hybrid credit scoring models. DSR strongly emphasizes the creation of artifacts, like hybrid models, to address particular problems, in this case, raising the accuracy of credit scoring. Using DSR, scientists can methodically create a hybrid model that combines several approaches, like merging RNNs and DNNs, to take advantage of their complementary advantages. Additionally, DSR includes iterative testing and refinement, which enables the model to be continuously improved depending on empirical input. This strategy guarantees that the hybrid credit scoring model is both practically and theoretically sound, offering a strong remedy to the shortcomings of the current credit scoring system.

DSR is suitable for this study because of its Iterative Improvement. DSR adopts an iterative design process, allowing researchers to improve credit scoring models in response to user feedback and performance assessments. This iterative process might result in ongoing improvements and flexibility to adjust shifting credit environments (Carstensen&Bernhard, 2019).

According to Brooke and Maedche (2019), DSR seeks to produce information that may be used to create Information Systems (IS) artifacts such as software, methodologies, models, and concepts. DSR seeks to produce artifacts that directly address issues in the real world. This can result in better credit evaluation procedures, lower risk for financial institutions, and easier access to credit for people and businesses, which makes it better for this study. Research methodologies from both the quantitative and qualitative domains can be included in DSR. Quantitative methods can handle the statistical modeling and prediction components, although qualitative methods can give a clearer understanding of the contextual and social factors that affect creditworthiness (Offermann et al., 2019).

According to Pello (2018), the DSR process typically consists of six stages or phases; Identification of the problem; Definition of objectives for a solution; Design and development of artifacts; Demonstration by using the artifact to solve the problem; Evaluation of the solution and Communication of the problem, the artifact, its utility, and its message.

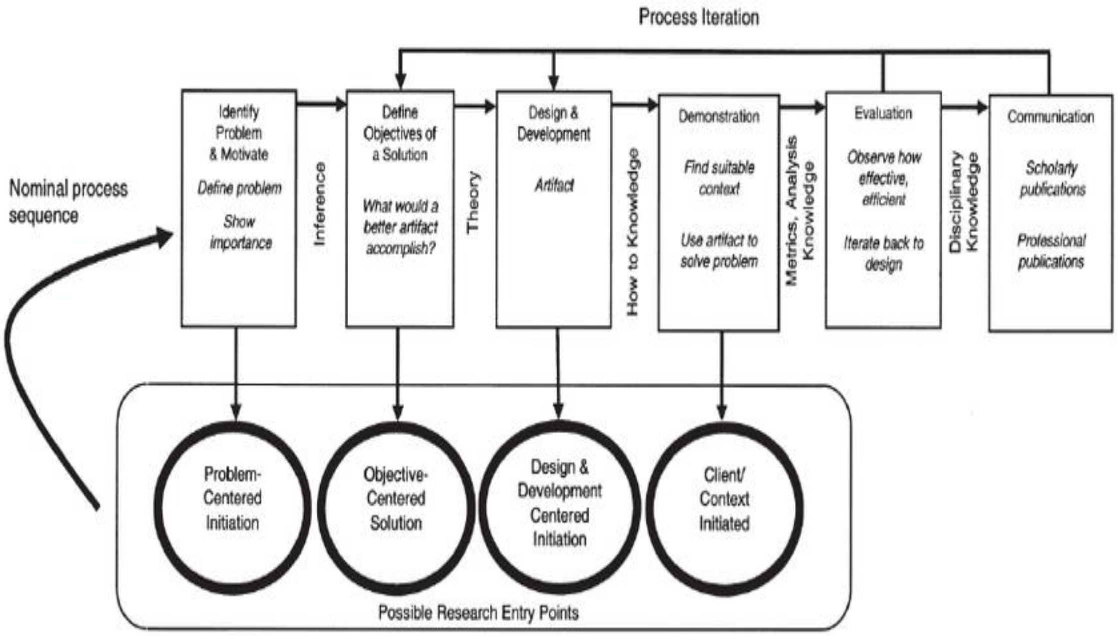

Process Iteration
Nominal process sequence
Identify Problem & Motivate
Define problem
Show importance
Inference
Define Objectives of a Solution
What would a better artifact accomplish?
Theory
Design & Development
Artifact
How to Knowledge
Demonstration
Find suitable context
Use artifact to solve problem
Metrics, Analysis Knowledge
Evaluation
Observe how effective, efficient
Iterate back to design
Disciplinary Knowledge
Communication
Scholarly publications
Professional publications
Problem-Centered Initiation
Objective-Centered Solution
Design & Development Centered Initiation
Client/ Context Initiated
Possible Research Entry Points

Figure 3. 1: Design Science Research Methodology framework.

3.2 Design Science Research Phases

The DSR process includes six phases. The study begins with problem identification and motivation, followed by the definition of the objectives for a solution, then design and development, demonstration, evaluation, and lastly, communication (Pello, 2018).

3.2.1 Identifying problem and motivation

The Design Science Research (DSR) design played a pivotal role in systematically identifying and defining the research problem, offering a structured approach that ensured both theoretical significance and practical relevance (Brocke et al.,2020). Providing an organized technique that guaranteed conceptual weight and practical relevance, the Design Science study (DSR) design was essential in thoroughly identifying and characterizing the study problem. The underutilization of deep learning models and the integration of behavioral data, in particular, were highlighted as significant gaps in the current credit scoring models by DSR, which helped identify them through an extensive and systematic literature review. After going through a rigorous process, it was discovered that the existing models rarely incorporated behavioral data (Ampountolas et al., 2021).

By filling in these gaps, the research problem for the misclassification of borrowers and the inability of traditional credit scoring models to reliably predict creditworthiness was clearly stated. This was brought into line with the real-world requirements of financial institutions looking for more comprehensive and reliable credit assessment instruments. This connection highlights the research's relevance by guaranteeing that the highlighted issue is both practically and conceptually grounded, directly addressing the difficulties financial institutions experience in the credit scoring area.

The Design Science Research (DSR) design emphasizes the critical alignment of research objectives with the identified problem, ensuring that the research is both focused and relevant. In this study, DSR guided the formulation of specific objectives that were directly aimed at addressing the shortcomings of existing credit scoring models. These objectives included evaluating the role of behavioral and traditional data in current models, developing a hybrid deep learning model, validating the model, and creating a web-based tool for visualization. By aligning these objectives with the identified problem, the research maintained a coherent structure where each objective was purposefully designed to contribute to solving the inadequacies of current models. The development of a hybrid model and the integration of behavioral data were thus directly motivated by the limitations of traditional models, ensuring that the research outcomes were both practically relevant and theoretically grounded.

The Design Science Research (DSR) design, with its focus on practical relevance and justification, played a crucial role in articulating the motivation behind this research. DSR ensured that the identified problem was not only theoretically significant but also had clear practical implications for enhancing credit scoring models in real-world applications. The motivation to develop a hybrid model was strongly justified by its potential to reduce biases, improve accuracy, and provide a more comprehensive assessment of credit risk. As a result, the motivation for the research was clearly articulated, highlighting the practical benefits of the proposed hybrid model and the critical importance of integrating behavioral data to enhance the assessment of creditworthiness.

The Design Science Research (DSR) provided a clear roadmap for developing the research artifact, which included the hybrid deep learning model and the accompanying web-based visualization tool. By adhering to DSR principles, the research ensured that problem identification and motivation were directly connected to the design and development of these artifacts, resulting in a coherent and focused research process. This structured and methodical approach established clear links between the identified problem, the motivation for the research, and the creation of the hybrid model and visualization tool, ensuring that each stage of the research was aligned and purpose-driven.

The Design Science Research played a crucial role in the problem identification and motivation step by providing a structured approach to defining the problem, aligning research objectives, and justifying the motivation for the research. The DSR design ensured that the research was grounded in practical relevance, with a clear focus on developing a solution that addresses the identified gaps in current credit scoring models. This foundation set the stage for the subsequent development and validation of a hybrid deep learning model that integrates behavioral and traditional data for improved credit score prediction.

3.2.2 Define the objectives of a solution

The research aims to develop a novel hybrid model that combines traditional credit data with behavioural data, thereby transforming the credit score prediction process. This study aims to clarify the functions and relationships between traditional and behavioural data in improving credit scoring algorithms, which is done by developing an advanced deep learning model that seamlessly combines traditional and behavioural data, transforming the precision and predictive ability of credit score forecasts. Model Validation is performed to thoroughly validate and evaluate the established model's performance to ensure it satisfies the exacting criteria of accuracy and dependability needed to make wise lending decisions.

Essentially, the goal of this research is to construct a cutting-edge hybrid model based on deep learning that would enable more transparent and accurate credit score forecasts by utilizing the combined strength of traditional and behavioural data.

3.2.3 Design and Development

In the Development stage of Design Science Research(DSR), data analysis and data preprocessing involves cleaning, transforming, and preparing data for analysis and model development. This is a crucial step where artifacts such as models, frameworks, or systems are created or designed. Following preprocessing, data analysis is conducted to refine the artifact's design, validate underlying assumptions, and model patterns and relationships, ensuring the development process is informed and guided by empirical insights.

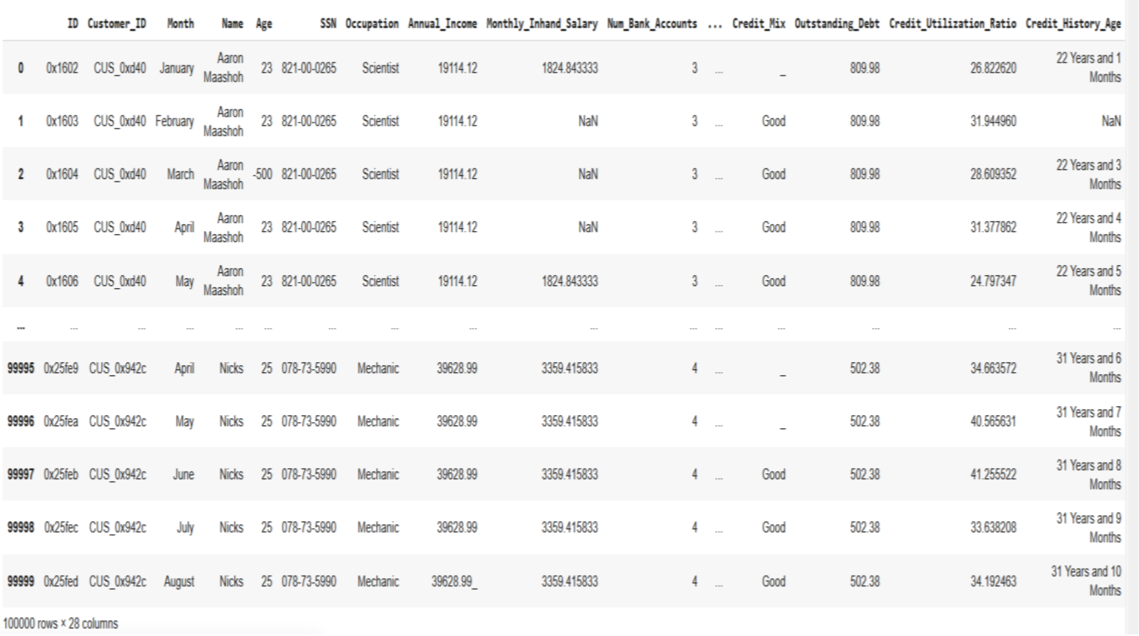
3.2.3.1 Data Analysis

The main goal of this section is to analyze the data used to create a hybrid model of deep learning for improved credit score prediction. This analysis is important because it established the foundation for comprehending the properties of the data and the connections among different elements, which are essential for developing a precise and trustworthy prediction model. This section examines both traditional and behavioural data to categorize the features that were used to inform the model. The key objective of data analysis was to identify features that would contribute significantly to the credit score prediction model and to make sure the features are error-free so that they can be read by the model.

3.2.3.1.1 Data description

In this study the researcher uses secondary data. The dataset consists of 100000 rows and 28 columns and in CSV format. The data come from https://www.kaggle.com/datasets/parisrohan/credit-score-classification.The columns represent the features (variables) present in the dataset while the rows contain 100000 entries for each feature. The sample dataset consists both traditional and behavior data.

The data is made up of numerous factors that contribute to the credit score. A glimpse of the dataset is shown below.



	ID	Customer_ID	Month	Name	Age	SSN	Occupation	Annual_Income	Monthly_Inhand_Salary	Num_Bank_Accounts	...	Credit_Mix	Outstanding_Debt	Credit_Utilization_Ratio	Credit_History_Age
0	0x1602	CUS_0xd40	January	Aaron Maashoh	23	821-00-0265	Scientist	19114.12	1824.843333	3	...	_	809.98	26.822620	22 Years and 1 Months
1	0x1603	CUS_0xd40	February	Aaron Maashoh	23	821-00-0265	Scientist	19114.12	NaN	3	...	Good	809.98	31.944960	NaN
2	0x1604	CUS_0xd40	March	Aaron Maashoh	-500	821-00-0265	Scientist	19114.12	NaN	3	...	Good	809.98	28.609352	22 Years and 3 Months
3	0x1605	CUS_0xd40	April	Aaron Maashoh	23	821-00-0265	Scientist	19114.12	NaN	3	...	Good	809.98	31.377862	22 Years and 4 Months
4	0x1606	CUS_0xd40	May	Aaron Maashoh	23	821-00-0265	Scientist	19114.12	1824.843333	3	...	Good	809.98	24.797347	22 Years and 5 Months
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
99995	0x25fe9	CUS_0x942c	April	Nicks	25	078-73-5990	Mechanic	39628.99	3359.415833	4	...	_	502.38	34.663572	31 Years and 6 Months
99996	0x25fea	CUS_0x942c	May	Nicks	25	078-73-5990	Mechanic	39628.99	3359.415833	4	...	_	502.38	40.565631	31 Years and 7 Months
99997	0x25feb	CUS_0x942c	June	Nicks	25	078-73-5990	Mechanic	39628.99	3359.415833	4	...	Good	502.38	41.255522	31 Years and 8 Months
99998	0x25fec	CUS_0x942c	July	Nicks	25	078-73-5990	Mechanic	39628.99	3359.415833	4	...	Good	502.38	33.638208	31 Years and 9 Months
99999	0x25fed	CUS_0x942c	August	Nicks	25	078-73-5990	Mechanic	39628.99_	3359.415833	4	...	Good	502.38	34.192463	31 Years and 10 Months

100000 rows × 28 columns

Figure 3. 2: Sample Dataset Snippet

The variables are both categorical and numerical data. Categorical data is made up of non-numerical values that indicate labels or categories, and numerical values that represent quantities or measurements. Table 3.1 represents both categorical and numerical data with their data type details.

Table 3. 1 Categorical and numerical data with data type

	Features/variables	**Classification**	**Data type**
1	Annual_Income	Numerical data	Float64
2	Num_Bank_Accounts	Numerical data	Int64
3	Interest_Rate	Numerical data	Int64
4	Delay_from_due_date	Numerical data	Int64
5	Num_Credit_Inquiries	Numerical data	Float64
6	Outstanding_Debt	Numerical data	Float64
7	Payment_of_Min_Amount	Numerical data	Int64
8	Monthly_Inhand_Salary	Numerical data	float64
9	Num_Credit_Card	Numerical data	int64
10	Num_of_Loan	Numerical data	float64
11	Changed_Credit_Limit	Numerical data	float64
12	Credit_Mix	Numerical data	int64
13	Credit_Utilization_Ratio	Numerical data	float64
14	Total_EMI_per_month	Numerical data	float64
15	Credit_Score	Numerical data	int64
16	Age	Numerical data	float64
17	ID	Categorical data	object
18	Customer_ID	Categorical data	object
19	Name	Categorical data	object
20	Occupation	Categorical data	object
21	Num_of_Delayed_Payment	Categorical data	object
22	Amount_invested_monthly	Categorical data	object
23	Month	Categorical data	object

24	SSN	Categorical data	object
25	Type_of_Loan	Categorical data	object
26	Credit_History_Age	Categorical data	object
27	Monthly_Balance	Categorical data	Object
28	Payment_Behaviour	Categorical data	object

The researcher converted the dataset to float using a type conversion. Because of irregularities in the data file, numeric data could be read as strings. Therefore, this conversion was required to ensure data type consistency. Converting to float facilitates the handling of missing or inconsistent data and enables the seamless execution of numerical operations, such as statistical analysis, computations, and visualizations. Compatibility with machine learning algorithms that need numerical input is enabled by this critical stage, which makes efficient data handling and analysis possible.

3.2.3.1.2 Data Cleaning

Data cleaning is an important process in this research, where errors, inconsistencies, and inaccuracies from raw data are eliminated and corrected. In this study, the researcher found that the dataset had Missing values, Outliers, negative entries and empty spaces. The researcher standardized the data to avoid inconsistencies and encoded categorical variables so that the model could read the data. The researcher adopted the following methods to clean the data set before feature embarking on engineering: eliminating missing values, handling Outliers, eliminating inconsistency of data, standardizing data and encoding categorical features.

i. Missing values

After identifying the missing values, the researcher managed them at the preprocessing stage of the data to reduce bias and preserve the dataset's integrity for modelling and analysis. In this research, two methods were used to deal with missing values, one for numerical columns. The misplaced values were replaced with the mean, 'filing' method, where these features were affected Monthly_Inhand_Salary, Num_Credit_Inquiries, Changed_Credit_Limit, and Interest_Rate. The second method eliminates null values by employing column-wise elimination, where the researcher deleted any columns with a significant percentage of missing data. The following features (6) were eliminated on the basis of high missing values, Num_of_Delayed_Payment, Num_Credit_Card, Amount_invested_monthly, Num_Bank_Accounts, Monthly_Balance and Credit_History_Age.

ii. Handling Outliers

A feature skewness analysis was used to find outliers. The degree of skewness in data signifies whether it is positively skewed (right-skewed), negatively skewed (left-skewed), or symmetrically distributed. Understanding the characteristics of skewness allowed the researcher to spot high- or low-value outliers in the data. High-value outliers are indicated by positive skewness (right-skewed data), whereas low-value outliers are indicated by negative skewness (left-skewed data). A bar chart illustrating the skewness of various financial metrics is presented in Figure 4.2. The irregularity of a real-valued arbitrary variable's probability distribution with respect to its mean is measured by its skewness.

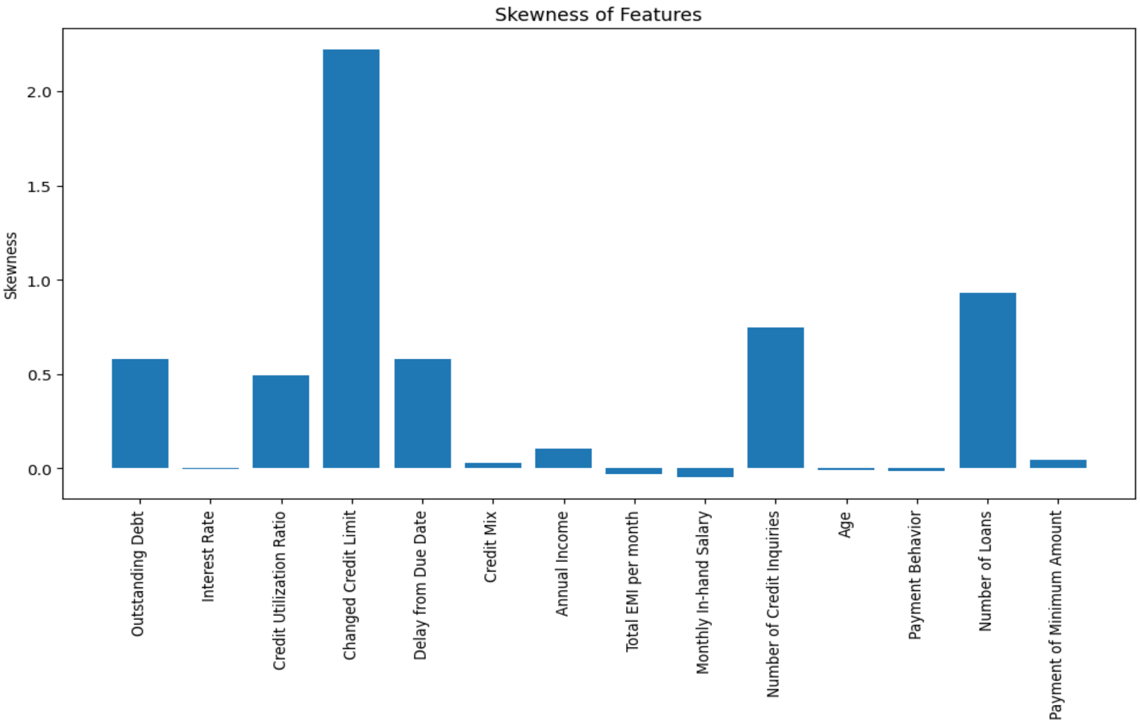


Figure 3. 3: Skewness of data

The Y-axis (Skewness) quantifies the skewness value of each financial feature, while the X-axis (Features) lists many datasets financial features, including Outstanding Debt, Interest Rate, Credit Utilization Ratio, etc. When the distribution is positively skewed, the right tail is longer or fatter than the left, and when it is negatively skewed, the left tail is longer or fatter. The highest skewness feature is Charged Credit Limit; it is significantly higher than any other feature and indicates many outliers. Other notable skewness features are the Number of Credit Inquiries, which is also notable but much lower than the Charged Credit Limit, and the Number of Loans, which is moderately skewed. The credit limit that has been charged has the largest skewness.

After identifying the features with outliers, the researcher used the Interquartile Range (IQR) method to deal with them, as outlined below.

Calculating the Quartiles:

First Quartile (Q1): The value below which 25% of the data fall.

Third Quartile (Q3): The value below which 75% of the data fall.

Interquartile Range (IQR): $IQR = Q3 - Q1$

Defining Thresholds:

Lower Bound: $Q1 - 1.5 \times IQR$

Upper Bound: $Q3 + 1.5 \times IQR$

iii. Ensuring Consistency of data.

Ensure consistency in data formatting for certain variables so that they can eliminate errors. This was done by eliminating data entry errors. The string "_" is removed from the entries. In the Num_of_Loan column, the negative entries (-100) were considered erroneous and replaced with NaN, which was dealt with as a missing value.

iv. Standardizing Data

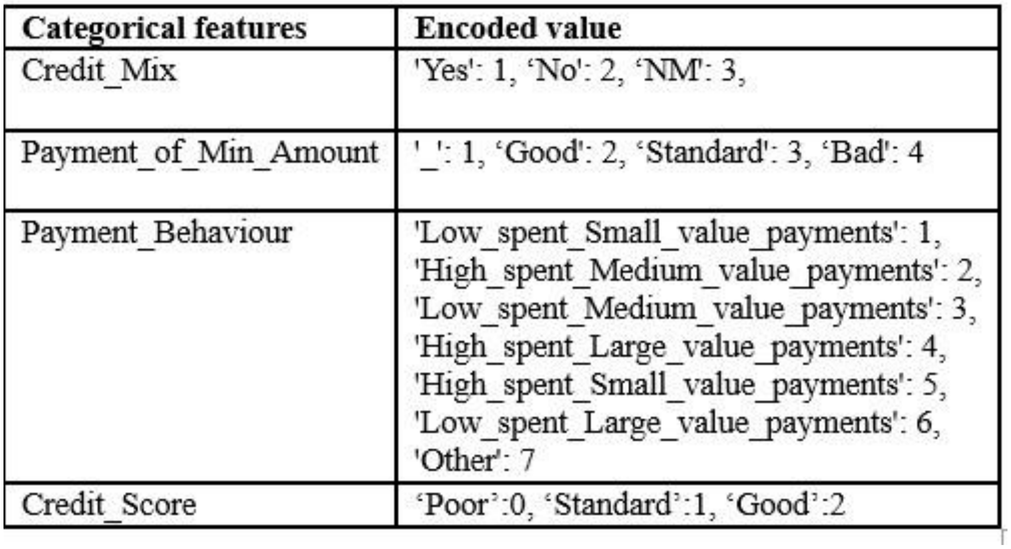
In data analysis and machine learning, standardizing data also referred to as feature scaling or normalization, is a preprocessing procedure. It entails varying a dataset's properties so that its mean is 0 and its standard deviation is 1. Data standardization guarantees that various attributes have a comparable scale, which can be essential for some algorithms to function well. In this research, standardization was done using a standard scaler, as shown below.

```
scaler = StandardScaler()
data_scaled = scaler.fit_transform(data)
data = pd.DataFrame(data_scaled, columns=data.columns)
```

v. Encoding Categorical Variables

Given that the majority of machine learning methods require numerical input data, encoding categorical variables is an essential preprocessing step. Qualitative data is represented by categorical variables, which have values that can come from a small number of pre-established categories. These categorical variables are encoded so that they can be processed by algorithms in an efficient numerical manner. The technique used in this research to encode is One hot encoding and label encoding where it Creates a binary matrix from categorical variables in Table 4.5, with each category represented as a binary column and transforms categorical data into ordinal number values by allocating a distinct integer to each category respectively.

Table 3. 2 Encoded value

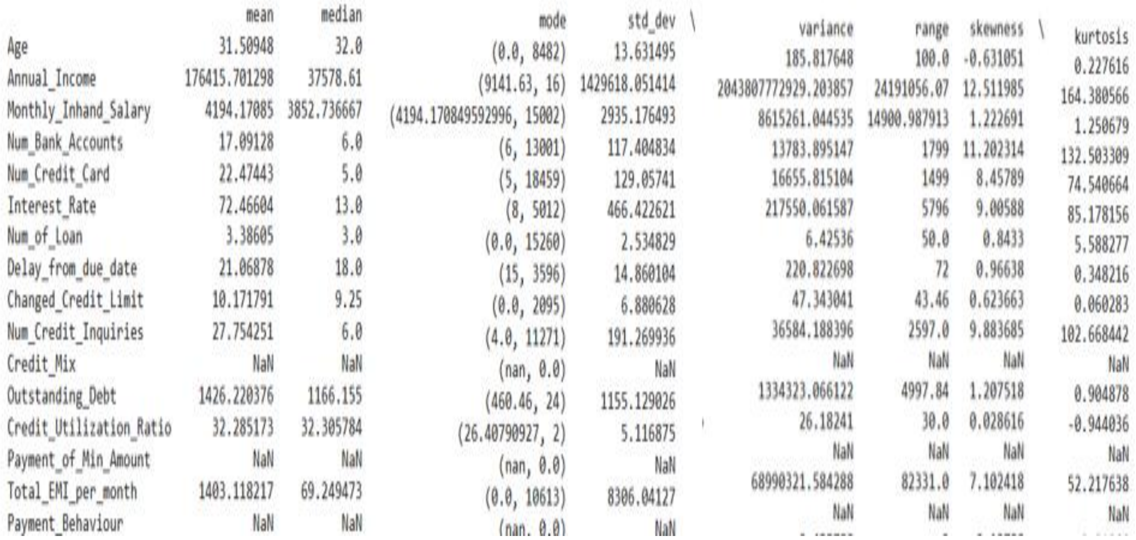


Categorical features	Encoded value
Credit_Mix	'Yes': 1, ‘No': 2, ‘NM': 3,
Payment_of_Min_Amount	'_': 1, ‘Good': 2, ‘Standard': 3, ‘Bad': 4
Payment_Behaviour	'Low_spent_Small_value_payments': 1, 'High_spent_Medium_value_payments': 2, 'Low_spent_Medium_value_payments': 3, 'High_spent_Large_value_payments': 4, 'High_spent_Small_value_payments': 5, 'Low_spent_Large_value_payments': 6, 'Other': 7
Credit_Score	‘Poor’:0, ‘Standard’:1, ‘Good’:2

3.2.3.1.3 Feature Selection

The next step after data cleaning was feature selection, which involved identifying the most relevant data features. Descriptive statistics (Figure 4.5) were conducted to identify feature importance. If a feature shows the highest correlation or variation with the

Credit_Score, which is the target variable, it means it significantly contributes to credit scoring; hence, these features were retained. Features that contribute very little to predicting credit scores were dropped using the Dropna function. The following thirteen (13) features were dropped; Id, Customer_ID, Type_of_Loan, Month, Name, Occupation, Ssn, Num_of_Delayed_Payment, Num_Credit_Card, Amount_invested_monthly, Num_Bank_Accounts, Monthly_Balance and Credit_History_Age. As a result, these features were dropped. This procedure is performed after descriptive statistics have been thoroughly examined, as shown in Figure 3.4.



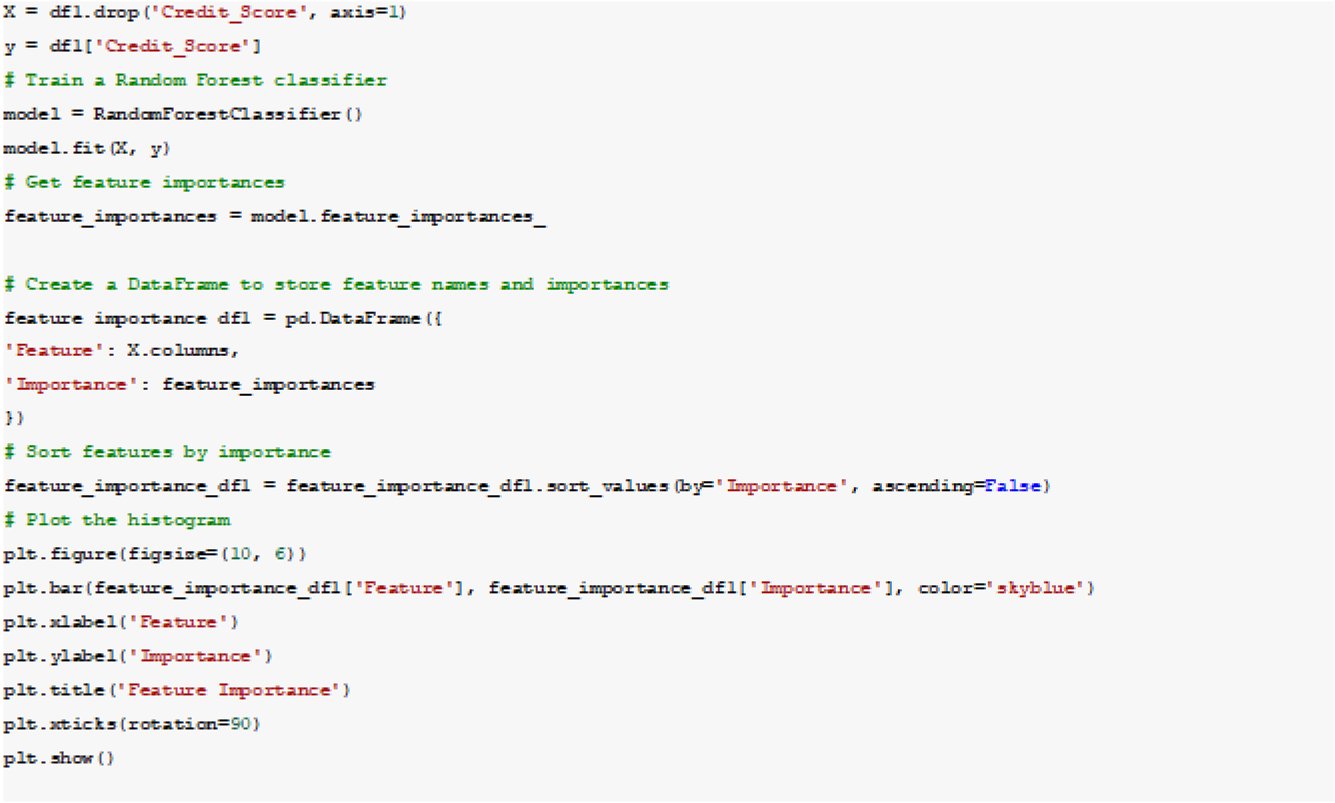
```
                                 mean       median                           mode         std_dev  \            variance         range   skewness  \   kurtosis
Age                          31.50948         32.0                     (0.0, 8482)       13.631495              185.817648         100.0  -0.631051      0.227616
Annual_Income            176415.701298     37578.61                   (9141.63, 16)  1429618.051414  2043807772929.203857   24191056.07  12.511985    164.380566
Monthly_Inhand_Salary       4194.17085  3852.736667  (4194.170849592996, 15002)     2935.176493        8615261.044535  14900.987913   1.222691      1.250679
Num_Bank_Accounts            17.09128          6.0                     (6, 13001)      117.404834          13783.895147          1799  11.202314    132.503309
Num_Credit_Card              22.47443          5.0                     (5, 18459)       129.05741          16655.815104          1499    8.45789     74.540664
Interest_Rate                72.46604         13.0                      (8, 5012)      466.422621         217550.061587          5796    9.00588     85.178156
Num_of_Loan                   3.38605          3.0                    (0.0, 15260)        2.534829               6.42536          50.0     0.8433      5.588277
Delay_from_due_date          21.06878         18.0                      (15, 3596)       14.860104            220.822698            72    0.96638      0.348216
Changed_Credit_Limit        10.171791         9.25                     (0.0, 2095)        6.880628             47.343041         43.46   0.623663      0.060283
Num_Credit_Inquiries        27.754251          6.0                    (4.0, 11271)      191.269936          36584.188396        2597.0   9.883685    102.668442
Credit_Mix                        NaN          NaN                      (nan, 0.0)             NaN                   NaN           NaN        NaN           NaN
Outstanding_Debt          1426.220376     1166.155                    (460.46, 24)     1155.129026        1334323.066122       4997.84   1.207518      0.904878
Credit_Utilization_Ratio    32.285173    32.305784                (26.40790927, 2)        5.116875              26.18241          30.0   0.028616     -0.944036
Payment_of_Min_Amount             NaN          NaN                      (nan, 0.0)             NaN                   NaN           NaN        NaN           NaN
Total_EMI_per_month       1403.118217    69.249473                    (0.0, 10613)      8306.04127        68990321.584288       82331.0   7.102418     52.217638
Payment_Behaviour                 NaN          NaN                      (nan, 0.0)             NaN                   NaN           NaN        NaN           NaN
```

Figure 3. 4: Descriptive statistics

When examining numerical features, the researcher looks for ones that have a strong connection or significant variance with the target variable. High variance indicated that the feature contains significant information hence, the following features fifteen (15) were retained: Age, income, Monthly_Inhand_Salary, Interest Rate, Num of Loans, Delay from due date, Changed Credit Limit, Num_Credit_Inquiries, credit, Outstanding Debt, Credit Utilization Ratio, Payment of Min Amount, Total EMI per month, Payment Behaviour and Credit_Score. Strong correlation points or direct relationships with the target variable show the feature is important.

i. Feature importance

Feature importance was informed by a feature significance score. Feature significance scores are metrics employed to assess each feature's relative importance to a target variable inside a dataset. These scores can help understand each feature's input to the forecast or classification challenge. In this research, feature significance scores were extracted by training a Random Forest classifier, as seen in Figure 4.7. The Random Forest model yields a fundamental characteristic significance measure after training. This significance is determined by calculating the regular decrease in impurity (e.g., Gini impurity) that each feature has for all trees in the forest. The algorithm adds up the reduction in impurity for every feature across all trees, weighted by the likelihood of reaching the relevant node. Higher information gain or impurity decrease features are given greater weight (Sharma, 2021). These scores are then used to prioritize the features that the algorithm has determined to be the most influential.



```
X = dfl.drop('Credit_Score', axis=1)
y = dfl['Credit_Score']
# Train a Random Forest classifier
model = RandomForestClassifier()
model.fit(X, y)
# Get feature importances
feature_importances = model.feature_importances_

# Create a DataFrame to store feature names and importances
feature importance dfl = pd.DataFrame({
'Feature': X.columns,
'Importance': feature_importances
})
# Sort features by importance
feature_importance_dfl = feature_importance_dfl.sort_values(by='Importance', ascending=False)
# Plot the histogram
plt.figure(figsize=(10, 6))
plt.bar(feature_importance_dfl['Feature'], feature_importance_dfl['Importance'], color='skyblue')
plt.xlabel('Feature')
plt.ylabel('Importance')
plt.title('Feature Importance')
plt.xticks(rotation=90)
plt.show()
```

Figure 3. 5: Random Forest classifier Code Snippet

The strong feature importance mechanism offered by the Random Forest classifiers code in Figure 4.7 is essential for feature selection, interpretability of the model, and obtaining insights from the data. The researcher created more effective and understandable models by using Random Forests to emphasize the most important feature through the use of Mean Decrease Impurity and Permutation Importance techniques. The feature importances were extracted from the model after training. The researcher used Seaborn and Matplotlib to show the importance of the feature in a bar plot (Figure 3.6). This procedure guarantees that the most pertinent characteristics are employed, which may improve model performance and reduce complexity in addition to enhancing comprehension of the data.

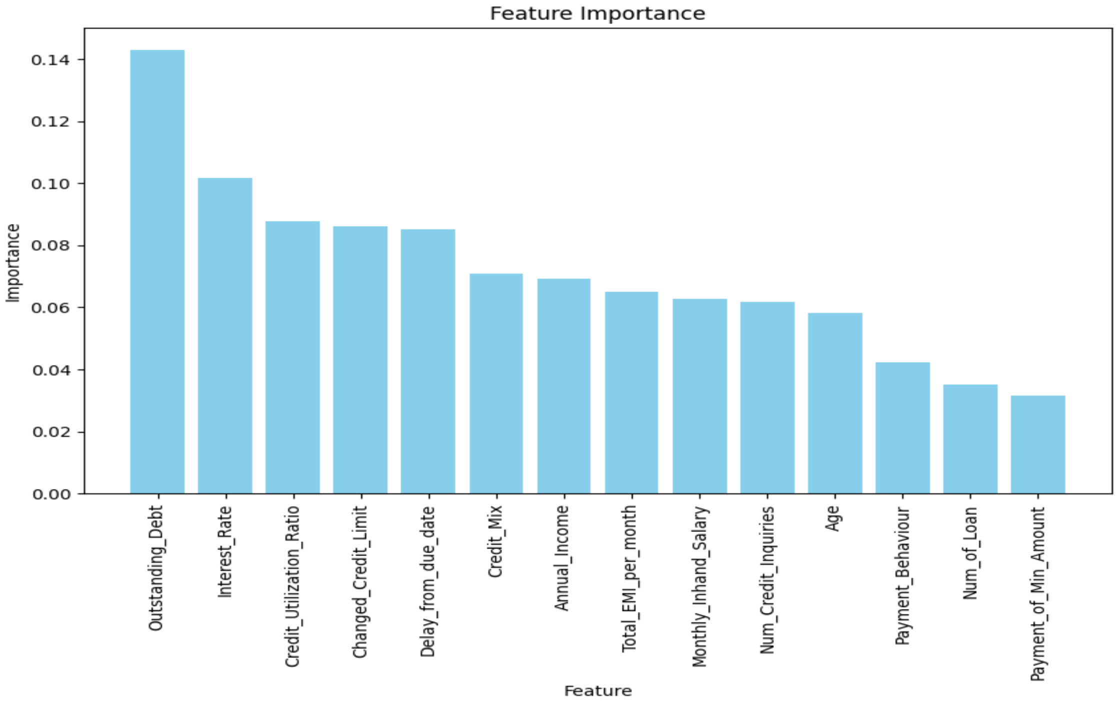
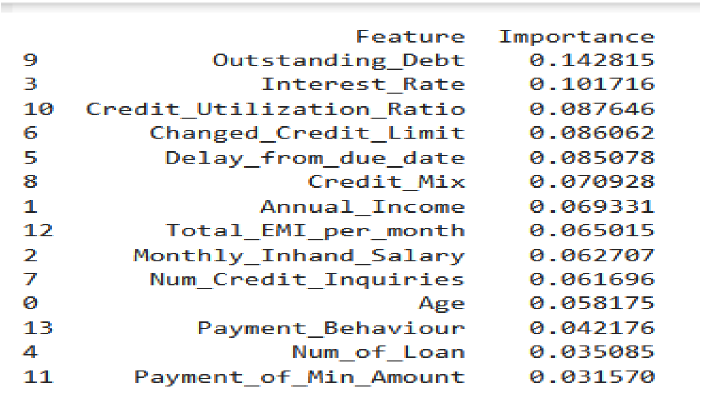


Figure 3. 6: Feature importance



```
                     Feature  Importance
9           Outstanding_Debt    0.142815
3              Interest_Rate    0.101716
10  Credit_Utilization_Ratio    0.087646
6       Changed_Credit_Limit    0.086062
5        Delay_from_due_date    0.085078
8                 Credit_Mix    0.070928
1              Annual_Income    0.069331
12       Total_EMI_per_month    0.065015
2      Monthly_Inhand_Salary    0.062707
7       Num_Credit_Inquiries    0.061696
0                        Age    0.058175
13         Payment_Behaviour    0.042176
4                Num_of_Loan    0.035085
11     Payment_of_Min_Amount    0.031570
```

Figure 3. 7: Feature Importance Scores

From the results in Figure 3.7, Outstanding debt with a score of 0.142815 has the highest influence on the forecast of credit score, and the order of importance decreases to payment_of _ Min _Amount with a score of 0.031570. This clearly shows the contribution of traditional data as well as behavioural data in credit score prediction.

ii. L1 Regularization- Least Absolute Shrinkage and Selection Operator

To determine feature importance in this research, regularization was used to improve the model's generalization performance by preventing overfitting. When a model learns the training set too well, it becomes overfitted and captures noise or random oscillations that are unique to the training set but do not translate well to new data. To reduce this problem, L1 Regularization (LASSO) was used.

L1 Regularization (LASSO) Consists of introducing a penalty term that is inversely related to the number of coefficients. Forcing certain coefficients to zero typically yields sparse solutions, which essentially carry out feature selection. When it comes to feature selection, embedded techniques such as LASSO. They are helpful since they penalize the features according to their coefficients, which promotes sparsity and helps identify the most significant features. The LASSO path plot in Figure 4.9 shows how each feature's coefficients change as the regularization parameter Alpha (α) varies. This makes the lasso crucial to Enhanced Predictive Precision. Through regularization, LASSO can lessen overfitting and enhance the model's predictive ability on hypothetical data. We can determine which features are most significant by looking at which features have non-zero coefficients despite having high Alpha (α).

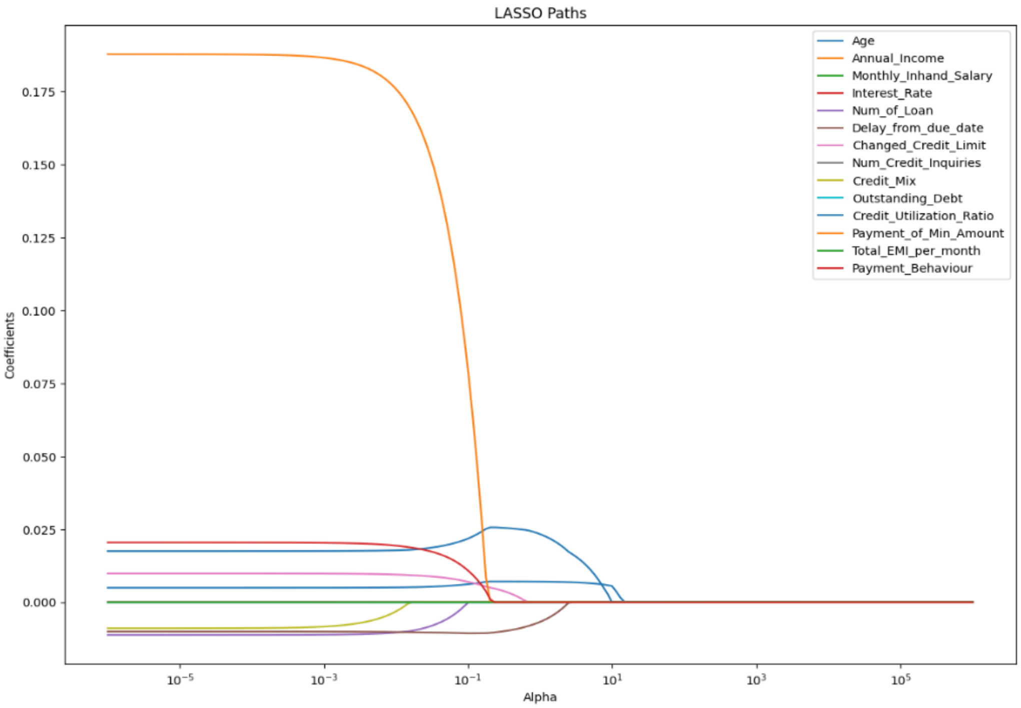


Figure 3. 8: LASSO paths graph

Understanding the impact of regularization on each feature is aided by the LASSO route plot. We can determine which features are most significant in the model by looking at how the coefficients of the features vary with different alpha values. In this instance, parameters like Age and Delay_from_due_date hold significance as the regularization strength increases, while Annual_ Income is especially important at lower regularization levels. This graphic is helpful for feature selection because it shows which characteristics are most important to the model's predictions at various regularization settings.

iii. Data Visualization

The researcher used visualization to graphically represent the data and feature importance, emphasizing its vital function in data analysis to support analysts and decision-makers in visualizing, comprehending, and disseminating data-driven insights. Heat maps, which offer a visual depiction of feature correlations and facilitate the identification of strongly linked features that may be less interpretable, redundant, or prone to overfitting, were employed as a feature selection technique. This method helped to guide the creation of new features or data transformations required to extract significant patterns, as well as the exploration of possible feature relationships. Ultimately, the investigator underlined that heatmaps were crucial instruments for exploratory data examination, providing insightful information on what attributes from the models.

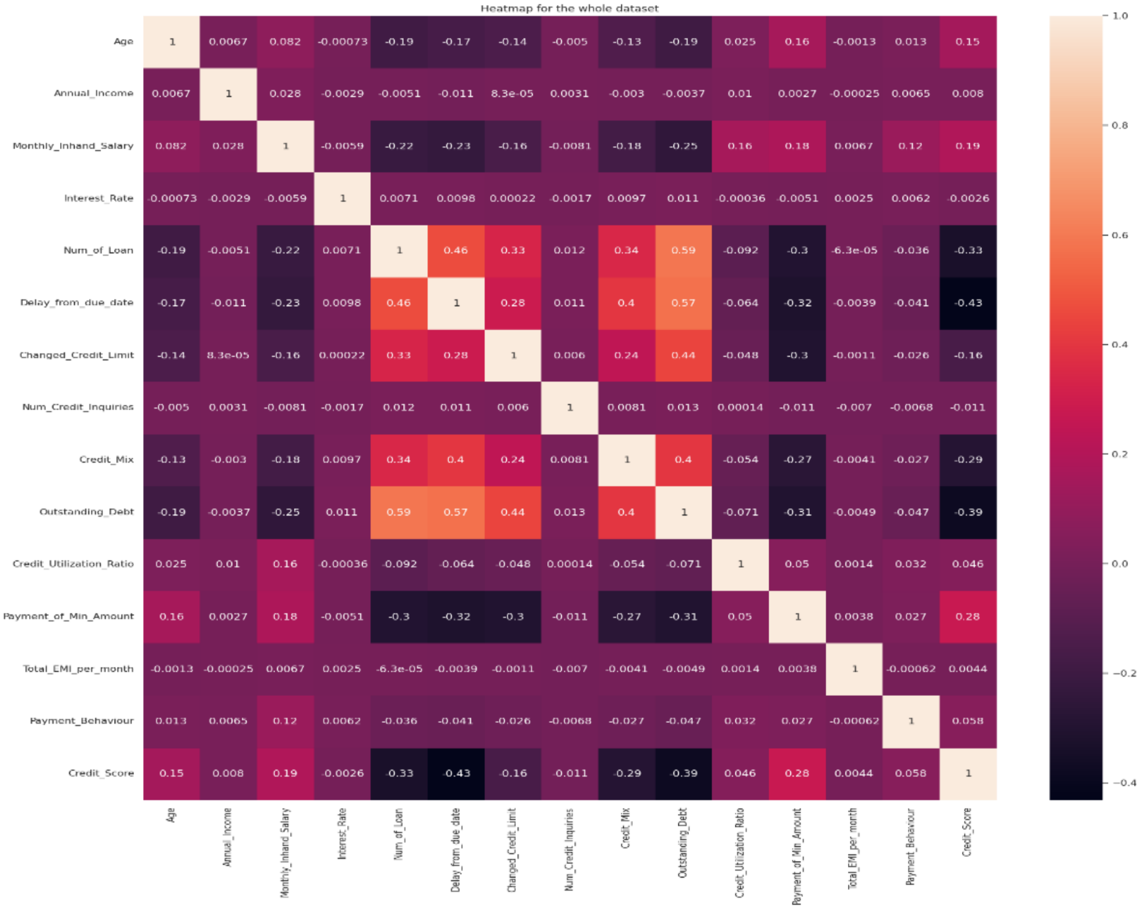


Figure 3. 9: Heat map for the whole dataset after encoding

The features with the biggest negative connections with credit scores are Outstanding Debt, Delay from the due date, and Number of Loans; these features' higher values are likely linked to lower credit scores.

Significant correlations between the features themselves also exist, suggesting multicollinearity, the state in which certain features are very closely related to one another. These correlations aid in the comprehension of the connections between the features. From the heat map, we can feature an importance graph, and the role of traditional and behavioural data is clearly outlined.

3.2.3.1.4 Data splitting

According to Abraham et al,(2021) Splitting data into a 70:30 ratio is a common and well-justified practice in machine learning, as it balances the need for model training and evaluation. Allocating 70% of the data to the training set provides the model with a substantial amount of information to learn underlying patterns and relationships, which helps it, generalize effectively to new, unseen data. The remaining 30% is reserved for testing, allowing for a thorough evaluation of the model's performance (Abraham et al,.2021). Alternatively, an 80:20 split is also frequently used, where 80% of the data is designated for training and 20% for testing. In this research, the training data was further subdivided into 80% for training and 20% for validation (Abraham et al,.2021). This approach results in an overall data split of 56% for training, 14% for validation, and 30% for testing. This stratification helps ensure that the model is not only well-trained but also rigorously validated and tested, enhancing its reliability and performance evaluation. The dataset was split using the train_test_split function from Python libraries. The data was split into three parts: training, validation, and testing, and it was set in a ratio of 56:14:30 of the total data. There are 56,000 for training,14,000 for validation, and 30,000 for testing data.

3.2.3.1 Model Development Techniques and Tools

The development of hybrid deep learning models, including Deep Neural Networks (DNNs) and Recurrent Neural Networks (RNNs) with Long Short-Term Memory (LSTM) units, was conducted using robust Python libraries such as NumPy, Pandas, Scikit-learn, TensorFlow/Keras, and Matplotlib, chosen for their capabilities in handling complex data processing, feature engineering, and model optimization. Emphasizing rigorous data preparation, feature engineering and preprocessing transformed raw data into a format that enhances learning efficiency by addressing missing values, scaling features, and encoding categorical variables. Advanced optimization and regularization techniques were applied to improve model precision and resilience, with the RNN, having 3,061 trainable parameters, and the DNN, with 170 parameters, being optimized using dropout and the AdamW optimizer. Dropout helped prevent overfitting by randomly deactivating neurons, while AdamW, an advanced variant of the Adam optimizer, incorporated weight decay directly into the optimization process. This approach, which decouples weight decay from the learning rate, improved regularization, and model stability by discouraging excessively large weights, ultimately enhancing the models' overall performance and reliability.

3.2.4 Demonstration

In Design Science Research (DSR), the Demonstration phase is crucial for validating the designed artifact by showcasing its functionality and utility in addressing the identified problem. This phase involves implementing the artifact such as a hybrid deep learning model for credit scoring by design specifications, rigorously testing it to ensure it performs as expected, and evaluating it against criteria like accuracy and robustness. Feedback from users or stakeholders is collected to gauge how well the artifact meets their needs and to identify areas for improvement. For the hybrid deep learning model developed for credit scoring, this phase involves validating the model in a real or simulated financial environment, assessing its performance against actual credit outcomes, and verifying its integration of behavioral and traditional data. This process not only demonstrates the model's capability and effectiveness but also provides empirical evidence of its value, thereby addressing Objective 3 by validating the hybrid model and ensuring it delivers the expected benefits and improvements over existing methods.

3.2.5 Evaluation

In Design Science Research (DSR), the Evaluation phase is crucial for assessing the effectiveness and performance of the designed artifact. For this research, the evaluation of the hybrid deep learning model's predictive performance was conducted using a range of metrics. The researcher utilized a confusion matrix to assess classification accuracy by showing true positives, true negatives, false positives, and false negatives. Area Under the Curve (AUC) was employed to measure the model's ability to distinguish between classes, with higher values indicating better performance. Sensitivity and specificity provided insights into the model’s ability to correctly identify positive and negative instances, respectively. Additionally, Mean Squared Error (MSE) quantified the average squared difference between predicted and actual values, while Root Mean Squared Error (RMSE) offered an intuitive measure of prediction error in the same units as the response variable. By incorporating these diverse metrics, the Evaluation phase ensured a comprehensive analysis of the model’s performance, validating its effectiveness in predicting credit scores and addressing the identified research problem.

The choice to use RMSE (Root Mean Square Error) and MSE (Mean Square Error) for model evaluation is informed by their ability to quantify the accuracy of predictions by measuring the average magnitude of errors between the predicted and actual values. MSE calculates the average of the squared differences between predictions and actual values, which penalizes larger errors more than smaller ones, making it sensitive to outliers. RMSE, the square root of MSE, expresses this error in the same units as the target variable, making it easier to interpret. These metrics are particularly useful when the goal is to minimize large errors and when the scale of the errors matters, which is crucial in applications like credit scoring where prediction accuracy directly impacts decision-making.

3.2.6 Communication

In Design Science Research (DSR), the Communication phase involves sharing research findings and insights with relevant stakeholders, including academic and professional audiences. For the hybrid deep learning model developed for credit scoring, this phase encompasses several key activities. First, detailed documentation of results, including evaluation metrics such as confusion matrix, AUC, sensitivity, specificity, MSE, and RMSE, is prepared to ensure clarity and accessibility. The research is then disseminated through academic papers, conferences, and workshops, allowing for peer review and feedback. Engaging with financial institutions and industry professionals is also crucial, where the model's capabilities and practical applications are presented, often using a web-based visualization tool to illustrate how it integrates behavioral and traditional data. This tool helps stakeholders interact with the data and interpret the model's effectiveness. Finally, collecting feedback from both academic and industry audiences allows for further refinement, ensuring the model’s continued relevance and impact. Through these efforts, the research findings are effectively communicated, validated, and utilized by both researchers and practitioners.

The research effect is further enhanced by archiving, knowledge transfers for real-world applications, and iterative modifications. Its intended audience will more likely access research when it is communicated effectively. In this study, the researcher uses secondary data. The data come from https://www.kaggle.com/datasets /parisrohan/credit-score-classification. The dataset comprises 28 variables and 100000 instances for the training set and 27 variables and 50,000 instances for the test set.

The dataset is fit for this research since it contains both traditional data and behavioural data that are useful in the research. With the two types of data, it will be easier to do a bivariate analysis and able to determine whether there are any correlations, dependencies, or relationships between the two variables. This is useful for seeing possible trends or patterns in the data.

3.3 Web Application Tool

The web application tool has been created using the Streamlit framework. Streamlit is a lightweight, adaptable micro-framework for Python web development that is perfect for developers looking for quick development and customization because of its simplicity, ease of use, and minimalist features. With Streamlit, developers easily connect new components to meet the specific needs of small to medium-sized web applications or APIs. Streamlit provides basic tools and libraries. Its predilection for projects that value flexibility, simplicity, and low overhead highlights how well-suited it is for a range of development scenarios

The process of creating a web application tool using the Streamlit framework starts with a thorough examination of user requirements gleaned from datasets, which directs the design stage to give priority to architectural and UI/UX issues. Because of Streamlit simplicity and flexibility, as well as development, are done iteratively by dividing work into small, manageable modules and introducing them one at a time. The tool's back-end logic is implemented to guarantee dynamic response creation by utilizing Streamlit routing and templating features. Comprehensive testing is carried out to ensure functionality, dependability, and user pleasure. This includes unit, integration, and usability testing. For automation and scalability, deployment is handled using Docker and Kubernetes, and user feedback is used for validation and assessment to provide insights for future developments. This methodology places Streamlit at the center of building a robust and intuitive online application tool, optimizing the processes of user engagement, deployment, and development.

3.4 Ethical consideration

Developing the credit scoring model requires handling sensitive data and making crucial choices that have a big impact on people's financial well-being. Ethical considerations must be prioritized to ensure fairness, transparency, and the proper use of data throughout the process. First, the researcher will seek permission to conduct this research from

Kirinyaga University; further permission will also be sought from NACOSIT. This is necessary as the two institutions will only approve if they believe that this research aligns with their respective policies, standards, and ethical guidelines, ensuring the research's ethical and academic integrity. Fairness and bias mitigation are important ethical factors to consider while developing credit scoring model predictions using data science techniques. The researcher makes sure the model is built to be impartial and fair to all demographic categories (such as race, gender, and age). Use fairness measures and approaches to detect and lessen bias in the data and model predictions. Keep a close eye on the model's different effects and adjust it if bias shows up while it's being used. This will be achieved through clearly recording the credit scoring model's development, including the data sources used, the features used, and the model architecture. This will make the model transparent and easier to understand. Utilize comprehensible models or methods to shed light on the model's decision-making processes. This assists candidates in understanding the reasoning behind their particular credit score.

The researcher seeks all the required approvals and follows all required rules and legislation related to any secondary data that has been used in this study. The research implements strong security measures, data encryption, and access limits to protect sensitive client data. To secure customer information, the study abides by all applicable rules and regulations regarding data protection. The study ensures that the information used to create the credit scoring model is precise and current and avoids using inaccurate data, which may lead to unfair and biased decisions. The study validates and tests the model to evaluate the correctness and performance of the credit scoring model. The research makes sure it satisfies predefined criteria by using appropriate assessment metrics and validation approaches

3.5 SUMMARY

The chapter describes the creation of a hybrid credit score prediction model that addresses shortcomings in typical scoring techniques by combining behavioural indications with traditional credit data. The study moves iteratively from problem identification to model validation using the Design Science Research approach. By putting out a hybrid strategy for improved prediction, it highlights the significance of removing biases and increasing accuracy. TensorFlow and Python libraries are used to enable techniques like ensemble methods and deep learning. Data protection procedures are in place, and ethical concerns guarantee justice and openness. The study's objective is to test the model's efficiency while abiding by legal and ethical restrictions while using secondary data from Kaggle.

CHAPTER FOUR: RESULTS AND DISCUSSION

4.1 INTRODUCTION

This chapter will look into the results and discussion of all the research objectives presented in section 1.5

4.2 Results and Discussion

This section outlined the results of all the objectives. The first, second, third, and fourth objectives have been outlined in subsections 4.2.1,4.2.2,4.2.3 and 4.3.4, respectively.

4.2.1 Role of behavioral and traditional data in the existing credit scoring model

The first objective is to evaluate the role of behavioral and traditional data in the existing credit scoring model. The researcher did feature selection to determine the crucial elements for model construction. This approach minimizes model complexity and gets rid of unnecessary characteristics, which is important in machine learning and deep learning. By choosing only the most significant features, we improve the performance of the model. Data cleaning, description, and feature selection were undergone in the data analysis, and the features in Figure 4.9 were selected. Among the selected 15 features,14 will be categorized as the data to be used in predicting, while one (1) feature will be used as a target variable (Credit_Score).

Traditional Data	Behavioral Data
Age Annual_Income Monthly_Inhand_Salary Interest_Rate Num_of_Loan Credit_Utilization_Ratio Total_EMI_per_month Changed_Credit_Limit Num_Credit_Inquiries Outstanding_Debt	Credit_Mix Payment_of_Min_Amount Payment_Behaviour Delay_from_due_date **Target Variable** Credit_Score

Figure 4. 1: Final Set of Selected Features

The feature importance scores offer insightful information about the variables most influencing the model's ability to predict outcomes. The most important traditional feature is outstanding debt, with a score of 0.142815, which suggests that it plays a big part in evaluating creditworthiness and may be an indication of overstretching or financial stress. The interest rate follows closely with a score of 0.101716, underscoring its significance in determining borrowing rates and lenders' perceptions of risk. Additional behavioural measures that provide important insights into people's money management strategies and changing credit risk profiles include credit utilization ratios with (0.087646), credit limit (0.086062), Delay_from_due_date (0.085072), Age (0.058175) and annual income (0.069331) are important demographic characteristics. However, behavioural indications take precedence over them, highlighting the necessity for a comprehensive evaluation strategy.

Credit Mix (0.070928), Payment Behavior (0.042176), Payment of Minimum Amount (0.031570), and Delay from Due Date (0.085078) are examples of behavioural variables with significance scores that highlight their important roles in credit scoring models. A varied credit mix shows that different kinds of credit are managed responsibly, and creditworthiness is improved by regular payments and low delinquencies, which are signs of positive payment behaviour. On the other hand, persistently making little payments and letting payment delays happen can indicate financial difficulty and raise credit risk. Lenders can use these behavioural indicators to manage credit risk in their portfolios and make better lending decisions by gaining sophisticated insights into people's financial habits and activities. Lenders can improve their comprehension of borrowers' creditworthiness and make better lending decisions and risk management procedures by combining traditional and behavioral data elements.

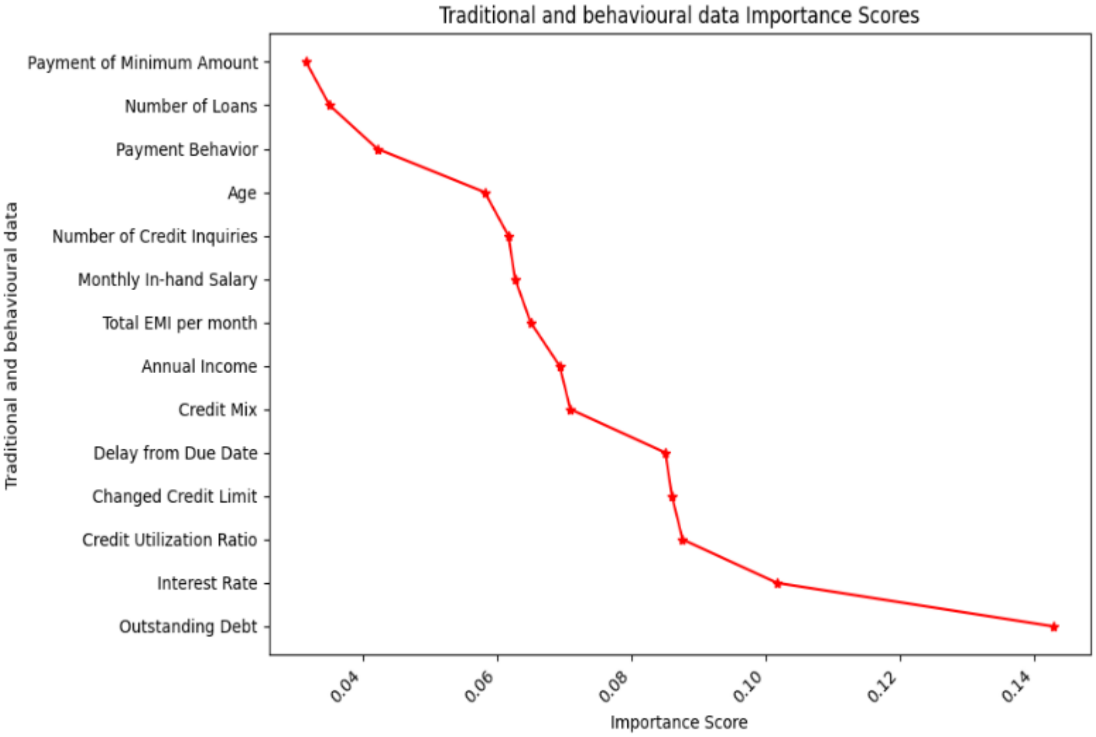


Figure 4. 2: Traditional and Behavioral data Importance score

To clearly understand and summarize the role of traditional and behavioural features in credit score prediction, the researcher used the SHAP (Shapley Additive exPlanations) tool. SHAP is a useful tool for understanding complex machine-learning models. It assists by highlighting the most crucial elements and their interactions. Selecting the appropriate features is facilitated by SHAP values, which order features according to their influence on the model's output. In order to verify the accuracy of the model, SHAP values help diagnose particular outcomes. SHAP increases the model's transparency and dependability by providing a detailed explanation of each prediction, highlighting the contributions of each feature. Biases in the model can be identified by SHAP. The graphs below show the visualization of the features.

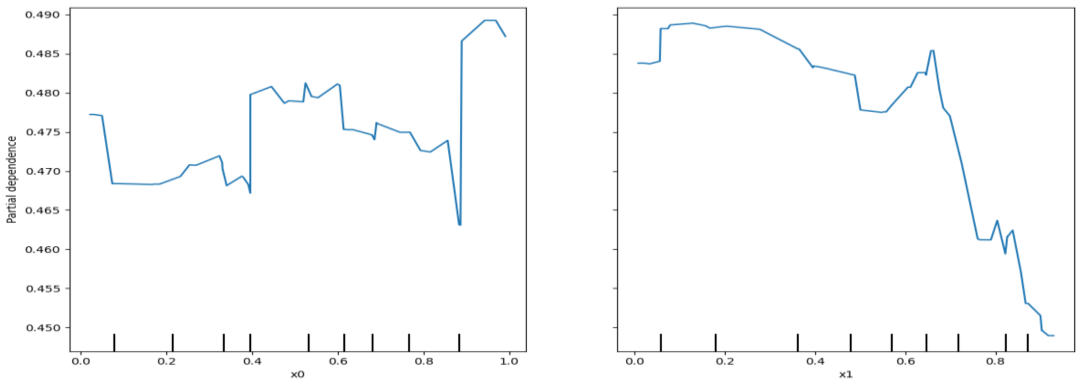


Figure 4. 3: Partial Dependence Plots (PDPs)

These are two Partial Dependence Plots (PDPs), each illustrating the connection between a feature and the model's forecast result while averaging out the effects of all other features. The Inherent Dependency Plots, which average out the impacts of other factors, show the link between specific features (x_0 and x_1) and the outcome predicted by the model. The Y-axis displays partial dependence, and the X-axis displays feature values.

While the plot for x_1 displays a general declining trend with smaller fluctuations, revealing a complicated, non-linear influence; the plot for x_0 shows considerable changes, indicating a non-linear impact.

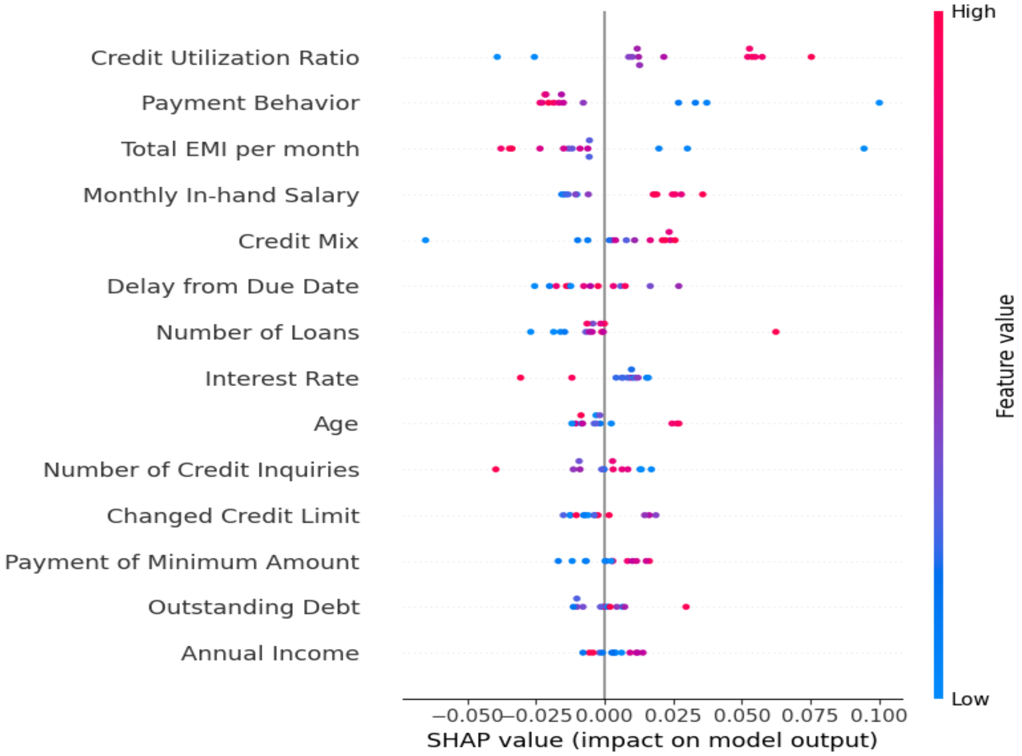


Figure 4. 4: SHAP summary plot

By showing the SHAP values for each feature, the SHAP summary plot shows how different features affect a deep learning model's output. Features are plotted along the Y-axis, and their effect on the model's predictions is indicated by the matching SHAP values on the X-axis. Based on the feature value (high for red, low for blue), each dot indicates a single prediction. The large range of SHAP values for factors like "Credit Utilization Ratio," "Payment Behavior," and "Total EMI per month" indicates that these features have a substantial impact. High values of these features typically increase the output of the model, but it can also be decreased by low values. On the other hand, the

SHAP values of factors like "Annual Income" and "Age" are centred around zero, indicating a lower influence on the predictions. This graphic aids in determining the most important features and how their values impact the results of the model.

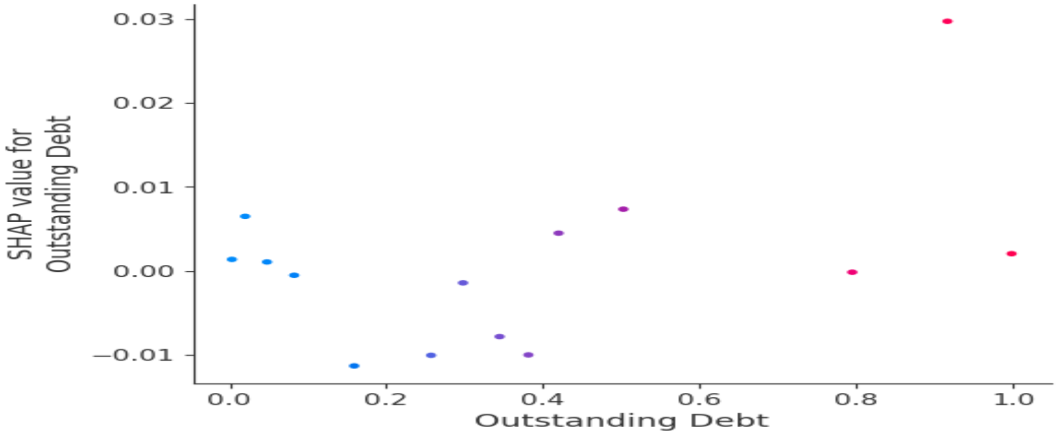


Figure 4. 5: Most important feature

The above figure shows a scatter plot that illustrates the impact of the "Outstanding Debt" feature on the model's predictions. The X-axis represents the "Outstanding Debt" values, and the feature's contribution to the model's prediction is indicated by the Y-axis, which displays the SHAP value. Based on the feature value, each dot is colored differently: blue corresponds to lower values, red to higher values, and purple to intermediate levels. Plotting indicates that the SHAP values generally increase as "Outstanding Debt" rises from 0.0 to 1.0, indicating that a higher outstanding debt boosts the model's prediction. Higher debt levels consistently contribute positively, whereas lower debt levels exhibit greater variability, with SHAP values near zero or slightly negative. The way that "Outstanding Debt" affects the model's prediction is clearly shown by this plot.

4.2.2 Model development

The second objective of the research was to develop a hybrid deep learning model that predicts credit scores by integrating traditional and behavioural data. To achieve this objective, the researcher developed a hybrid model that combined the strengths of deep neural networks (DNNs) and recurrent neural networks (RNNs) to create a better model that will correctly predict creditworthiness.

The researcher started by developing two separate neural networks for credit scoring. To be more precise, the researcher developed Deep Neural Networks (DNNs) and Recurrent Neural Networks (RNNs) specifically for credit score prediction.

Deep Neural Networks (DNNs) are Feed Forward Networks (FFNNs), where data goes from the input layer to the output layer without ever traveling backward. The links connecting the layers are one-way, forward-moving, and never come into contact with another node. DNNs are strong tools for large data and complicated tasks because these layers can train to represent data at ever-higher degrees of abstraction; however, due to their high capacity to learn complex patterns, they suffer overfitting, and they don't have a memory to remember what they started. This problem is handled with a Recurrent Neural Network (RNN), which is an FFNN has a temporal twist and can process input sequences by utilizing their internal state or memory, making it suitable for this research.

In this research, a hybrid model was developed to combine both the strengths of DNNs and RNNs in order to come up with a better model that will predict creditworthiness correctly. To enhance efficiency and class balancing in the developed model, Synthetic Minority Over-Sampling Technique (SMOTE) and SMOTE + Edited Nearest Neighbors

(SMOTEENN) were used to reduce overfitting in the model. SMOTE is an oversampling technique that balances the dataset by creating artificial samples for the minority class, which enhances the model's capacity to generalize from the training set. In this research, SMOTE+ENN improves the robustness of the model by balancing the dataset and cleaning it up by eliminating noisy samples using the Edited Nearest Neighbors algorithm. By utilizing these strategies, the model's capacity to manage class imbalances and minimize overfitting will increase, improving its performance and dependability when generating predictions from the data. In the following subsections, 4.3.2.1 and 4.3.2.2, the researcher discussed the two models, DNNs and RNNs, and their output separately, and in section 4.3.2.3 the hybrid model (RNNs DNNs)) which merges the two models that have been developed separately.

4.2.2.1.1 Training and Validation Loss Curves for RNN

A Recurrent Neural Network (RNN) training and validation loss across 100 epochs is depicted in Figure 4.6

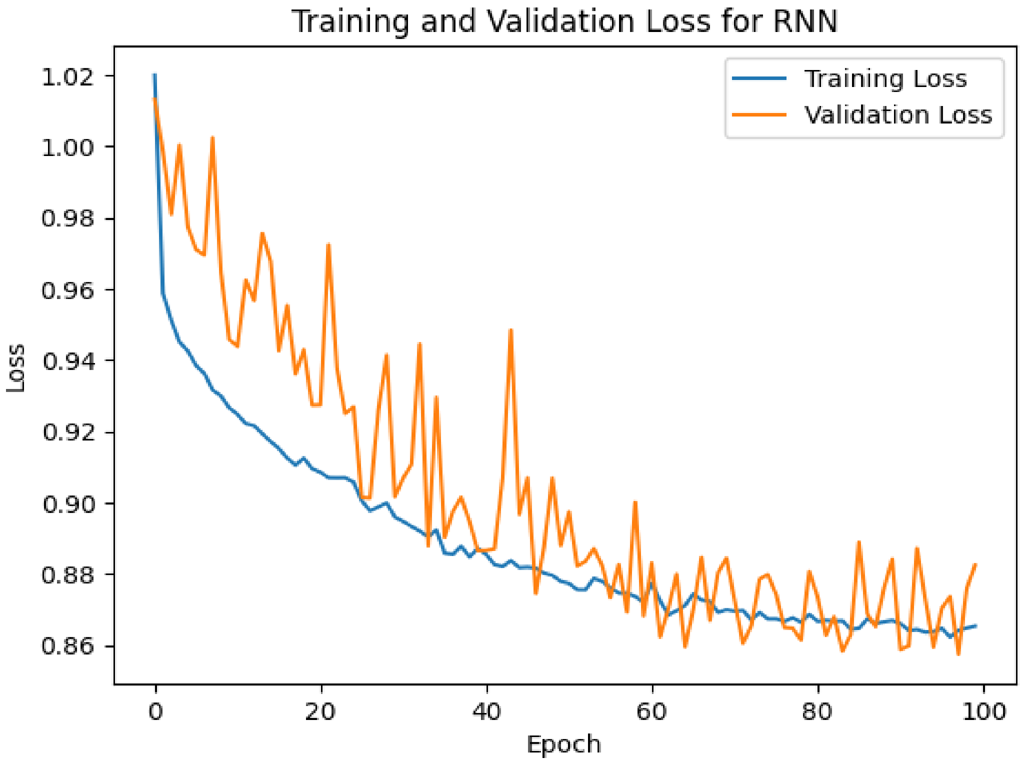


Figure 4. 6*:* Loss Function for RNNs Model

The loss is represented by the y-axis and epochs by the x-axis. The orange line denotes validation loss, and the blue line shows training loss. Both losses show effective learning as they begin high (around 1.02) and gradually decline. That being said, the validation loss exhibits variations, indicating unpredictability in performance on the validation set, while the training loss diminishes steadily. Overfitting, in which the model performs well on training data but inconsistently on validation data, may cause this variability. Although the RNN is learning efficiently overall, more stages such as regularization, dropout, or additional data may lessen volatility and enhance generalization

4.2.2.1.2 Root Mean Squared Error (RMSE) and Mean Squared Error (MSE) for RNN

The model's performance was evaluated using the chosen evaluation metrics, Root Mean Squared Error (RMSE) and Mean Squared Error (MSE), which resulted in an RMSE of 0.7456 and an MSE of 0.556, as shown below. These results align with traditional cutoff points that indicate reliable data prediction.

4.2.2.1.3 Confusion matrix showing Prediction vs. Actual labels for RNN

The RNN model's performance on a three-class classification issue is displayed on the confusion matrix, which has predicted labels on the x-axis and true labels on the y-axis. Along the diagonal, it displays the number of accurate predictions for each class (6542 for class 0, 7717 for class 1, and 4459 for class 2) and the misclassifications in the cells off the diagonal. Notable misclassifications of the model include class 1 being predicted for 4044 occurrences of actual class 0 and class 2 being predicted for 4118 instances of actual class 1. Darker hues indicate higher numbers and colour intensity reflects the frequency of occurrences. This matrix aids in assessing the correctness of the model and pinpointing areas in need of development.

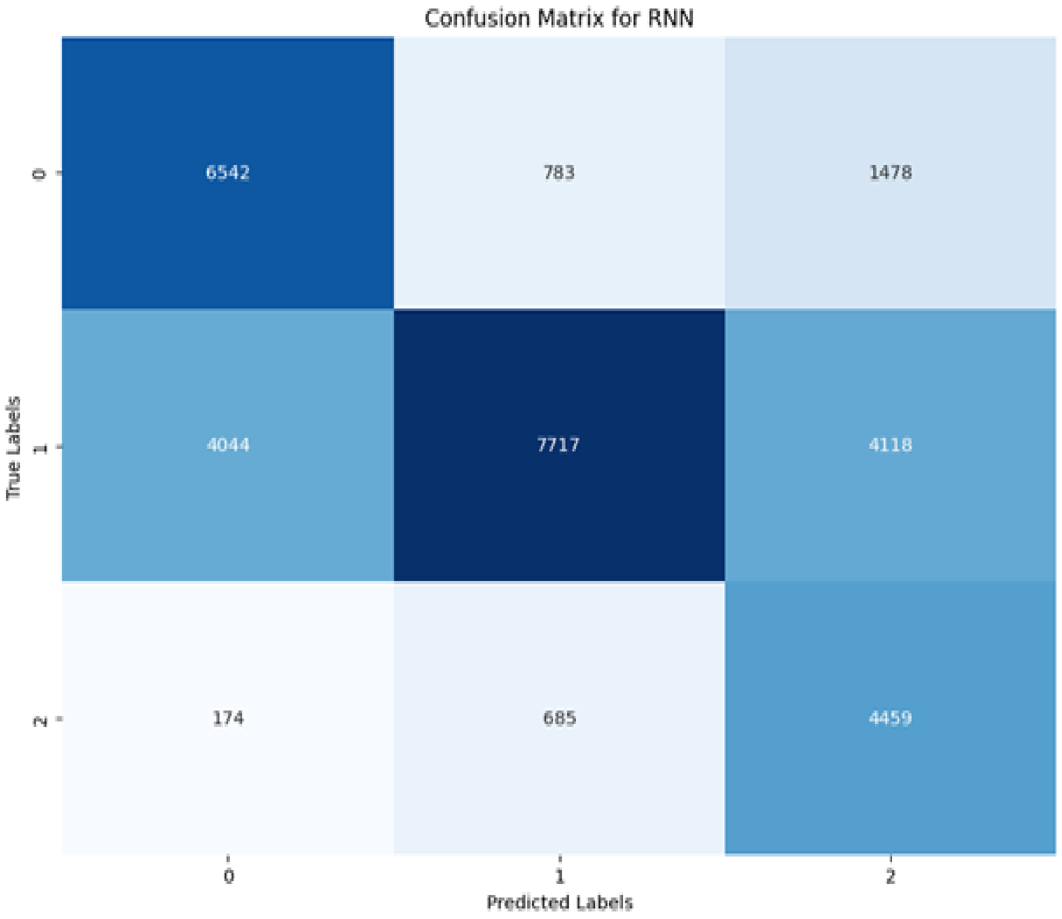


Figure 4. 7: Plotting predictions for RNNs using a confusion matrix

Based on the confusion matrix, The RNN model is performing well. A sizable portion of the examples for each class 6542 for class 0, 7717 for class 1, and 4459 for class 2 are correctly classified by the model, suggesting that it has some understanding of each class.

4.2.2.1.4 ROC and AUC curve for RNN

The ROC curves depict the RNN model's performance in three classes; the areas under the curve (AUC) are 0.79 for class 0, 0.74 for class 1, and 0.85 for class 2. These graphs demonstrate how well the model distinguishes between classes 2 and 1, and slightly low for class 0, and how well it does for class 2. The model performs better than random guessing, as the curves above the diagonal baseline. Overall, the model's performance for class 1 is Slightly low, indicating the need for more changes, particularly for class 1, even though it performs very well for class 2 and well for class 0.

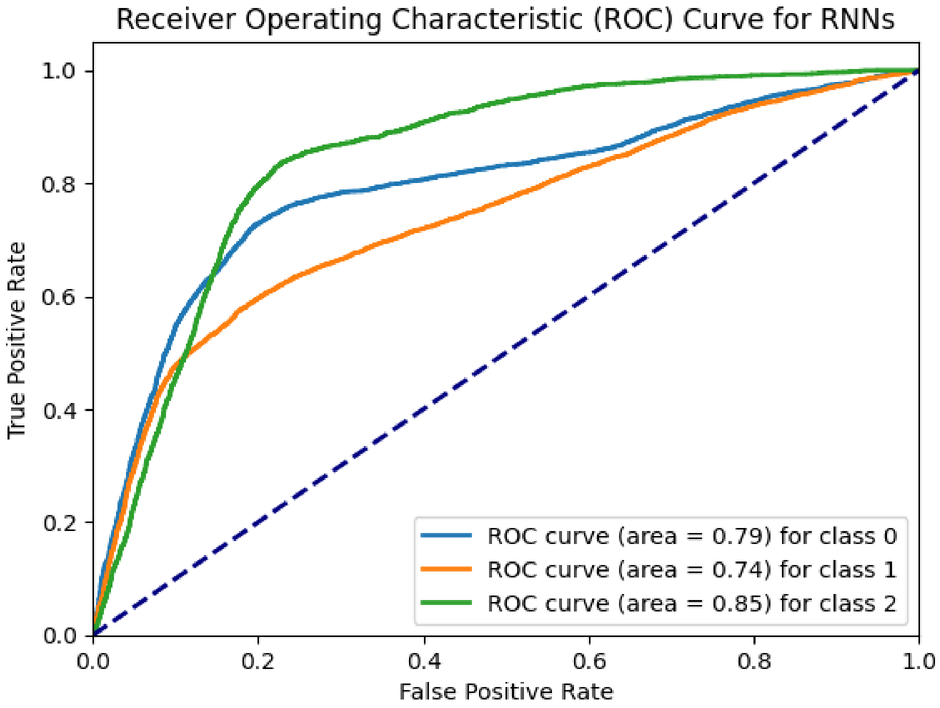
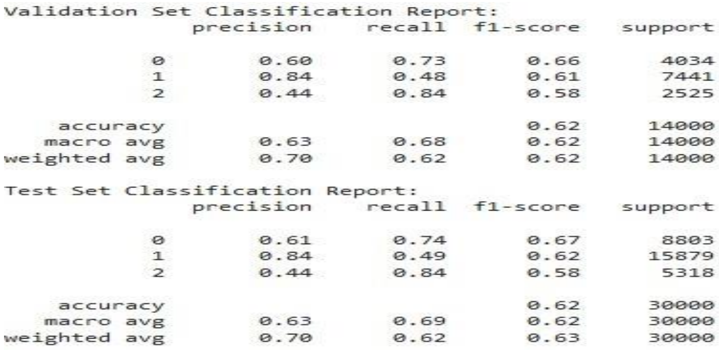


Figure 4. 8: RNNs ROC and AUC curve

4.2.2.1.5 Classification Report for RNN

Precision in the validation set classification report in Figure 4.20 indicates the frequency of correct positive predictions made by the model; in this research model, class 0 has a precision of 0.60, class 1 has a precision of 0.84, and class 2 has a precision of 0.44. Recall, also known as sensitivity, measures how well the model can locate all pertinent instances; class 0 had a recall of 0.73, class 1 had a recall of 0.48, and class 2 had a recall of 0.84. The balance between precision and recall is shown by the F1-score, which is a harmonic mean of the two metrics with class 0 at 0.66, class 1 at 0.61, and class 2 at 0.58. The number of real instances of each class is shown in the support column. With a macro average F1-score of 0.62, the validation set's overall accuracy is 0.62. Comparable metrics are given for the test set in the test set classification report, demonstrating consistent performance patterns between the two sets.



```
Validation Set Classification Report:
              precision    recall  f1-score   support

           0       0.60      0.73      0.66      4034
           1       0.84      0.48      0.61      7441
           2       0.44      0.84      0.58      2525

    accuracy                           0.62     14000
   macro avg       0.63      0.68      0.62     14000
weighted avg       0.70      0.62      0.62     14000

Test Set Classification Report:
              precision    recall  f1-score   support

           0       0.61      0.74      0.67      8803
           1       0.84      0.49      0.62     15879
           2       0.44      0.84      0.58      5318

    accuracy                           0.62     30000
   macro avg       0.63      0.69      0.62     30000
weighted avg       0.70      0.62      0.63     30000
```

Figure 4. 9*:* Classification report for RNNs model.

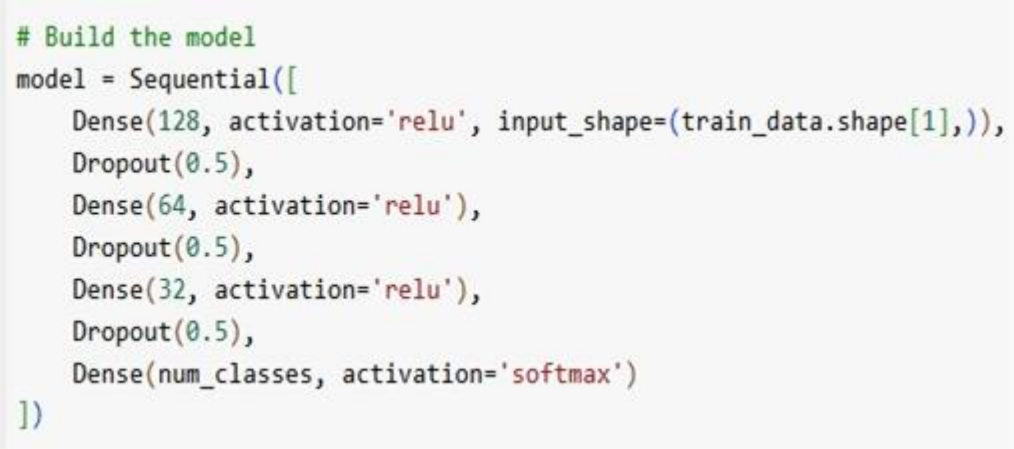
With an accuracy of 0.62 for both the test and validation sets, the model RNN correctly predicted the class labels for almost 62% of the cases in each dataset.

4.2.2.2 Deep Neural Network (DNN)

A DNN or deep neural network model was developed for this thesis. A DNN is an Artificial Neural Network (ANN) model with more than three layers. An input layer with a predetermined number of features makes up the DNN model developed in this study. The model code in Figure 4.10 consists of three (3) hidden layers with 128, 64, and 32 units each. The output layer of the model uses the Rectified Linear Unit (ReLU) or Hyperbolic Tangent (tanh) activation function to terminate the model.

Model development was followed by compilation using the Adam optimization algorithm with a Mean Square Error (MSE) loss function. During the training phase, the full dataset is iterated over 100 times (epochs=100) using train_data and train_targets for input features and corresponding labels, respectively. In order to facilitate memory-saving and enable efficient parameter updates, data is separated into smaller batches of size 32 (batch_size=32). Without changing the model's training, validation data (validation_data) is used to evaluate model performance and spot overfitting. Gradients beyond a threshold (clipvalue=1.0) are capped by the Gradient Clipping callback, which preserves numerical stability and avoids explosive gradients during training. Together, these components provide a thorough training regimen that includes batch size, epochs, validation data use, and gradient clipping for efficient model optimization and

performance assessment.



```
# Build the model
model = Sequential([
    Dense(128, activation='relu', input_shape=(train_data.shape[1],)),
    Dropout(0.5),
    Dense(64, activation='relu'),
    Dropout(0.5),
    Dense(32, activation='relu'),
    Dropout(0.5),
    Dense(num_classes, activation='softmax')
])
```

Figure 4. 10: Snippet code for DNNs model

The model architecture in Figure 4.11 displays what the DNNs model developed looks like. A batch normalization layer stabilizes training after the input layer for 14 features in the given Deep Neural Networks (DNNs) model. Four dense (completely linked) hidden layers exist in it, with 128, 64, and 32 neurons in each. Dropout layers are placed after each layer to provide regularization and avoid overfitting. A three-neuron output layer at the network's end makes it appropriate for a three-class classification challenge. Multiple layers of dense connections and regularization in this structure provide strong classification and efficient feature transformation.

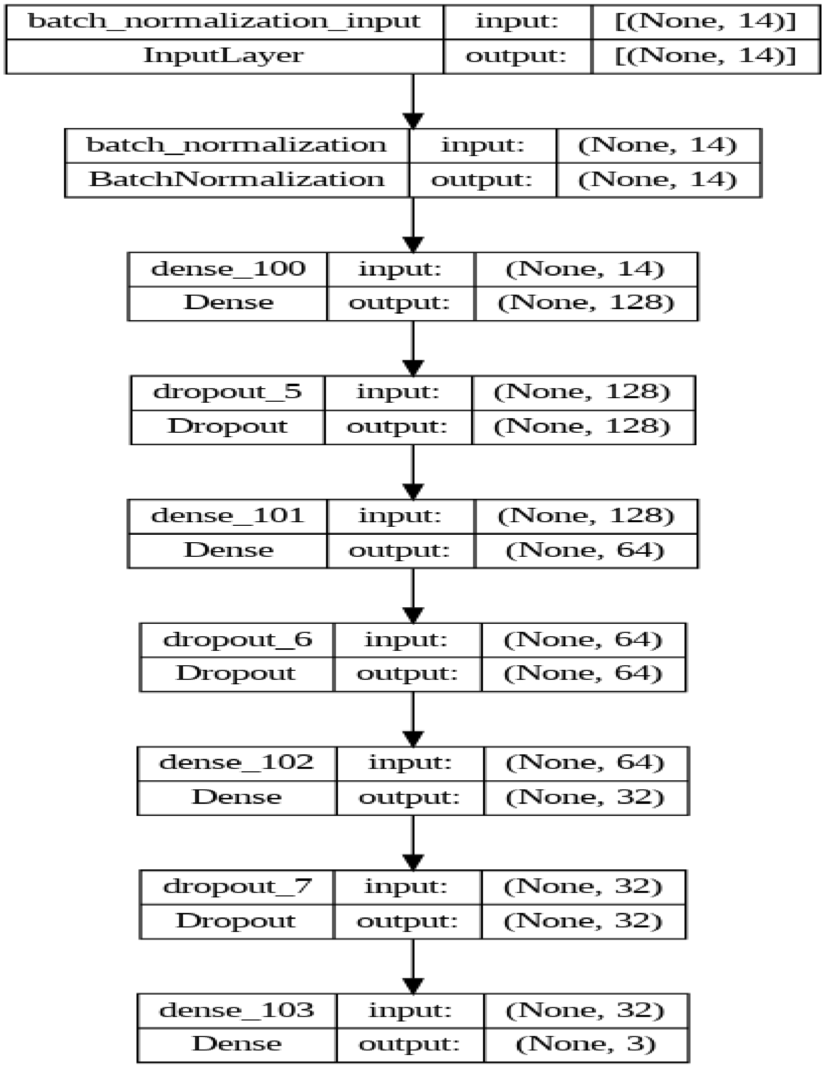
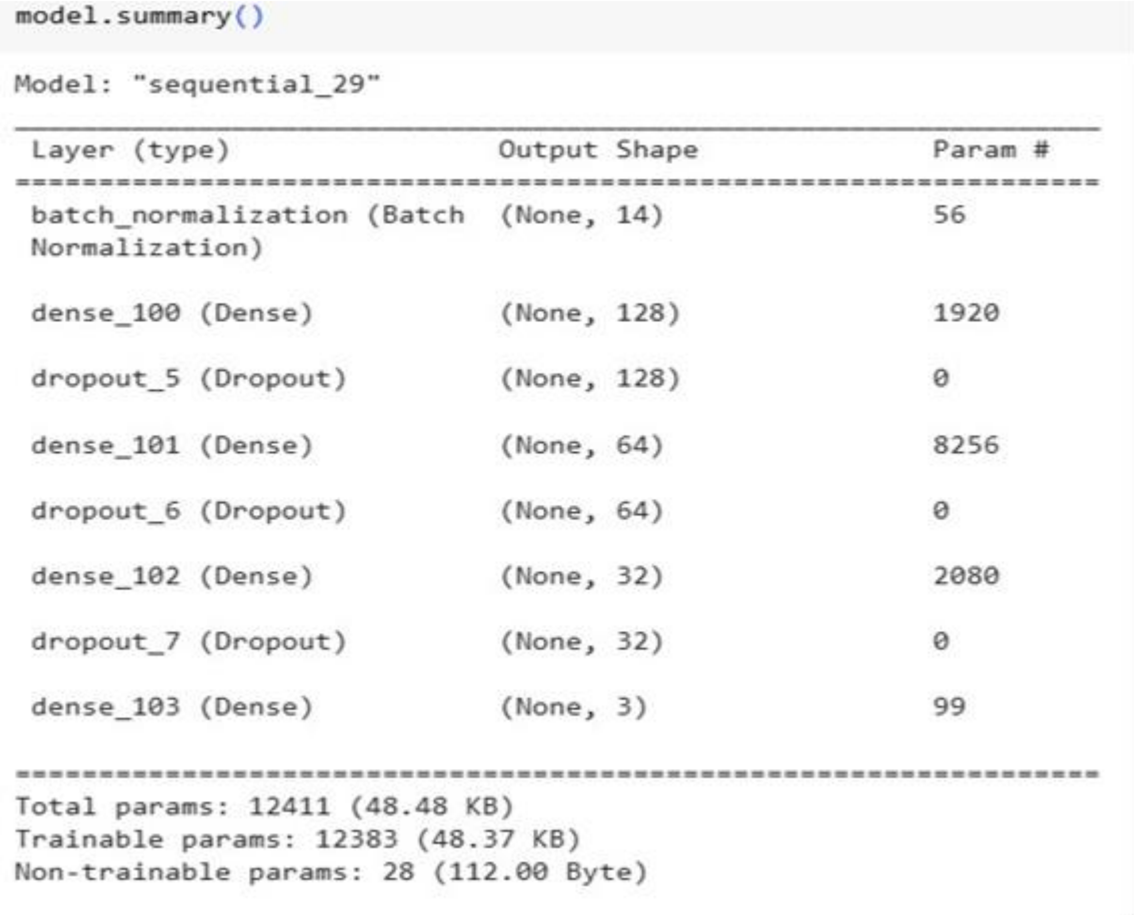

batch_normalization_input | input: | [(None, 14)]
InputLayer | output: | [(None, 14)]
batch_normalization | input: | (None, 14)
BatchNormalization | output: | (None, 14)
dense_100 | input: | (None, 14)
Dense | output: | (None, 128)
dropout_5 | input: | (None, 128)
Dropout | output: | (None, 128)
dense_101 | input: | (None, 128)
Dense | output: | (None, 64)
dropout_6 | input: | (None, 64)
Dropout | output: | (None, 64)
dense_102 | input: | (None, 64)
Dense | output: | (None, 32)
dropout_7 | input: | (None, 32)
Dropout | output: | (None, 32)
dense_103 | input: | (None, 32)
Dense | output: | (None, 3)

Figure 4. 11: The DNN model architecture

The DNN model summary overview is shown in Figure 4.12 below, which shows the parameter count, the types of layers used, and their corresponding output forms. The normalization layer of the model consists of 56 non-trainable parameters, which are weights that do not change during training and are not modified by backpropagation. The mean and standard deviation values that are essential for activation during model testing are retained in this layer. There are a total of 12383 trainable parameters in the three dense layers: the first contains 1920 trainable parameters, the second contains 8256 trainable parameters, the third contains 2080 trainable parameters, and the last layer has 99 trainable parameters. Meanwhile, there are 28 non-trainable parameters, for a total of 12411 model parameters.



```
model.summary()
```

```
Model: "sequential_29"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 batch_normalization (Batch  (None, 14)                56
 Normalization)

 dense_100 (Dense)           (None, 128)               1920

 dropout_5 (Dropout)         (None, 128)               0

 dense_101 (Dense)           (None, 64)                8256

 dropout_6 (Dropout)         (None, 64)                0

 dense_102 (Dense)           (None, 32)                2080

 dropout_7 (Dropout)         (None, 32)                0

 dense_103 (Dense)           (None, 3)                 99

=================================================================
Total params: 12411 (48.48 KB)
Trainable params: 12383 (48.37 KB)
Non-trainable params: 28 (112.00 Byte)
```

Figure 4. 12: DNN Summary Model

4.2.2.2.1 Training and validation loss curves for DNN

The training and validation losses of a DNN model across 100 epochs are displayed in Figure 4.13 in this research. Both losses are large at first but quickly decline, suggesting efficient learning. The validation loss (orange line) also declines but varies more than the training loss (blue line), which constantly decreases and smoothens out. This variation indicates that although the model may be having some variance problems, in general, the model is learning effectively.

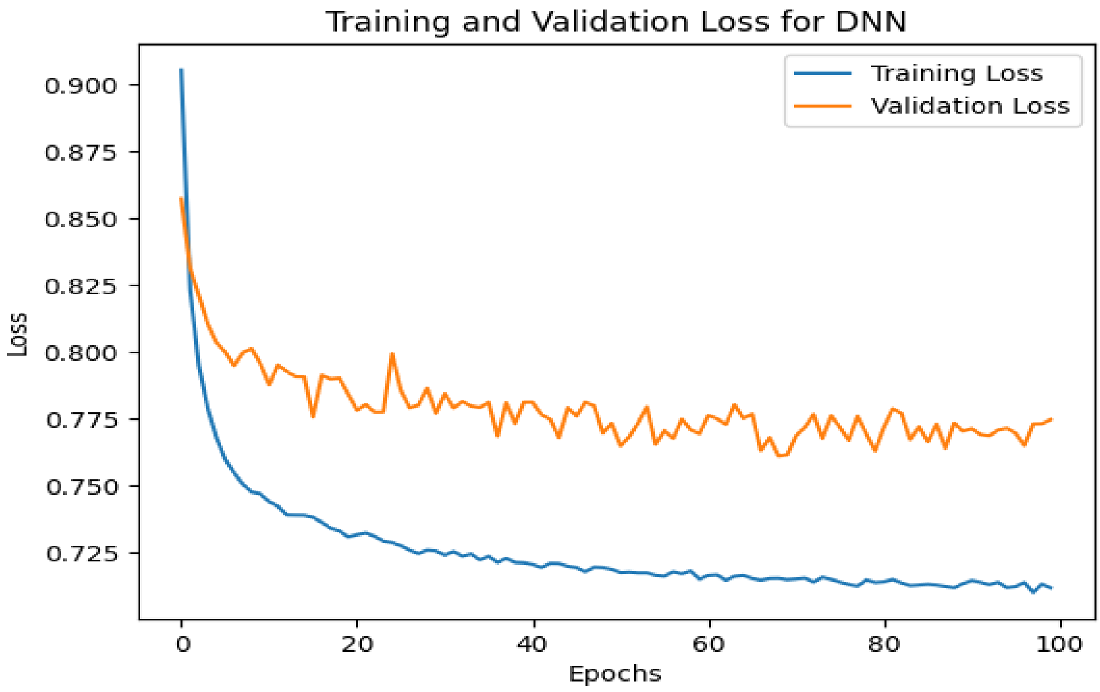


Figure 4. 13: Training and validation loss curves

As the model optimizes its parameters, both loss curves start high and progressively decrease. The near alignment of the validation and training losses indicates strong generalization. Furthermore, the absence of significant divergence between the loss curves implies less overfitting.

4.2.2.2.2 Root Mean Squared Error (RMSE) and Mean Squared Error (MSE) for DNN

The DNN model is appropriate for precise data prediction because it produced RMSE and MSE values of 1.142 and 1.304, respectively. An RMSE of 1.142 indicates reasonable but potentially improved model performance, meaning that the model's predictions are, on average, off by roughly 1.142 units from the actual values.

4.2.2.2.3 Confusion matrix for predicted vs. True labels for DNN

The Deep Neural Network (DNN) model's performance on this study's credit score classification problem is displayed in the confusion matrix in Figure 4.14. The model properly classified 6584 occurrences of class 0 but incorrectly classified 937 instances as class 1 and 1282 examples as class 2. It successfully identified 8742 cases for class 1 but incorrectly identified 3524 as class 2 and 3613 as class 0. It successfully identified 4357 instances for class 2 but incorrectly identified 182 as class 0 and 779 as class 1. The model's strengths and potential areas for error reduction are highlighted by the diagonal values, which indicate correct classifications and off-diagonal values, which show misclassifications.

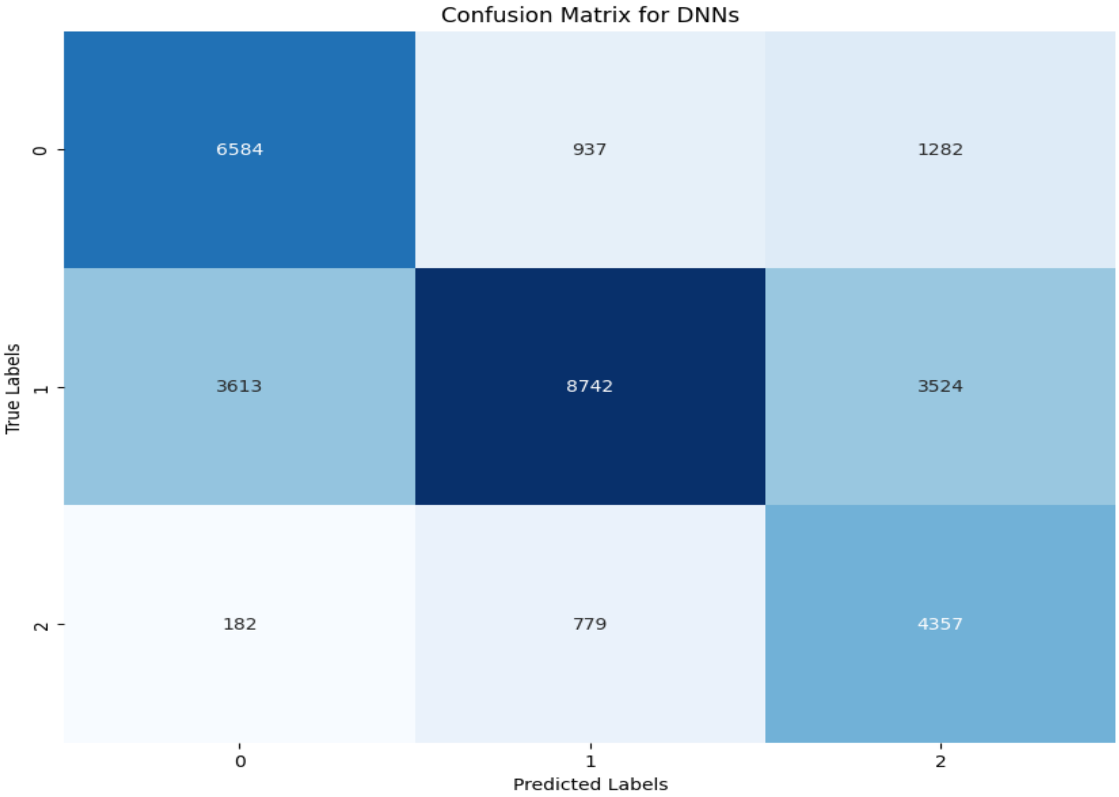


Figure 4. 14: Plotting predictions for DNNs using confusion matrix

The model developed in this research shows reasonable performance with many correct classifications, suggesting it has learned to distinguish between classes to some extent. The model's performance can be considered moderately good.

4.2.2.2.4 Receiver Operating Characteristic (ROC) and Area Under the Curve (AUC)for DNN

The model's capacity to discriminate between classes is gauged by the Receiver Operating Characteristic (ROC) area under the curve (AUC) in Figure 4.15. The AUC values for your model are 0.84 for class 0, 0.74 for class 1, and 0.85 for class 2, which show that classes 0 and 2 perform very well while class 1 performs fairly.

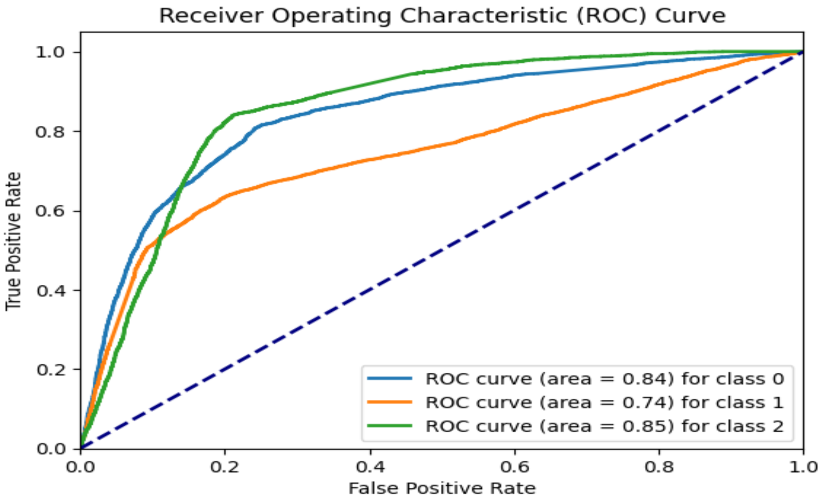
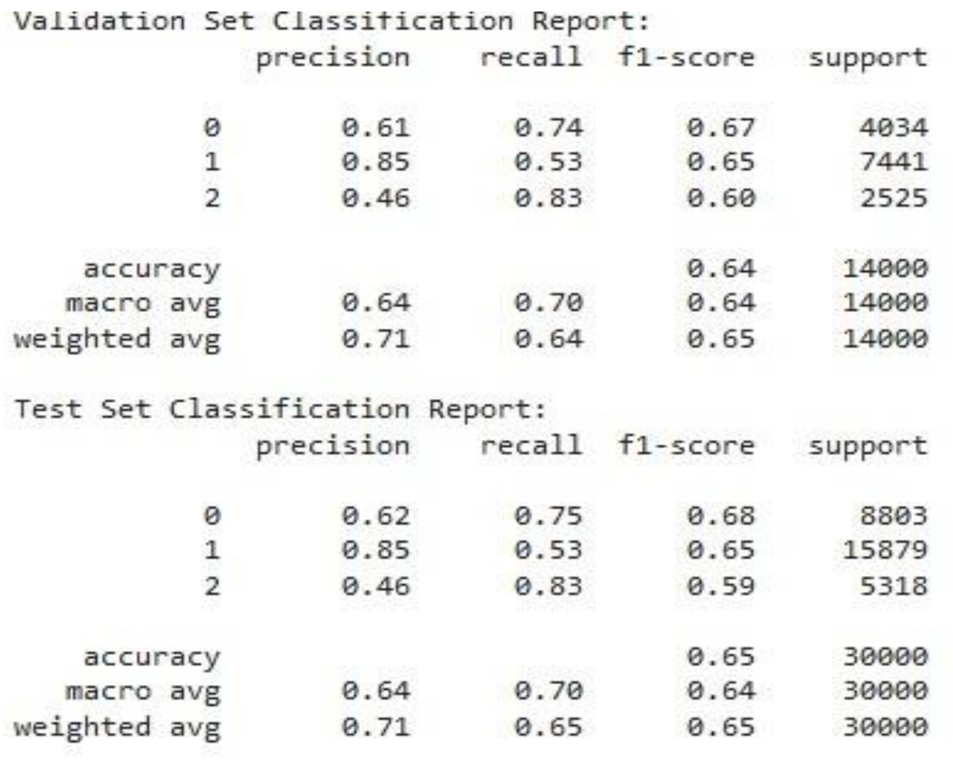


Figure 4. 15: DNN ROC and AUC curve

These metrics indicate that although there is potential for improvement in class 1 differentiation, the model can consistently discriminate instances of classes 0 and 2 from other examples. The AUC-ROC Score for the DNN model is 0.7504. When developing, evaluating, and validating models, AUC is a useful metric that may be used to guide additional refinement to improve the model's discriminatory power across all classes and assess classification performance.

4.2.2.2.5 Classification Reports for DNN

For each class (0, 1, and 2) in DNN, the precision, recall, and F1-score metrics are given together with the corresponding support values for the validation and test set classification reports in Figure 4.16. Class 1 has an F1-score of 0.65, a recall of 0.53, and an accuracy of 0.85 for the validation set, while class 0 has an F1-score of 0.67, a recall of 0.74, and a precision of 0.61. Class 2's F1-score is 0.60, recall is 0.83, and precision is 0.46. The validation set yielded an overall accuracy of 0.64, with a weighted average F1-score of 0.65 and a macro F1-score of 0.64.



```
Validation Set Classification Report:
              precision    recall  f1-score   support

           0       0.61      0.74      0.67      4034
           1       0.85      0.53      0.65      7441
           2       0.46      0.83      0.60      2525

    accuracy                           0.64     14000
   macro avg       0.64      0.70      0.64     14000
weighted avg       0.71      0.64      0.65     14000

Test Set Classification Report:
              precision    recall  f1-score   support

           0       0.62      0.75      0.68      8803
           1       0.85      0.53      0.65     15879
           2       0.46      0.83      0.59      5318

    accuracy                           0.65     30000
   macro avg       0.64      0.70      0.64     30000
weighted avg       0.71      0.65      0.65     30000
```

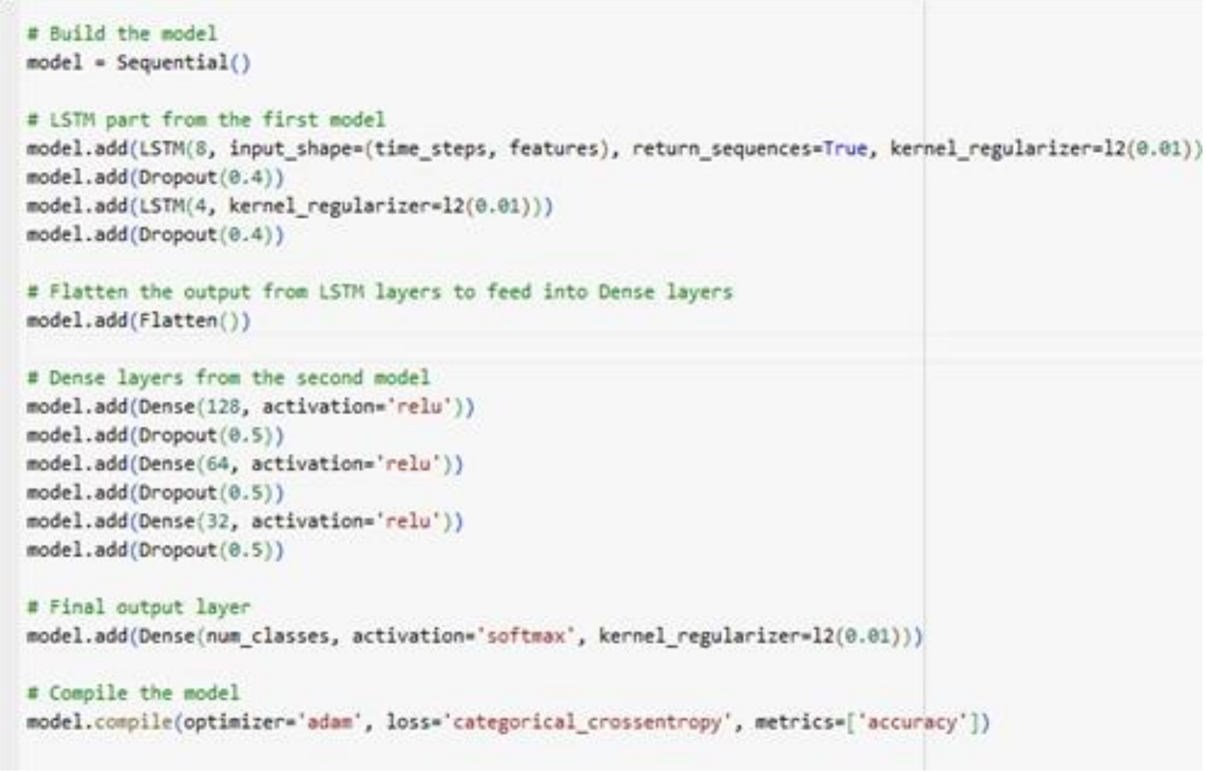
Figure 4. 16: Classification report for DNNs model.

In the test set, class 0 has 0.62 precision, 0.75 recall, and 0.68 F1 score; class 1 has 0.85 precision, 0.53 recall, and 0.65 F1 score. (see figure above). Class 2's F1-score is 0.59, recall is 0.83, and precision is 0.46. With a weighted average F1-score of 0.65 and macro F1-score of 0.64, the test set's overall accuracy is 0.65. The model performs well in this study according to the precision, recall, and F1 scores for each class in the validation and test sets. Certain classes perform better than others in terms of recall and precision. Although the model's overall accuracy of 0.64 to 0.65 indicates modest performance.

4.2.2.3 Hybrid Deep Learning Model

This research developed a hybrid deep learning model by combining the RNNs discussed in section 4.3.2.1 with the DNNs model discussed in section 4.3.2.2.

This thesis has developed a hybrid model consisting of two types of neural networks, as seen in Figure 4.17. Recurrent Neural Networks (RNNs) with two Long Short-Term Memory (LSTM) layers with dropout layers, input shape adjusted to 3-dimensional space, and a dense output layer; and a deep neural network (DNN) with three hidden layers (128, 64, and 32 units) using Rectified Linear Unit (ReLU) or Hyperbolic Tangent (tanh) activation functions. Using the Mean Square Error (MSE) loss function and the Adam optimization strategy, the model is trained and tested over a batch size of 6 and 100 epochs. This can be summarized in the snippet code below.



```
# Build the model
model = Sequential()

# LSTM part from the first model
model.add(LSTM(8, input_shape=(time_steps, features), return_sequences=True, kernel_regularizer=l2(0.01))
model.add(Dropout(0.4))
model.add(LSTM(4, kernel_regularizer=l2(0.01)))
model.add(Dropout(0.4))

# Flatten the output from LSTM layers to feed into Dense layers
model.add(Flatten())

# Dense layers from the second model
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(64, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(32, activation='relu'))
model.add(Dropout(0.5))

# Final output layer
model.add(Dense(num_classes, activation='softmax', kernel_regularizer=l2(0.01)))

# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

Figure 4. 17: Snippet code for Hybrid model

This hybrid model has several coupled RNN and DNN layers, as shown in Figure 4.18. Two Long Short-Term Memory (LSTM) layers at the architecture's top allow the model to recognize temporal dependencies in the data. To prevent overfitting, dropout layers are placed between each LSTM layer and randomly a portion of the input units during training. The LSTM output is then flattened to create a one-dimensional array, making it easier to work with later dense layers. The data is then subjected to linear transformations by three dense layers (128, 64, and 32) with decreasing units per layer. To further regularize the model, a dropout layer comes after each dense layer.

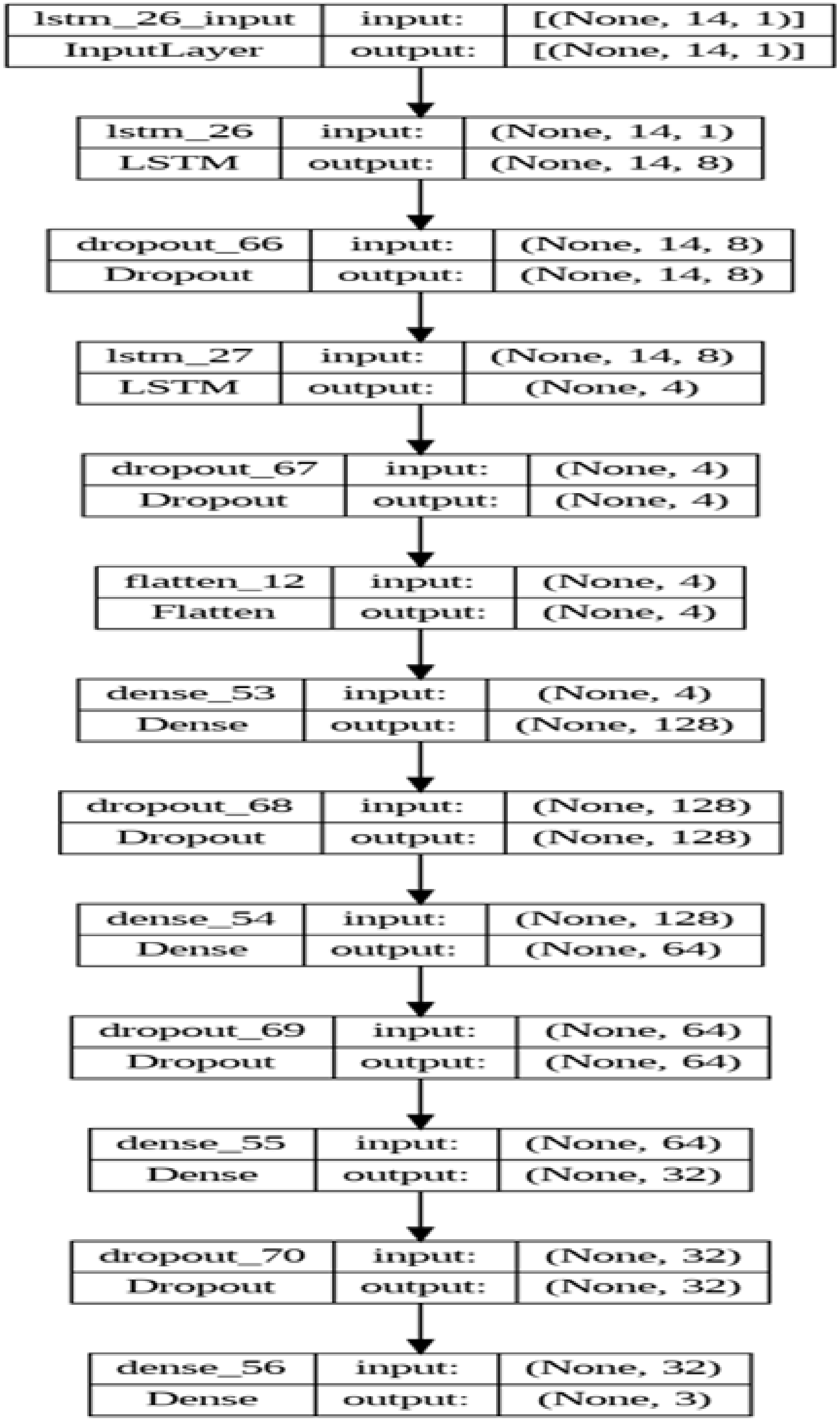
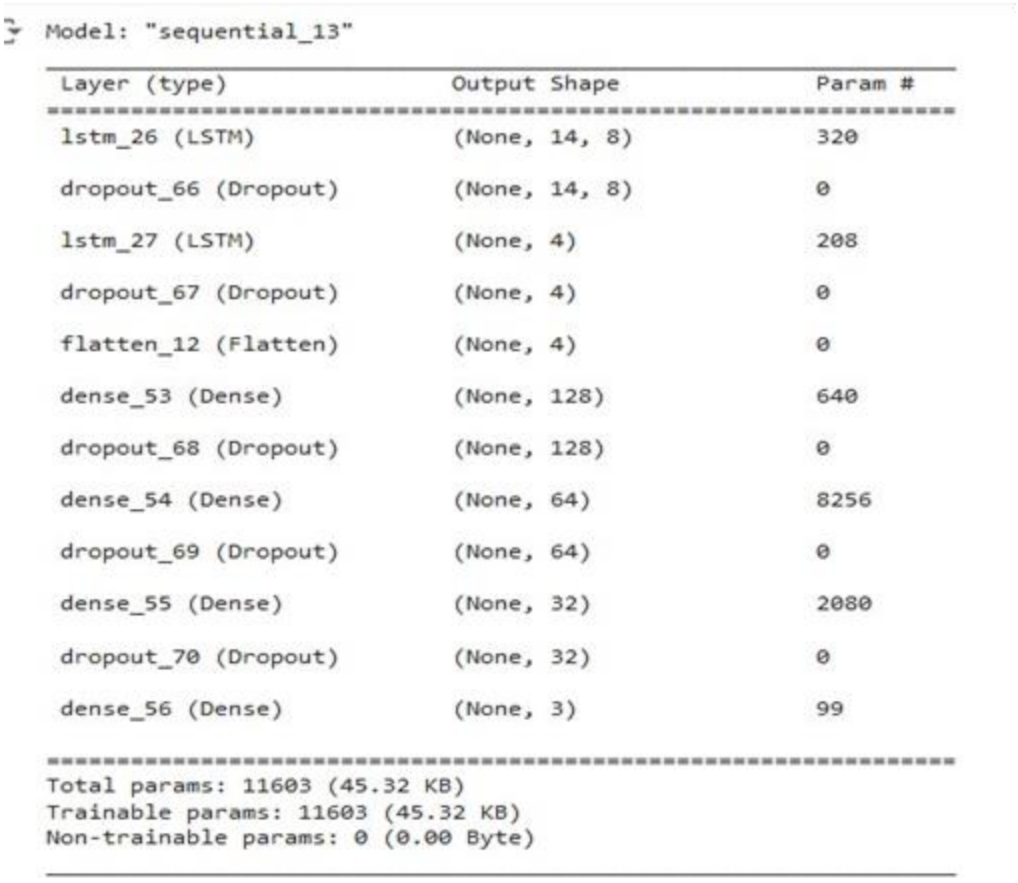

lstm_26_input | input: | [(None, 14, 1)]
InputLayer | output: | [(None, 14, 1)]
lstm_26 | input: | (None, 14, 1)
LSTM | output: | (None, 14, 8)
dropout_66 | input: | (None, 14, 8)
Dropout | output: | (None, 14, 8)
lstm_27 | input: | (None, 14, 8)
LSTM | output: | (None, 4)
dropout_67 | input: | (None, 4)
Dropout | output: | (None, 4)
flatten_12 | input: | (None, 4)
Flatten | output: | (None, 4)
dense_53 | input: | (None, 4)
Dense | output: | (None, 128)
dropout_68 | input: | (None, 128)
Dropout | output: | (None, 128)
dense_54 | input: | (None, 128)
Dense | output: | (None, 64)
dropout_69 | input: | (None, 64)
Dropout | output: | (None, 64)
dense_55 | input: | (None, 64)
Dense | output: | (None, 32)
dropout_70 | input: | (None, 32)
Dropout | output: | (None, 32)
dense_56 | input: | (None, 32)
Dense | output: | (None, 3)

Figure 4. 18: Hybrid Model Architecture

In conclusion, the output layer comprises three units, which match the number of classes involved in the classification process. In total, 11603 trainable parameters make up this architecture, which aims to learn representations that maximize performance for predicting credit scores.



```
Model: "sequential_13"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_26 (LSTM)              (None, 14, 8)             320

 dropout_66 (Dropout)        (None, 14, 8)             0

 lstm_27 (LSTM)              (None, 4)                 208

 dropout_67 (Dropout)        (None, 4)                 0

 flatten_12 (Flatten)        (None, 4)                 0

 dense_53 (Dense)            (None, 128)               640

 dropout_68 (Dropout)        (None, 128)               0

 dense_54 (Dense)            (None, 64)                8256

 dropout_69 (Dropout)        (None, 64)                0

 dense_55 (Dense)            (None, 32)                2080

 dropout_70 (Dropout)        (None, 32)                0

 dense_56 (Dense)            (None, 3)                 99

=================================================================
Total params: 11603 (45.32 KB)
Trainable params: 11603 (45.32 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figure 4. 19**: Hybrid Model Summary**

The performance of the hybrid model in this research has been evaluated in the form of Model Evaluation Results (MSE and RMSE), Plotting predictions where actual values are plotted against predicted values using a confusion matrix, Receiver Operating Characteristic (ROC) area under the curve (AUC) and Classification report metrics.

4.2.2.3.1 Hybrid Model Receiver Operating Characteristic (ROC) Area Under the Curve (AUC)

Figure 4.20 illustrates a Receiver Operating Characteristic (ROC) curve, which depicts the trade-off between True Positive Rate (TPR) and False Positive Rate (FPR) and serves to evaluate the performance of a binary classification model. The Y-axis represents the true positive rate, while the X-axis represents the false positive rate. The orange line depicts the ROC curve for training data, demonstrating better performance with an Area Under the Curve (AUC) of 0.92. The validation ROC curve, represented by the blue line, shows good performance on the validation data with an AUC of 0.79.

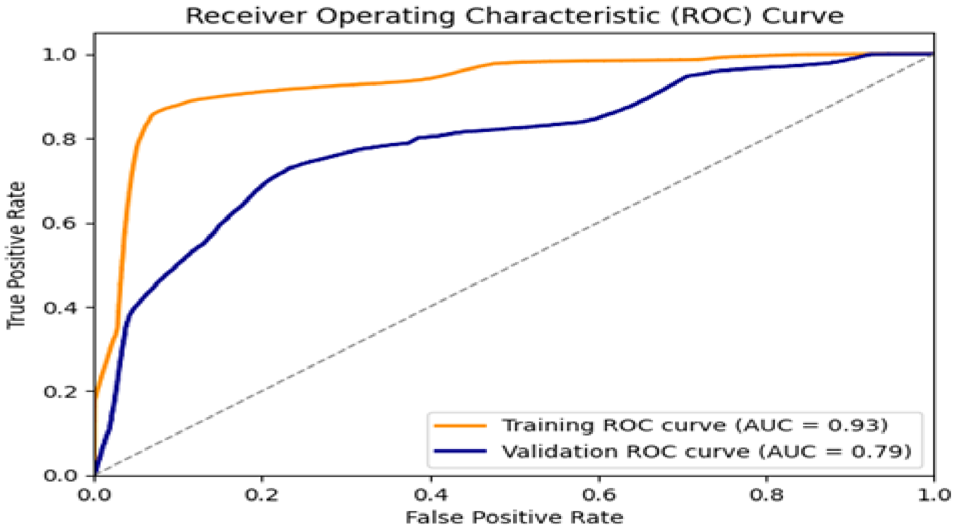


Figure 4. 20: Area Under the Curve (AUC)

Receiver Operating Characteristic (ROC) curves for a hybrid model are shown in Figure 4.21, with separate curves for each class (0, 1, and 2). The True Positive Rate (TPR) against False Positive Rate (FPR) is plotted against the ROC curve at different thresholds, and the AUC (Area Under the Curve) values provide a summary of the classifier's performance for each class. Class 2 has the best AUC of 0.85, followed by Class 0 with

0.82, Class 1 with 0.74, and Class 2 with 0.85. The improved performance of the model is indicated by all ROC curves positioned above the diagonal line. All things considered, the classifier performs well overall, especially for Class 2. However, there is room for improvement, especially for Class 1. ROC curves and AUC values are used to thoroughly assess the model's class-separation capabilities.

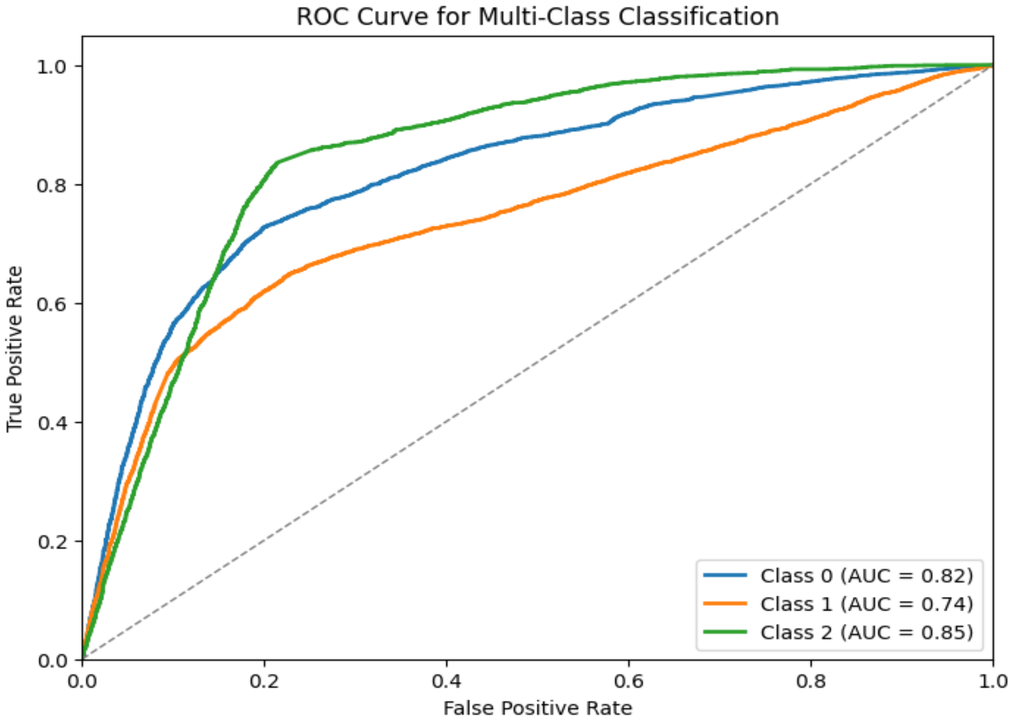


Figure 4. 21: Receiver Operating Characteristic (ROC)

The hybrid model developed has an AUC-ROC Score of 0.7971, which indicates that the model correctly ranks 79.71% of the positive samples higher than the negative samples, on average. Higher numbers on this scale, from 0 to 1, indicate that the model can differentiate across classes. While 0.7971 indicates practical classification abilities, scores above 0.5 generally indicate better performance.

4.2.2.3.2 Hybrid Model Evaluation Results (MSE and RMSE)

The hybrid model in this study yields a Mean Squared Error (MSE) of 0.5075 and a Root Mean Squared Error (RMSE) of 0.71239. The hybrid model's average difference between its predicted and actual values is measured by the Mean Squared Error (MSE); a lower MSE denotes more excellent performance. Consequently, the hybrid model performs better with an MSE of 0.5075, a tiny value. RMSE offers an interpretable metric in the original data units since it is the square root of MSE. In this case, an RMSE of roughly 0.71239 denotes a reasonably strong match, meaning the model performs well.

4.2.2.3.3 Hybrid Model Confusion Matrix

The confusion matrix in Figure 4.22 shows a model combining Recurrent Neural Networks (RNN) and Deep Neural Networks (DNN). With proper labels on the vertical axis and predicted labels on the horizontal, it displays model performance over three classes (0, 1, and 2). Each cell in the matrix indicates the number of times the actual class matches the anticipated class. Correct classifications (true positives) are highlighted by the diagonal cells, which display 5480 cases correctly classified as Class 0, 9374 as Class 1 (the highest class among all classes), and 4404 as Class 2. Erroneous predictions of Class 0 (1880) or Class 2 (1443) and comparable mistakes in other classes are indicated by off-diagonal cells. The research highlights areas for improvement, particularly in lowering misclassifications for Classes 0 and 2, while also underscoring the model's good performance for Class 1.

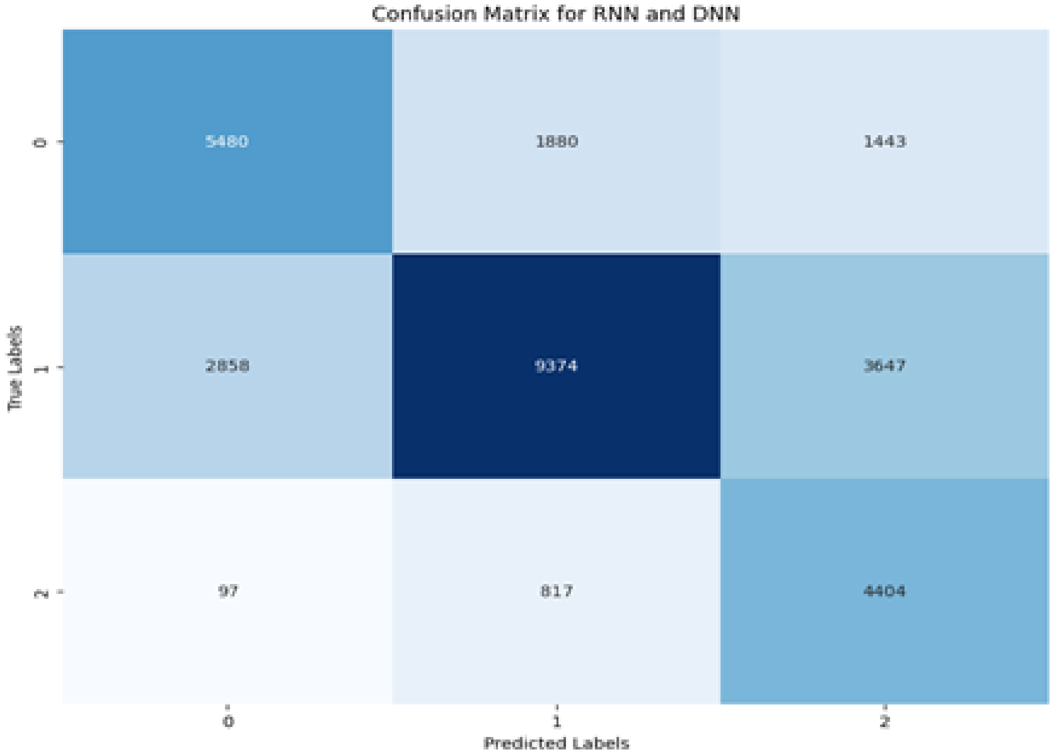
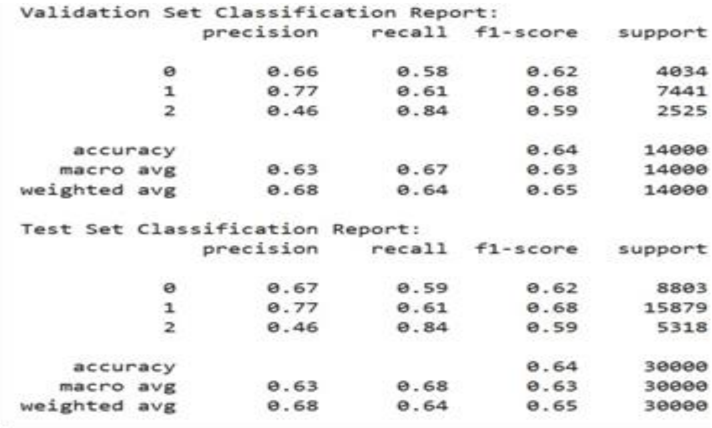


Figure 4. 22: Confusion matrix for a hybrid model

The model's performance was viewed in terms of sensitivity (True Positive Rate) and specificity (True Negative Rate). Sensitivity: [0.5773, 0.6093, 0.8372], Specificity: [0.87920, 0.7952, 0.7844] for classes 0,1 and 2 respectively. Sensitivity, the genuine positive rate, indicates how well the model detects positive examples. The third class has the highest sensitivity, at 83.72%, showing a strong ability to detect true positives in that group. Specificity, which measures the genuine negative rate and the capacity of the model to prevent false positives, performs admirably as well, as seen by values as high as 87.90% for the first class. These metrics show that the model performs exceptionally well at accurately detecting positives and negatives in various classes.

4.2.2.3.4 Hybrid Model Classification Report metrics

Figure 4.23 displays classification reports, thoroughly assessing the hybrid model's performance on test and validation sets. Precision measures, which reflect varied levels of predictive accuracy across different classes, have values of 0.66 for Class 0, 0.77 for Class 1, and 0.46 for Class 2 in the validation set. These metrics show the accuracy of optimistic predictions. Recall metrics assess the model's accuracy in recognizing instances of each class. In the validation set, scores for Class 0, Class 1, and Class 2 were 0.58, 0.61, and 0.84, respectively, suggesting good performance in detecting real cases, especially for Class 2. The F1 score offers additional information about the overall efficacy of the model by striking a balance between recall and precision.



```
Validation Set Classification Report:
              precision    recall  f1-score   support

           0       0.66      0.58      0.62      4034
           1       0.77      0.61      0.68      7441
           2       0.46      0.84      0.59      2525

    accuracy                           0.64     14000
   macro avg       0.63      0.67      0.63     14000
weighted avg       0.68      0.64      0.65     14000

Test Set Classification Report:
              precision    recall  f1-score   support

           0       0.67      0.59      0.62      8803
           1       0.77      0.61      0.68     15879
           2       0.46      0.84      0.59      5318

    accuracy                           0.64     30000
   macro avg       0.63      0.68      0.63     30000
weighted avg       0.68      0.64      0.65     30000
```

Figure 4. 23: Classification reports for a hybrid model

The validation and test sets have an overall accuracy of 64%. Macro and weighted averages provide aggregated performance views for all classes, highlighting areas of strength and areas where the model's predictive abilities can be improved.

4.2.3 Evaluation of the Models' Performance (RNN, DNN and hybrid model)

The researcher employed multiple evaluation indicators, including a confusion matrix, Results for Area under the curve (AUC), sensitivity and specificity, Mean Squared Error (MSE), and Root Mean Squared Error (RMSE), to evaluate the models' performance. Credit_Score has three categories: 0 represents poor, 1 indicates standard, and 2 signifies good credit score performance.

4.2.3.1 Confusion matrix for all models

Table 4. 1 Confusion matrix values for all models

Model	class	Correct	Misclassified as class 0	Misclassified as class 1	Misclassified as class 2
RNN+DNN	0	5480	-	1880	1443
	1	9374	727	-	1364
	2	4404	598	1593	-
RNN	0	5981	-	1768	1054
	1	8546	3323	-	4010
	2	4030	133	1155	-
DNN	0	6584	-	937	1282
	1	8742	3613	-	3524
	2	4357	182	779	-

Performance analysis in Table 4.1

DNN Model:

The DNN model performs well, particularly in Class 0, with 6584 valid classifications. Class 1 and Class 2 misclassifications are marginally higher than in the combined RNN + DNN model, although they are still lower than in the RNN model. The DNN model works effectively and significantly when decreasing Classes 1 and 2 misclassifications.

RNN Model:

The RNN model exhibits excellent misclassification rates, particularly for Classes 1 and 2, but correctly identifies a sizable portion of occurrences in each class. Among the three models, Class 0 has the most excellent correct classification (5981), suggesting that the RNN performs well in this class. Class 1 and Class 2 have noticeably high misclassification rates, indicating a need for improvement.

HYBRID Model: RNN + DNN

The performance of the combined RNN and DNN model is good, especially for Class 1, which has the highest number of accurate classifications (9374). Misclassifications are less than when using the RNN model alone, but they are still relatively evenly distributed between Classes 0 and 2. The combined model performs well by utilizing the advantages of both RNN and DNN. While the RNN model performs well in accurately classifying Class 0, its misclassification rates for Classes 1 and 2 are more excellent. Compared to the combined model, The DNN model performs better, especially in Class 0 and 1. However, it needs to be more accurate in Class 2 more frequently. With the fewest overall misclassifications, the RNN + DNN model provides a stable, balanced performance across all classes, making it the best choice among the three.

4.2.3.1 Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) for all models

The mean square error (MSE) measures the discrepancies between the predicted and actual results. MSE measures the average squared difference between the actual and predicted values. When the MSE is lower, the model's predictions are more accurate than the actual values. The target variable's square represents the MSE units. The MSE formula is given as:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Equation 4. 1:Mean Squared Error (MSE)

Where y_i are actual values, yi are predicted values, and n is the number of observations.

RMSE is the square root of the MSE. The standard deviation of the prediction errors is estimated via RMSE. Similar to MSE, a lower RMSE denotes improved model performance. On the other hand, because RMSE is expressed in the same units as the target variable, it is easier to link to the actual values.

$$\mathrm{RMSE} = \sqrt{\mathrm{MSE}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Equation 4. 2: Root Mean Squared Error (RMSE)

Table 4.2 compares the Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) for the DNN, RNN, and combined RNN+DNN models.

Table 4. 2 MSE-RMSE score

Model	**MAP**	**RMSE**	**Best performance**
DNN	1.349	1.162	No
RNN	1.253	1.119	No
RNN+DNN HYBRID	0.523	0.723	YES

The RNN+DNN combination model performs best with the lowest values for Mean Squared Error (MSE) and Root Mean Squared Error (RMSE).

4.2.3.2 Results for Area under the curve (AUC)

The performance of the three models employed in this research on the credit score data set is shown in Table 4.7 below. Based on the area under the ROC curve (AUC) values displayed, we can deduce that the RNN-DNN hybrid model outperformed the other two models.

Table 4. 3 AUC-ROC score

MODEL	**AUC-ROC SCORE**
RNN+DNN HYBRID	0.7971
RNN	0.7896
DNN	0.7504

Figure 4.24 shows a Receiver Operating Characteristic (ROC) curve, illustrates the trade-off between the True Positive Rate (TPR) and False Positive Rate (FPR) and is used to assess the performance of a binary classification model. The Y-axis and the false positive rate by the X-axis show the actual positive rate. The blue line represents an AUC of 0.79, indicating good validation data performance.

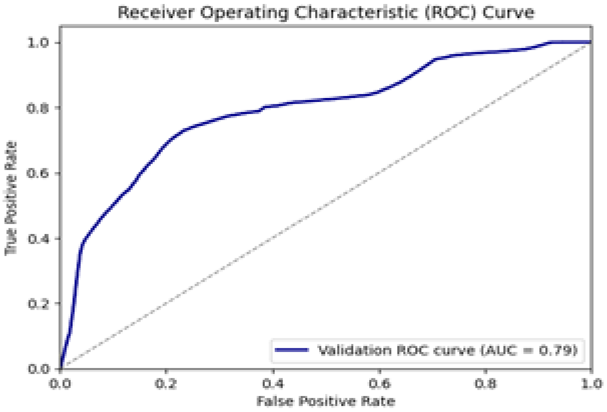


Figure 4. 24: RNN+DNN AUC

4.2.3.3 Sensitivity and Specificity of the models.

Sensitivity and specificity are statistical measures applied to assess a binary classification model performance, as seen in Table 4.8. They aid in evaluating how well a model distinguishes between positive and negative occurrences. Sensitivity measures the proportion of actual positives that the model correctly identifies.

$$\mathbf{Sensitivity} = \frac{\mathbf{True\ Positives}}{\mathbf{True\ Positives} + \mathbf{False\ Negatives}}$$

Formula 4. 1 Sensitivity

True Positives (TP): The number of correctly identified positive instances.

False Negatives (FN): The number of positive instances incorrectly identified as harmful.

High sensitivity indicates that the model is good at identifying positive instances, meaning it has a low rate of false negatives.

Specificity measures the proportion of actual negatives that the model correctly identifies.

$$\mathbf{Specificity} = \frac{\mathbf{True\ Negatives}}{\mathbf{True\ Negatives} + \mathbf{False\ Positives}}$$

Formula 4. 2 Specificity

True Negatives (TN): The number of correctly identified negative instances.

False Positives (FP): The number of negative instances incorrectly identified as positive.

High specificity indicates that the model is good at identifying negative instances, meaning it has a low rate of false positives.

Table 4.4 Comparison of performance of the models

Metric	DNN Class 0)	RNN (Class 0)	RNN + DNN (Class 0)	DNN (Class 1)	RNN (Class 1)	RNN + DNN (Class 1)	DNN (Class 2)	RNN (Class 2)	RNN + DNN (Class 2)
Sensitivity	0.7400	0.6582	0.5773	0.5300	0.5333	0.6093	0.8300	0.7529	0.8372
Specificity	0.9986	0.8351	0.8790	0.8935	0.7818	0.7952	0.7947	0.7907	0.7844

From Table 4.4, DNN shows high sensitivity in class 0 (0.7400) and class 2 (0.8300), with lower sensitivity in class 1 (0.5300). RNN + DNN shows moderate sensitivity overall, with the highest sensitivity in class 2 (0.8372). RNN shows the highest sensitivity in class 0 (0.6582) among the three models, moderate sensitivity in class 2 (0.7529), and lower sensitivity in class 1 (0.5333). DNN has the highest specificity in class 0 (0.9986) and class 1 (0.8935), with moderate specificity in class 2 (0.7947). RNN + DNN shows high specificity in class 0 (0.8790), with lower values in class 1 (0.7952) and class 2 (0.7844). RNN shows consistent specificity values across all classes, with the highest value in class 0 (0.8351). Based on the AUC-ROC score and the balance between sensitivity and specificity, **RNN + DNN** was considered the best performer overall.

4.2.3.4 Conclusion

The Hybrid Model (RNN+DNN) for credit score prediction consistently performs better in all areas, according to a thorough evaluation of the model's performance using a variety of metrics, including confusion matrix, Area Under the Curve (AUC-ROC), sensitivity, specificity, Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). According to the confusion matrix analysis, the RNN+DNN hybrid model maintains a balanced performance across all classes and achieves the most significant number of correct classifications in class 1. Sensitivity and specificity metrics demonstrate a solid balance in identifying both positive and negative examples, further confirming the robustness of the hybrid model. To be more precise, it retains Better specificity in class 0 (0.8790) and sensitivity in class 2 (0.8372).

The RNN+DNN hybrid model has the lowest MSE (0.523) and RMSE (0.723). It performs better overall than the standalone RNN (0.7896) and DNN (0.7504) models, as evidenced by its AUC-ROC score of 0. 7971. As a result, the RNN+DNN hybrid model was the top performer out of all the assessed models. It offers a reliable, accurate, and balanced classification across all criteria, making it the best option for predicting credit scores.

4.2.4 To validate the hybrid prediction model

The research's third objective was to validate the developed prediction model, which was accomplished by employing the cross-validation technique. Using a statistical approach called cross-validation, the data is split into K folds, or subsets, some of which are used for model training and others for validation. This method ensures the model is tested on several data subsets, improving the assessment's dependability.

This study evaluated the hybrid model that combines RNNs and DNNs using stratified K-fold cross-validation. With this approach, the dataset is divided into K subsets. The model is then trained repeatedly on each subset, and its performance is assessed at the end of each training cycle. Because stratified K-fold cross-validation guarantees that every fold of the dataset retains a comparable class distribution to the original dataset, it was expressly chosen for this purpose. This is crucial to ensure that underrepresented classes are fairly represented in the evaluation process, especially for datasets with uneven class composition.

The average performance throughout K iterations provided a trustworthy estimate of the model's performance. This method will evaluate the model's efficacy more accurately and help avoid overfitting. James et al. (2013) caution against using the widely used values of 10 and 5 for the cross-validation parameter, k, because of possible problems with excessive variance or bias. As a result, an ideal k value is found, guaranteeing a fair distribution of the data into k groups, each with an equal number of samples. For stratified k-fold cross-validation in this thesis, a k value of six was selected. The validation in this research was conducted on the developed hybrid deep learning model.

Implementing stratified k-fold cross-validation in this research requires three phases: Data Splitting, Training and Validation, and Results Aggregation.

4.2.4.1 Data splitting for Stratified k-fold cross-validation

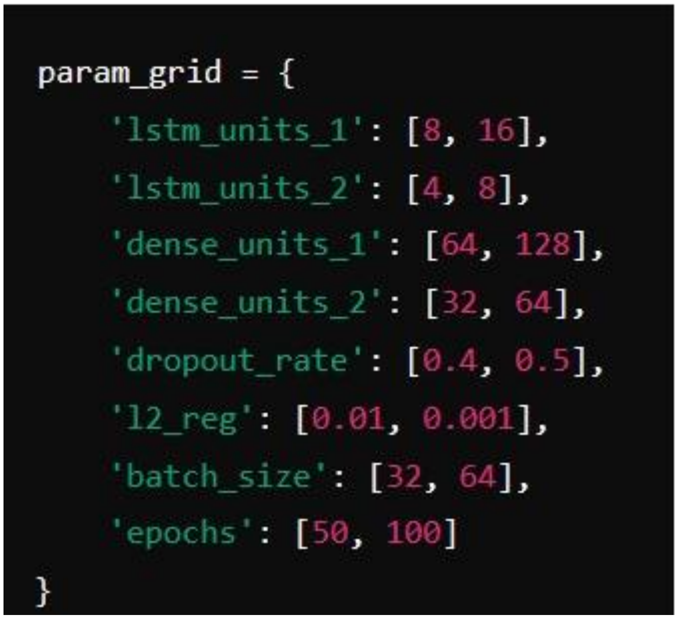
The data splitting stage of stratified k-fold cross-validation requires dividing the dataset into k-folds. The distribution of classes is kept throughout all folds since each fold preserves the same proportion of each class as the original dataset. For this research, the dataset was divided into six folds. The dataset used in this study had 100,000 entries. There are six stratified k-folds, and each should have roughly the same number of entries. By dividing the total number of entries by the number of folds, the entries in each fold were calculated: entries per fold = 100,000 / 6. This calculation results in Entries per fold of 16,666.67. Since the number of entries per fold must be an integer, the researcher distributed them as follows: folds will contain 16,667 entries each, and two folds will contain 16,666 entries each. This ensures that the total number of entries is exactly 100,000 while keeping the folds as balanced as possible.

4.2.4.2 Training and Validation of Stratified K-fold Cross-validation

The initial step in the training and validation process is data preparation, in which the target labels (targets) and input features (data) are first transformed into NumPy arrays. NumPy arrays offer a consistent data structure that makes it easier to do the following processing operations, including normalization, reshaping, and the use of neural networks in this study. The researcher ensures that the data is in an appropriate formatfor the subsequent actions by converting the data to NumPy arrays, laying a solid foundation for the model training and validation.

The research uses SMOTEENN to balance the training data and improve model generalization, as well as a nested loop with stratified K-Fold splits to optimize hyper parameters. The computation of class weight and data normalization using StandardScaler are essential stages in the LSTM model development process. Accurate and balanced learning is ensured via an LSTM model with early halting and extensive grid search for hyperparameter adjustment. These methods enhance the model's resilience, generalization, and performance on various datasets. This study uses StratifiedKFold from sci-kit-learn libraries to implement Stratified K-Fold cross-validation. It guarantees that the distribution of classes in each fold is similar, which is essential for imbalanced datasets and ensures that the distribution of classes inside each fold is accountable, which is necessary for imbalanced datasets to prevent biased evaluation.

A thorough grid search across a range of hyperparameter in this study aids in determining the model's ideal setup and improves performance. A parameter grid (param_grid), Figure 4.25, is defined to systematically explore various hyperparameter of the LSTM model used in this study. Options for the number of units in the dense layers, LSTM layers, batch sizes for training, L2 regularization values to regulate model complexity, dropout rates to prevent overfitting, and number of epochs for training time are all included in this grid. The purpose of the parameter grid in this research is to enable a comprehensive search to find the ideal combination that produces the best model performance by providing a range of values for each hyperparameter. The model was carefully explored to find a balance between overfitting and underfitting, which improved the model's accuracy and capacity for generalization.



```
param_grid = {
    'lstm_units_1': [8, 16],
    'lstm_units_2': [4, 8],
    'dense_units_1': [64, 128],
    'dense_units_2': [32, 64],
    'dropout_rate': [0.4, 0.5],
    'l2_reg': [0.01, 0.001],
    'batch_size': [32, 64],
    'epochs': [50, 100]
}
```

Figure 4. 25: Hyperparameter tuning Snippet code

The parameter grid in Figure 4.28 had eight parameters, the lstm_units_1 and lstm_units_2 layers, which use numbers 8 and 4, Dense_units_1 and dense_units_2 denote the 64 and 32 units, respectively, that are utilized in the first and second fully linked (dense) layers that come after the LSTM layers. To avoid overfitting, the dropout rate is evaluated at 0.4 or 0.5 by randomly changing a portion of the input units to zero during training. We investigate values of 0.01 and 0.001 for the L2 regularization parameter (l2_reg), which penalizes big weights to prevent overfitting. Batch_size, the number of samples per gradient update, is considered at 32 and 64. Epochs which indicate the number of times the complete dataset is passed through the neural network, are evaluated at 50 and 100.

The researcher took the following actions when using stratified K-Fold validation for hyperparameter tuning. Initially, the parameter grid was built using the information in param_grid, establishing various hyperparameter combinations for testing. In order to make sure that each fold has a comparable class distribution and that the evaluation metrics are representative and not influenced by an imbalanced class distribution, the researcher then sets up cross-validation using stratified K-Fold validation. Train the model using K-1(6-1) folds for each combination of hyperparameters in the param_grid, then validate it on the remaining fold. This process was repeated six times, using a different fold as the validation set to determine the average performance metrics for each combination of hyperparameters for all (6) K folds (e.g., accuracy, precision, recall, F1 score, AUC, etc.). In the end, the researcher chooses the set of hyperparameters that results in the optimal average performance metrics, as discussed in the following subsection, 4.3.4.3. By combining hyperparameter tuning with stratified K-Fold validation, the researcher makes sure that the model is evaluated effectively, considering class imbalance and choosing the best hyperparameters to enhance the model's performance.

4.2.4.3 Results Aggregation of Stratified K-fold cross-validation

Compiling summary statistics for every fold is a necessary step in aggregating results. Metrics like accuracy, recall, precision, specificity, sensitivity, F1-score, and area under the ROC curve (AUC) are calculated, along with their mean and standard deviation, to assess the overall performance and consistency of the model created in this study over the six folds. These metrics evaluate the model's dependability and show how well it can forecast data results that have yet to be observed. Performance measures are shown in Table 4.5 to depict performance and facilitate the analytical comparison of model performance. This technique helps identify the model's patterns, strengths, and potential weaknesses under various situations, which informs further refinement of the model or decision-making in practical applications.

Table 4. 5 Results of different hyperparameter combinations after stratified K-fold validation

LSTM Units 1	LSTM Units 2	Dense Units 1	Dense Units 2	Dropout Rate	L2 Reg	Batch Size	Epochs	Accuracy	Precision	Recall	F1 Score	Sensitivity	Specificity	AUC
8	4	64	32	0.4	0.001	32	50	0.6337	0.6346	0.6937	0.6306	0.675	0.8777	0.8036
8	4	64	32	0.4	0.001	32	100	0.6294	0.6318	0.6909	0.6265	0.6724	0.8762	0.8013
8	4	64	32	0.4	0.001	64	50	0.6269	0.6305	0.6896	0.6242	0.6628	0.8825	0.8006
8	4	64	32	0.4	0.001	64	100	0.6246	0.6283	0.6879	0.6223	0.6566	0.8817	0.7969
8	4	64	32	0.4	0.01	64	100	0.6202	0.6214	0.6803	0.6178	0.6581	0.8635	0.7848
8	4	64	32	0.4	0.01	32	100	0.614	0.6152	0.6744	0.6118	0.6493	0.8599	0.7807
8	4	64	32	0.4	0.01	32	50	0.6096	0.611	0.6694	0.6073	0.6457	0.8555	0.782
8	4	64	32	0.4	0.01	64	50	0.5889	0.6036	0.6632	0.5886	0.6132	0.8769	0.7775
8	4	64	32	0.5	0.01	32	50	0.5527	0.5697	0.6277	0.5529	0.578	0.8585	0.7606

Each parameter went through six iterations in the stratified k-fold validation, and the average result is recorded in Table 4.5. From the results, the combination of hyperparameters (8, 4, 64, 32, 0.4, 0.001, 32, 50) produced the best overall results, with an AUC of 0.8036 and an accuracy of 0. 6337. The performance measures are guaranteed to be accurate and unaffected by unequal class distributions because of the stratified K-Fold validation.

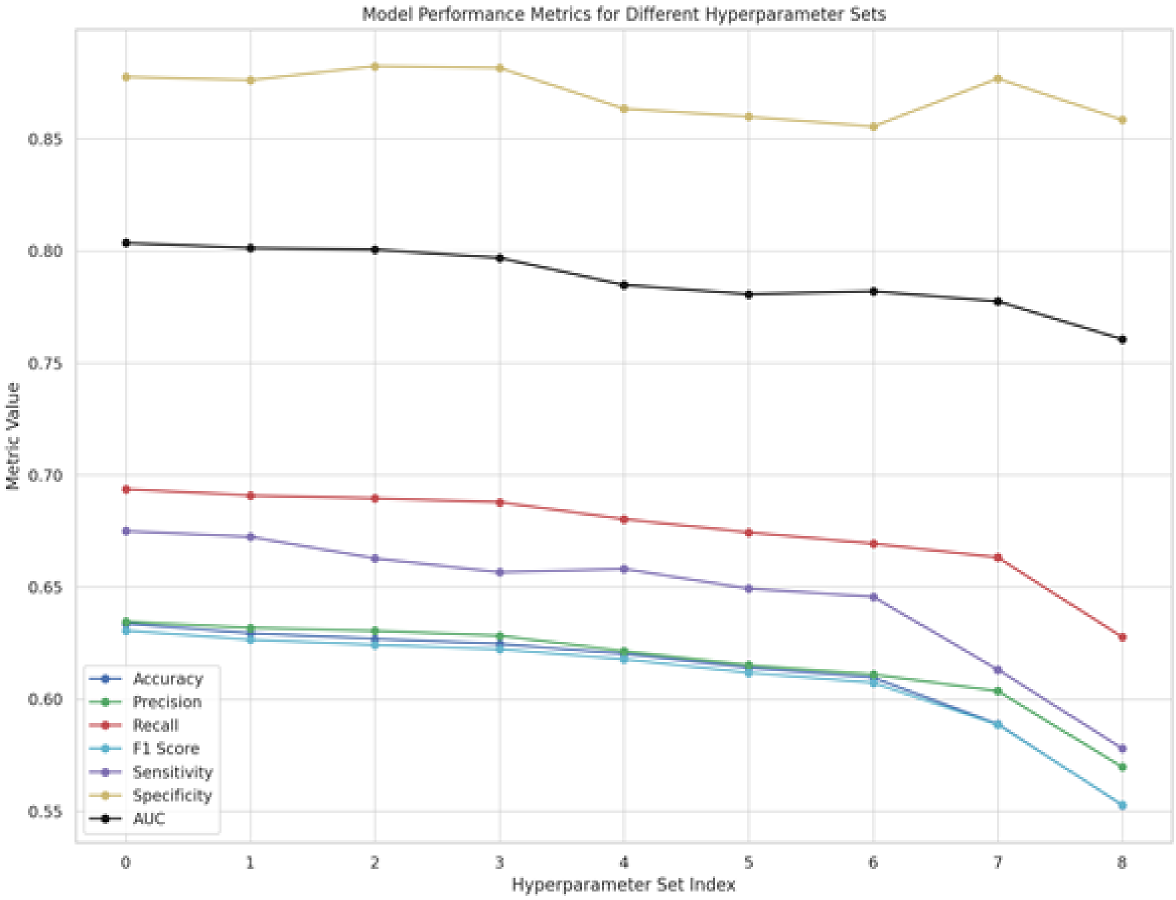


Figure 4. 26: Model performance for different hyperparameter sets

The outcomes demonstrate the significance of adjusting to produce the best results by demonstrating how various combinations of hyperparameters affect the model's performance, as visualized in Figure 4.26. The result is that the hybrid model performance is good, and the deep learning model is fit for real-life problems.

4.2.3 Credit Score Prediction Web Tool

In financial technology, risk reduction and well-informed lending decisions depend heavily on the precise evaluation of credit scores. A hybrid model that combines Recurrent Neural Networks (RNN) and Deep Neural Networks (DNN) has been developed to solve credit scoring classification problems and meet this urgent requirement. With the help of this advanced hybrid approach, which combines the strength of DNNs' feature extraction with the temporal analytic skills of RNNs, a reliable model that is Better at spotting trends and estimating creditworthiness from complicated financial data is developed. Using the Streamlit app framework in Python, a web tool has been developed to make this complex model user-friendly and accessible. It offers an interactive platform where users may interact with the model.

The Streamlit-based web tool offers a user-friendly interface designed to streamline the entire process of credit scoring analysis. User financial data is entered using a sidebar interface. Using sliders and checkboxes, they can enter values for Age, Annual Income, Monthly hand Salary, Interest Rate, Number of Loan, Delay from Due Date, Modified Credit Limit, Number of Credit Inquiries, Credit Mix, Outstanding Debt, Credit Utilization Ratio, Payment of Min Amount, Total Monthly EMI, and Payment Behavior. Assuming that the Credit_Score forecast is a probability array, the tool determines which index has the highest chance of correctly predicting the credit score category. It interprets this index, which is shown as the user's credit score, into understandable classifications "Good," "Standard," or "Poor" using an already-existing dictionary. Streamlit's effortless integration with hybrid RNN-DNN models improves credit scoring accuracy and efficiency while also making access to sophisticated machine learning techniques, which in turn leads to more responsible and fair lending practices.

4.2.4.1 Web-based tool development methodology

In this research, the researcher used Agile methodology to develop the web-based tool to display the output of the model developed in this research. Agile approaches improve adaptability, teamwork and response to modifications in Streamlit web development, which makes it an excellent option for This research's needs change over time. Streamlit's rapid prototyping and interactive features combine well with Agile's emphasis on iterative development, continual feedback, and adaptability.

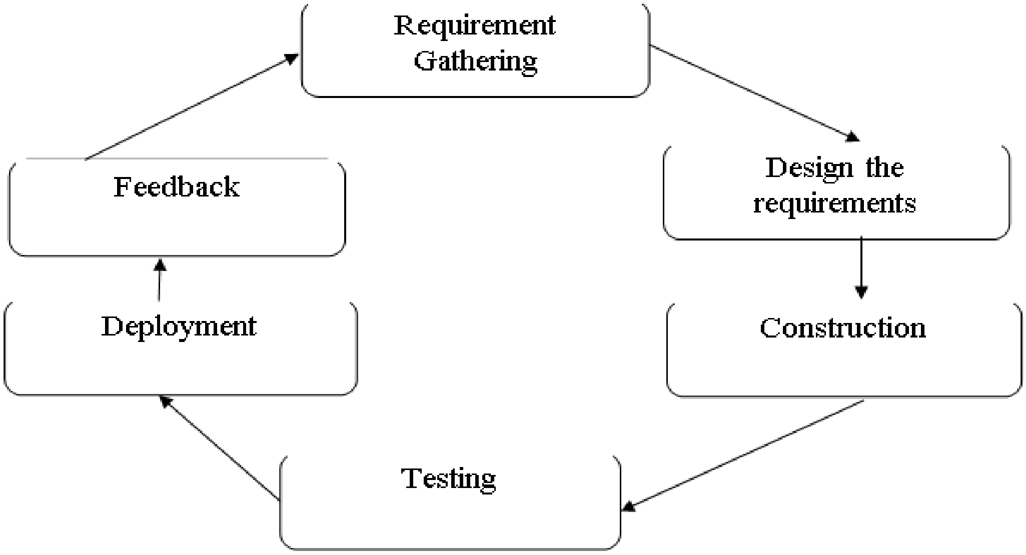


Figure 4. 27: Agile SDLC Model

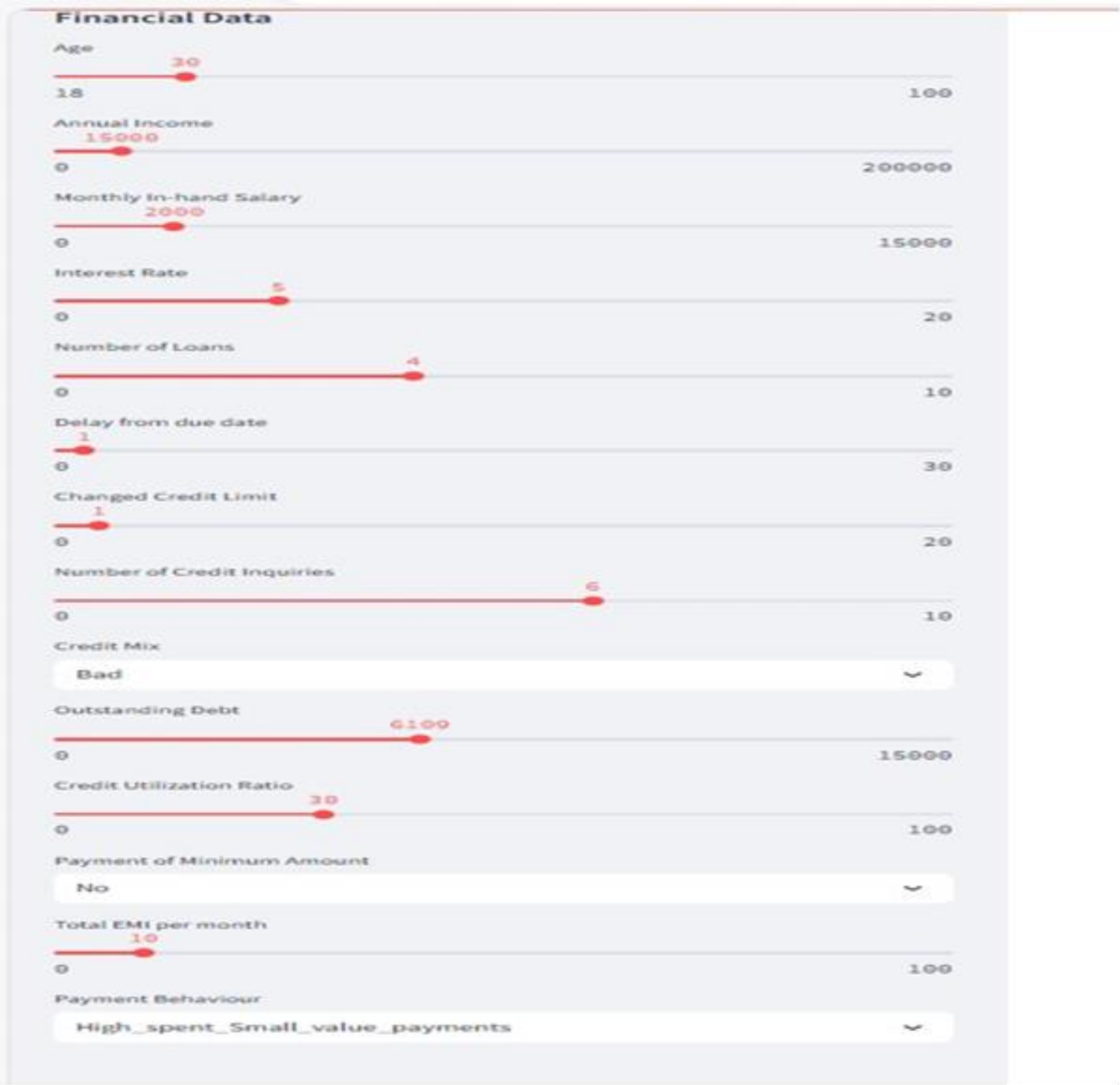
The Streamlit web tool for a hybrid RNN-DNN model in credit scoring was developed using an Agile methodology. Every process stage, from requirement gathering to feedback, was carried out collaboratively and iteratively.

The researcher used Keras's ability to show metrics and enable user input to develop a web-based user interface. At the same time, they created a simplified Streamlit interface specifically for presenting structured outputs and importing financial data. After ensuring it complied with performance requirements, they verified that the developed model saved as “your_model.h5” has portability and dependability attributes. Interactive components were then created for data processing and user input, and the model was smoothly incorporated into the Streamlit application. Based on the user's information, the program used the predict () function to produce precise credit score predictions. It was then put through a rigorous testing process to ensure that credit score predictions were accurate and that all user interface sliders operated as intended.

Using Streamlit, the researcher installed the web application so that users could enter data and get real-time credit score predictions. Before making the application available to stakeholders, they ensured it met accessibility guidelines and operational needs. The tool gave users instant feedback by displaying predicted credit score categories based on input data. It also included user feedback mechanisms to enhance the usability and functionality of the application continually.

4.2.4.2 Web Tool Interface

The web-based tool was used to display and visualize the hybrid credit scoring model developed in this research. There are 14 features to be inputted as they were in the developed model as seen in Figure 4.28



Financial Data

Age: 30 (18–100)

Annual Income: 15000 (0–200000)

Monthly In-hand Salary: 2000 (0–15000)

Interest Rate: 5 (0–20)

Number of Loans: 4 (0–10)

Delay from due date: 1 (0–30)

Changed Credit Limit: 1 (0–20)

Number of Credit Inquiries: 6 (0–10)

Credit Mix: Bad

Outstanding Debt: 6100 (0–15000)

Credit Utilization Ratio: 30 (0–100)

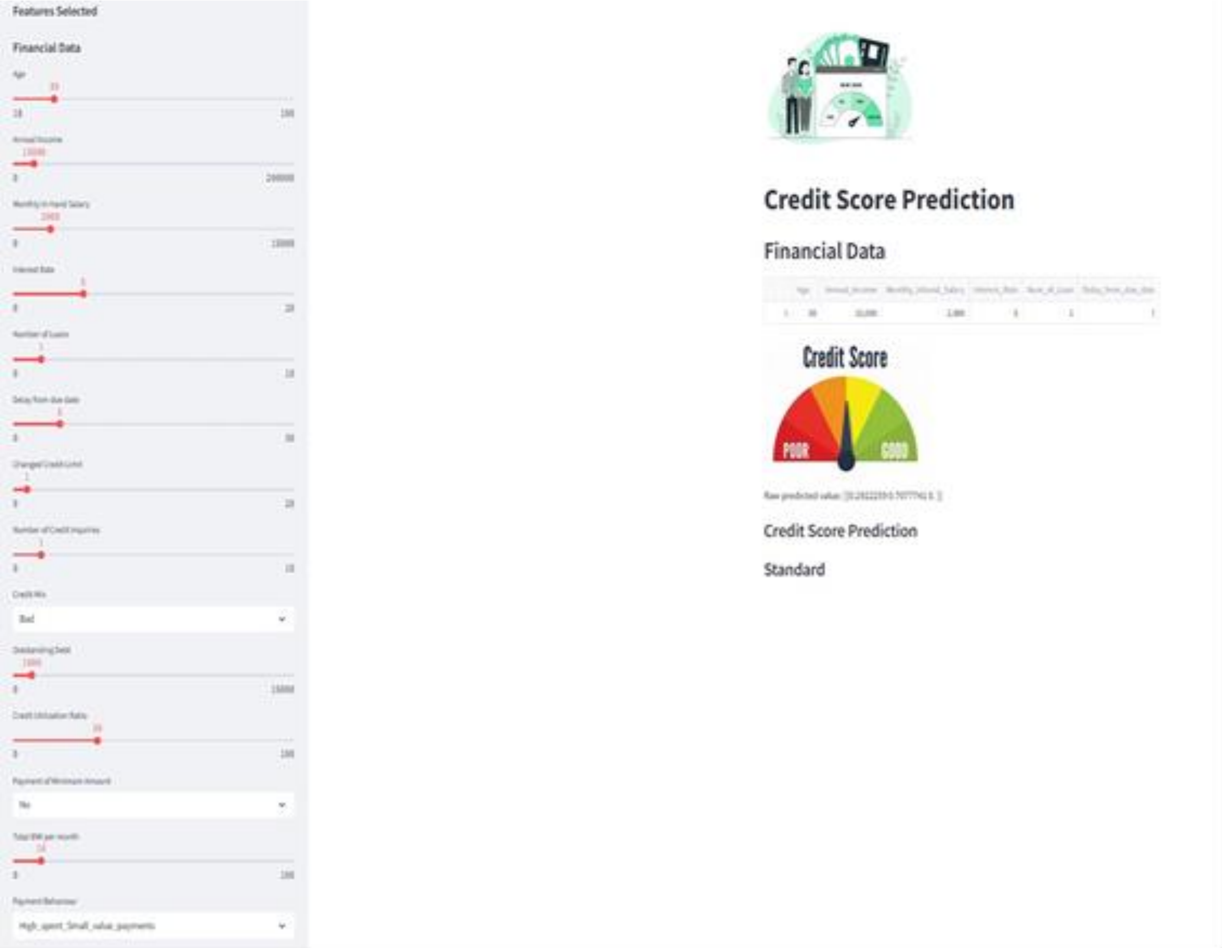
Payment of Minimum Amount: No

Total EMI per month: 10 (0–100)

Payment Behaviour: High_spent_Small_value_payments

Figure 4. 28: Features Input Menu

The web tool interface has three major areas. First area is the left slide bars where the users input the financial data by simply sliding to the value to be inputted second area is where the inputted data is displayed in a table format and lastly the third area is where the credit score predictions are displayed as displayed in Figure 4.29.



Features Selected

Financial Data

Age: 33 (18 – 100)

Annual Income: 15000 (0 – 200000)

Monthly In-hand Salary: 2000 (0 – 15000)

Interest Rate: 5 (0 – 20)

Number of Loans: 1 (0 – 10)

Delay from due date: 5 (0 – 30)

Changed Credit Limit: 1 (0 – 20)

Number of Credit Inquiries: 1 (0 – 10)

Credit Mix: Bad

Outstanding Debt: 1000 (0 – 15000)

Credit Utilization Ratio: 35 (0 – 100)

Payment of Minimum Amount: No

Total EMI per month: 14 (0 – 100)

Payment Behaviour: High_spent_Small_value_payments

Credit Score Prediction

Financial Data

	Age	Annual_Income	Monthly_Inhand_Salary	Interest_Rate	Num_of_Loan	Delay_from_due_date
1	30	15,000	2,000	5	1	7

Raw predicted value: [0.28222593 0.70777407 0.]

Credit Score Prediction

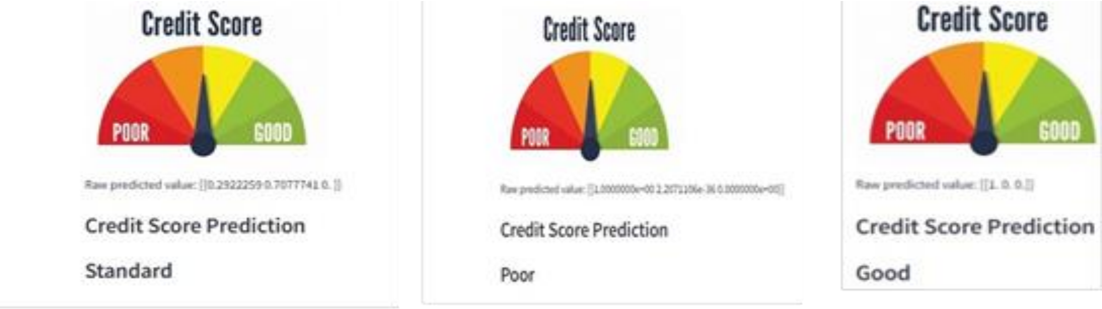
Standard

Figure 4. 29: Web app view

Using Kera’s' capabilities, the model was saved to disk in the your_model.h5 format after its performance satisfies the required standards. In addition to storing the trained weights, this format makes use of HDF5 maintains the architecture of the model, guaranteeing portability and making it simple to reload for forecasts or deployment in a credit score prediction tool.

The pre-trained model (your_model.h5) is loaded into the Streamlit web application, and then the tool is made to interactively gather user input. Through a sidebar interface, users enter their financial data. They can specify variables like age, salary, loan information, credit behavior, and more using sliders and choose boxes as in Figure 4.39.

Using the components of Streamlit, this input data is dynamically recorded and formatted into a structured DataFrame. The user-provided data is then shown by the application for perusal. The tool proceeds to employ the loaded model to forecast using the user data that has been gathered. To predict a credit score, it uses the models predict () function, showing the user the raw predicted number. The tool finds the index with the highest likelihood to predict the credit score category, assuming that the Credit_Score prediction is a probability array. Using an existing dictionary, it maps this index to meaningful categories such as 'Good,' 'Standard,' or 'Poor'(Figure 4.33).



```
Raw predicted value: [[0.2922259 0.7077741 0. ]]
```

Credit Score Prediction

Standard

```
Raw predicted value: [[1.0000000e+00 2.2071106e-36 0.0000000e+00]]
```

Credit Score Prediction

Poor

```
Raw predicted value: [[1. 0. 0.]]
```

Credit Score Prediction

Good

Figure 4. 30:Web Tool output

Lastly, Streamlit gives the user a quick evaluation based on their submitted data by displaying the anticipated credit score category. This interactive method makes use of Streamlit's features to provide a smooth and simple user interface for credit score prediction.

4.3. 5 Conclusion

Recurrent neural networks (RNNs) and deep neural networks (DNNs) are combined in a hybrid deep learning model that the study successfully designed and verified for credit score predictive modeling. By applying stratified K-fold cross-validation, the study made sure that all of the dataset's subsets were thoroughly evaluated, which helped to address concerns with class imbalance. This method produced reliable evaluations of the model's effectiveness in addition to a notable improvement in its performance.

Important results showed that an accuracy of 63.37% and an AUC of 0.8036 were reached by the optimum model configuration, which included 8 LSTM units spread across two layers, dense layers with 64 and 32 units, a dropout rate of 0.4, L2 regularization of 0.001, batch size of 32, and 50 epochs. Stratified K-fold cross-validation's rigorous methodology ensured objective performance measures, which strengthened the findings' validity and relevance for a range of real-world prediction tasks like credit score. Future directions in research could continue to increase predictive modeling in real-world applications across a variety of disciplines by enhancing model capabilities through different architectures and preprocessing techniques.

The researcher created an interactive web interface (Figure 4.29) that lets users examine measurements derived from the model's predictions. Users could enter data to generate forecasts in real time, making it easier for stakeholders to comprehend and use the results of the model. The model, saved as your_model.h5, kept weights and architecture consistent for simple deployment, and Streamlit's interface made it easy to input data and show it dynamically. Using Streamlit's ability to forecast and classify credit scores, this method gave consumers instant evaluations based on their financial inputs.

4.2.4.3 Web based tool testing

In this thesis, the credit web-based tool was thoroughly tested using a combination of automated and manual methods to ensure its reliability and effectiveness. Automated testing was conducted to verify the functionality of individual components through unit tests and to assess the interactions between different modules with integration tests. End-to-end tests simulated complete user journeys to ensure the tool worked seamlessly from start to finish. Performance testing, including load and stress tests, was carried out to evaluate the tool's ability to handle high traffic and operate under extreme conditions. Security was tested through vulnerability scans and penetration testing, which identified and mitigated potential threats. Usability was assessed using heuristic evaluations and cognitive walkthroughs to identify and resolve any interface issues. Cross-browser and cross-device testing ensured the tool's compatibility across various platforms, while code reviews and static analysis helped maintain code quality and preemptively address issues. Finally, simulated user environments were created to emulate real-world conditions, allowing for comprehensive testing without involving actual users.

CHAPTER FIVE: CONCLUSION, RECOMMENDATIONS AND FUTURE WORK

5.1 Introduction

In this chapter, conclusions are drawn from the previous chapter's discussion of the study's findings. It emphasizes the study's contribution to knowledge and information technology, as well as some of its drawbacks and challenges. It also offers suggestions for further research and work in the future

5.2 Conclusions

This research was guide by four objectives, which were achieved in chapter four. The study successfully examined how traditional and behavioral data integrate into credit scoring models, using rigorous feature selection and data cleaning procedures to improve the interpretability and performance of the models. Important characteristics that were found to be critical in determining creditworthiness included interest rates, credit utilization ratios, and outstanding debt. This emphasizes the need of combining behavioral and traditional financial measures in predictive modeling. The study sought to improve lending decisions and risk management tactics by offering thorough insights into borrower risk profiles through a methodical analysis of these variables.

According to the study's second objective, a hybrid deep learning model that predicts credit scores by smoothly integrating traditional and behavioral data was created. Recurrent neural networks (RNNs) and deep neural networks (DNNs) were combined to create a hybrid model that outperformed independent models on a variety of evaluation parameters. Its strong classification capabilities were highlighted by its notable a sensitivity of 0.8372 and better specificity of 0.8790. The hybrid model demonstrated effectiveness in providing precise and well-balanced credit score predictions appropriate for real-world use, with the lowest RMSE (0.723) and MSE (0.523) of all the models examined.

Then third objective of the study was validating the developed model and the researcher employed Stratified K-fold cross-validation to thoroughly validate the created model, guaranteeing comprehensive analysis and addressing concerns regarding class imbalance. The model's performance consistency and dependability were confirmed by this systematic technique, which is essential for using the model in practical situations. AUC (0.8036) and accuracy (63.37%) of the optimized model configuration which included 8 LSTM units, dense layers with specified units, dropout rate, L2 regularization, batch size, and epochs were significantly improved, further confirming the model's effectiveness in credit scoring tasks.

The final objective of the research was to develop a web-based visualization tool to make it easier to explore and analyze the outcomes of models. By allowing stakeholders to interactively assess credit score forecasts and comprehend the underlying causes impacting decisions, this technology improves accessibility and transparency. In order to further improve predictive modeling capabilities across a variety of domains and applications, future research goals include investigating alternate neural network topologies and sophisticated preprocessing techniques. Through constant methodology refinement and utilization of state-of-the-art technologies, the study hopes to further the continuing progress in credit scoring and related predictive analytics.

5.3 Contributions of the study

The research makes a substantial contribution to predictive modeling by utilizing hybrid deep learning models that combine Recurrent Neural Networks (RNNs) and Deep Neural Networks (DNNs) in the field of credit score prediction. By combining DNNs' capacity to recognize intricate patterns with RNNs' ability to handle sequential data, this integration increases the sophistication of credit scoring systems. The hybrid RNN+DNN model improves the field's ability to assess creditworthiness by improving predicted accuracy and robustness through the effective modeling of hidden linkages within credit data.

In addition, the study used a wide range of metrics, such as confusion matrices, AUC-ROC scores, sensitivity, specificity, MSE, and RMSE, to thoroughly assess the hybrid model's performance. The hybrid RNN+DNN model consistently performs better than the standalone RNN and DNN models across a range of evaluation criteria, according to the results. The model's accuracy and balance in predicting credit risk across various risk profiles are highlighted by this performance validation, which is important for financial decision-making.

Using stratified K-fold cross-validation as a technique, the study strengthens the validity of its conclusions. This technique reduces biases resulting from class imbalances in the dataset while ensuring comprehensive evaluation across a variety of data subsets. The strict technique supports the hybrid model's application in actual contexts like financial institutions and credit evaluation procedures, in addition to validating its effectiveness. In order to continuously improve predictive accuracy and reliability in credit scoring methodologies, the study's findings open up new research avenues that will focus on further refining hybrid deep learning architectures, investigating new data features, and incorporating complementary deep learning techniques.To increase the usefulness of the research, a web-based interface was created to show a hybrid credit scoring model. It has an easy-to-use interface that is divided into three sections: a centre table for input display, a sidebar with sliders for entering financial data, and a prediction area for credit ratings. The model was saved in.h5 format using Kera's to ensure mobility and preserve architecture. Data such as age and income are entered by users and processed into a DataFrame for forecasting. Using Streamlit's capabilities, predicted credit scores ('Good,' 'Standard,' or 'Poor') are displayed based on user inputs for a seamless experience. This tool tests the efficacy of the proposed model in real-world settings and shows how it may be used in practice.

5.4 Recommendations

Following the specific objectives of the study, a number of focused actions have been recommended to enhance hybrid deep learning models in credit score prediction. In order to fully understand the significance of traditional and behavioral data, it is imperative that greater emphasis be placed on gathering more extensive and varied datasets that fairly represent a wide range of socioeconomic backgrounds and demographic categories. This will improve the hybrid RNN+DNN models' performance and generalizability in real-world scenarios. To lessen model sensitivity and guarantee consistent performance across many contexts, it is advised that resilient hyper parameter optimization strategies, such as automated machine learning frameworks or Statistical optimization, be explored during the model-development process. Robust techniques such as stratified K-fold cross-validation should be used to validate the created model, with a focus on external validation with real credit scoring systems or distinct datasets to validate the model's performance in different scenarios. Furthermore, it's critical to improve interpretability when creating a web-based tool for visualizing the model by utilizing strategies like layer-wise relevance propagation and attention mechanisms. These strategies can offer insightful information about model decisions and increase transparency in credit scoring procedures. To advance the scalability, interpretability, ethical considerations, and reliability of hybrid deep learning models in credit scoring, key factors such as data scientists, machine learning engineers, financial institutions, software developers, and academic researchers should implement these recommendations in a collaborative manner.

5.5 Future Research

By focusing on a few key areas, future research should advance the development of hybrid deep learning models in credit score prediction. Beyond the obvious suggestions, further research should stress the significance of external validation using different datasets or real credit scoring systems, as this would validate the model's performance in a greater number of scenarios and boost confidence in its usefulness. Predictive accuracy and dependability may also continue to increase as new neural network topologies and sophisticated preprocessing methods are investigated. The effectiveness and moral rectitude of hybrid deep learning models in financial decision-making and other crucial applications can be greatly improved by following these research directions

5.6 Summary

This chapter summarizes the research findings, which highlights the study's important contributions to credit scoring through the creation of a hybrid RNN+DNN model.

Validated through rigorous cross-validation approaches, the model is exhibiting higher predictive performance, integrating both traditional and behavioral data. The development of a web-based visualization tool that improves credit scoring's accessibility and transparency is also covered in this chapter. In addition, suggestions for further study are being made, with an emphasis on obtaining a wider variety of datasets, refining hyper parameters, enhancing the interpretability of the model, and guaranteeing ethical concerns in predictive modeling.

REFERENCE

Abraham, M. T., Satyam, N., Lokesh, R., Pradhan, B., & Alamri, A. (2021). Factors affecting landslide susceptibility mapping: Assessing the influence of different machine learning approaches, sampling strategies and data splitting. Land, 10(9), 989.

Alessio Balduini, Douglas Dwyer, Sara Gian Freda, Reeta Hemminki, Lucia Yang &Janet Yinqing Zhao (2017) Combining Financial and Behavioral Information to Predict Defaults (moodysanalytics.com)

Alessio Balduini, S. G. (2017, 9). Combining Financial and Behavioral Information to Predict Defaults for Small and Medium-Sized Enterprises: A Dynamic Weighting Approach. Retrieved from https://www.moodysanalytics.com/articles/2017/combining-financial-and-behavioral-information

Ampountolas, A., Nyarko Nde, T., Date, P., & Constantinescu, C. (2021). A machine learning approach for micro-credit scoring. Risks, 9(3), 50

Aniceto, M. C., Barboza, F., & Kimura, H. (2020). Machine learning predictivity applied to Hamberg & Bouvin, & Bouvin, consumer creditworthiness. Future Business Journal, 6(1), 1-14.

Balduini, A. (2017). https://www.moodysanalytics.com. Retrieved from Combining Financial and Behavioral Information to Predict Defaults for Small and Medium-Sized Enterprises: https://www.moodysanalytics.com/articles/2017/combining-financial-and-behavioral-information#:

Barbon, M. F. (2019,7). Technical_Guide_CreditScore.pdf. https://www.cgap.org: https://www.cgap.org/sites/default/files/publications/2019_07_Technical_Guide_CreditScore.pdf

Benavides, E., Fuertes, W., Sanchez, S., & Sanchez, M. (2020). Classification of phishing attack solutions by employing deep learning techniques: A systematic literature review. Developments and Advances in Defense and Security: Proceedings of MICRADS 2019, 51-64

Bhat, A. (2023). Retrieved from Research Design: What it is, Elements & Types: https://www.questionpro.com/blog/research-design/

Bhatia, S., Sharma, P., Burman, R., Hazari, S., & Hande, R. (2017). Credit scoring using machine learning techniques. International Journal of Computer Applications, 161(11), 1-4.

Bhandari, P. (2023, 6 22). https://www.scribbr.com. Retrieved from Operationalization | A Guide with Examples, Pros & Cons: https://www.scribbr.com/methodology/operationalization/

Braight Technologies. (2023, 8 3). *What is Online Behavioral Data and How's it Used for Credit Scoring?* Retrieved from https://www.linkedin.com/pulse/what-online-behavioral-data-hows-used-credit-scoring

Carstensen, A. K., & Bernhard, J. (2019). Design science research–a powerful tool for improving methods in engineering education research. *European Journal of Engineering Education*, *44*(1-2), 85-102.

Chi, G., Uddin, M. S., Abedin, M. Z., & Yuan, K. (2019). Hybrid model for credit risk prediction: An application of neural network approaches. International Journal on Artificial Intelligence Tools, 28(05), 1950017.

Chopra, S. (2020). Current Regulatory Challenges in Consumer Credit Scoring Using Alternative Data

Clare Liu. (2022, 9 20). *More Performance Evaluation Metrics for Classification Problems YouShould Know*. Retrieved from https://www.kdnuggets.com/2020/04/performance-evaluation-metrics-classification

Driven Methodologies. Vand. J. Ent. & Tech. L., 23, 625.

Creditinfo.com. (2023, 04 26). https://chronicle.creditinfo.com/2023/04/26. Retrieved from the-role-of-artificial-intelligence-and-machine-learning-in-credit-scoring:

Dastile, X., Celik, T., & Potsane, M. (2020). Statistical and machine learning models in credit scoring: A systematic literature survey. Applied Soft Computing, 91, 106263.

Delua, J. (2021, March 12). Supervised vs. Unsupervised Learning: What's the Difference? IBM – United States. https://www.ibm.com/blog/supervised-vs-unsupervised-learning/

DeNicola, L. (2023, 5 22). credit Karma. Retrieved from What factors affect your credit scores? https://www.creditkarma.com/advice/i/what-affects-your-credit-scores

Dharwadkar, N. V., & Patil, P. S. (2018). Customer retention and credit risk analysis using ANN, SVM and DNN. International Journal of Society Systems Science, 10(4), 316-332.

Duan, J. (2019). Financial system modeling using deep neural networks (DNNs) for effective risk assessment and prediction. Journal of the Franklin Institute, 356(8), 4716-4731.

Erik Hamberg& Daniel Bouvin, (2022) Credit Scoring Based on Behavioral Data

FICO. (2020, August). FICO. Retrieved from using-alternative-data-credit-risk-modelling: https://www.fico.com/blogs/using-alternative-data-credit-risk-modelling#:

FICO. (2023). https://www.myfico.com. Retrieved from Myfico.com: https://www.myfico.com/credit-education/credit-scores/payment-history

Fintech. (2023) https://www.financemagnates.com/fintech/data/use-cases-of-alternative-data-sources-for-credit-scoring-and-risk-management/. Retrieved from use-cases-of-alternative-data-sources-for-credit-scoring-and-risk-management:

Hussain, A., Khan, M., Rehman, S. U., & Khattak, A. (2019). Credit scoring model for retail banking sector in Pakistan. Journal of Managerial Sciences, 14(4), 153-161. Hybrid deep-learning and machine-learning models for predicting COVID-19. Computational Intelligence and Neuroscience, 2021

IBM Data and AI Team. (2023, July 6). AI vs. machine learning vs. deep learning vs. neural networks: What's the difference? IBM - United States. https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks

IBM. (2021, 8 17). Retrieved from CRISP-DM Help Overview: https://www.ibm.com/docs/en/spss-modeler/saas?topic=dm-crisp-help-overview

Illion. (2022, 5 9). *Banking transaction data a useful credit assessment tool*. Retrieved from https://www.bankingday.com/login?p=%2fbanking-transaction-data-useful-credit-assessment-tool

J. Xiao and Z. Zhou, "Research Progress of RNN Language Model," 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA),Dalian,China,2020,pp.1285-1288doi: 0.1109/ICAICA50127.2020.9182390. keywords:{Computational modeling;Recurrent neural networks;Data models;Mathematical model;Natural language processing;Analytical models;language mode;recurrent neural network (RNN);research progress},

Johnson, D. (2023, 7 29). https://www.guru99.com. Retrieved from Supervised Machine learning: What Algorithms with Examples: https://www.guru99.com/supervised-machine-learning.html

Karagiannakos, S. (2020, February 26). Deep learning algorithms - The complete guide. AI Summer.https://theaisummer.com/Deep-Learning-Algorithms/

Kumar, A., Shanthi, D., & Bhattacharya, P. (2021, August). Credit Score Prediction System using deep learning and K-means algorithms. In Journal of Physics: Conference Series (Vol. 1998, No. 1, p. 012027). IOP Publishing.

Liu, J., Zhang, S., & Fan, H. (2022). A two-stage hybrid credit risk prediction model based on XGBoost and graph-based deep neural network. Expert Systems with Applications, 195, 116624.

Luo, C. (2020). A comprehensive decision support approach for credit scoring. Industrial

Management & Data Systems, 120(2), 280-290

Machado, M. R., & Karray, S. (2022). Assessing credit risk of commercial customers using hybrid machine learning algorithms. Expert Systems with Applications, 200, 116889.

Munguti, V. M., & Ngali, R. M. (2020). Evaluating credit worthiness of small and growing technology businesses. The University Journal, 2(1), XX-XX.

Mhina, M. C., & Labeau, F. (2021, May). Using Machine Learning Algorithms to Create a Credit Scoring Model for mobile money users. In 2021 IEEE 15th International Symposium on Applied Computational Intelligence and Informatics (SACI) (pp. 000079-000084). IEEE.

Nalić, J., Martinović, G., & Žagar, D. (2020). New hybrid data mining model for credit scoring based on feature selection algorithm and ensemble classifiers. Advanced Engineering Informatics, 45, 101130.

Nasir, V., & Sassani, F. (2021). A review on deep learning in machining and tool monitoring: Methods,opportunities, and challenges. *The International Journal of Advanced Manufacturing Technology*, *115*(9-10), 2683-2709.

Offermann, P., Levina, O., Schönherr, M., & Bub, U. (2019, May). Outline of a design science research process. In *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology* (pp. 1-11).

Pello, R. (2018, 10 31). Retrieved from Design science research:

https://medium.com/@pello/design-science-research-a-summary-bb538a40f669

Qaid, T. S., Mazaar, H., Al-Shamri, M. Y. H., Alqahtani, M. S., Raweh, A. A., & Alakwaa, W. (2021). Hybrid deep-learning and machine-learning models for predicting COVID-19.Computational Intelligenceand Neuroscience, 2021

Roy, P. K., & Shaw, K. (2023). A credit scoring model for SMEs using AHP and TOPSIS. International Journal of Finance & Economics, 28(1), 372-391.

Satyam Kumar, A. Joshi. (2023, 7 5). Key Factors That Influence Credit Score. Retrieved from www.forbes.com: https://www.forbes.com/advisor/in/credit-score/factors-that-affect-credit-score/

Sayjadah, Y., Hashem, I. A. T., Alotaibi, F., & Kasmiran, K. A. (2018, October). Credit card default prediction using machine learning techniques. In 2018 Fourth International Conference on Advances in Computing, Communication & Automation (ICACCA) (pp. 1-4). IEEE.

Selvaganapathy, S., Nivaashini, M., & Natarajan, H. (2018). Deep belief network-based detection and categorization of malicious URLs. Information Security Journal: A Global Perspective, 27(3), 145-161.

Sharma, G. (2021, May 27). Regression algorithms | 5 regression algorithms you should know. Analytics vidhya. Regression Algorithms | 5 Regression Algorithms you should know (analyticsvidhya.com)

Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D: Nonlinear Phenomena, 404, 132306.

Singh, A. (2023, july 14). Retrieved from A Comprehensive Guide to Ensemble Learning (with Python codes): https://www.analyticsvidhya.com/blog/2018/06/comprehensive-guide-for-ensemble- models

Standard Chartered Bank. (2022). https://www.sc.com. Retrieved from role-and-importance-of-credit-score: https://www.sc.com/in/stories/role-and-importance-of-credit-score

Sujaini. (2023, 5 19). Clear Tax. Retrieved from cleartext. in: https://cleartax.in/g/terms/credit-scoring

Sujaini. (2023,816). cleartext. Retrieved from judgmental-credit-analysis: https://cleartax.in/g/terms /Judgmental-credit-analysis

Sum, R. M., Ismail, W., Abdullah, Z. H., & Shah, N. F. M. N. (2022). A New Efficient Credit Scoring Model For Personal Loan Using Data Mining Technique for Sustainability Management. Journal of Sustainability Science and Management.

Suthanthiradevi, P., Srividhyasaradha, K., & Karthika, S. (2021). Modeling a Behavioral scoring system for lending loans using Twitter. In ITM Web of Conferences (Vol. 37, p. 01012). EDP Sciences.

Tobback, E., & Martens, D. (2019). Retail credit scoring using fine-grained payment data. Journal of the Royal Statistical Society Series A: Statistics in Society, 182(4), 1227-1246.

Tyagi, S. (2022). Analyzing Machine Learning Models for Credit Scoring with Explainable AI and Optimizing Investment Decisions. arXiv preprint arXiv:2209.09362.

Uthayakumar, J., Metawa, N., Shankar, K., & Lakshmanaprabu, S. K. (2020). Intelligent hybrid model for financial crisis prediction using machine

learning techniques. Information Systems and e-Business Management, 18, 617- 645.

Vom Brocke, J., Hevner, A., & Maedche, A. (2020). Introduction to design science research. *Design science research. Cases*, 1-13.

Worldbank. (2019). Credit scoring approaches guidelines final web. retrieved from https://thedocs.worldbank.org:https://thedocs.worldbank.org/en/doc/935891585869698451-0130022020/original/creditscoringapproachesguidelinesfinalweb.pdf

Wu, Y., & Pan, Y. (2021). Application analysis of credit scoring of financial institutions based on machine learning model. Complexity, 2021, 1-12.

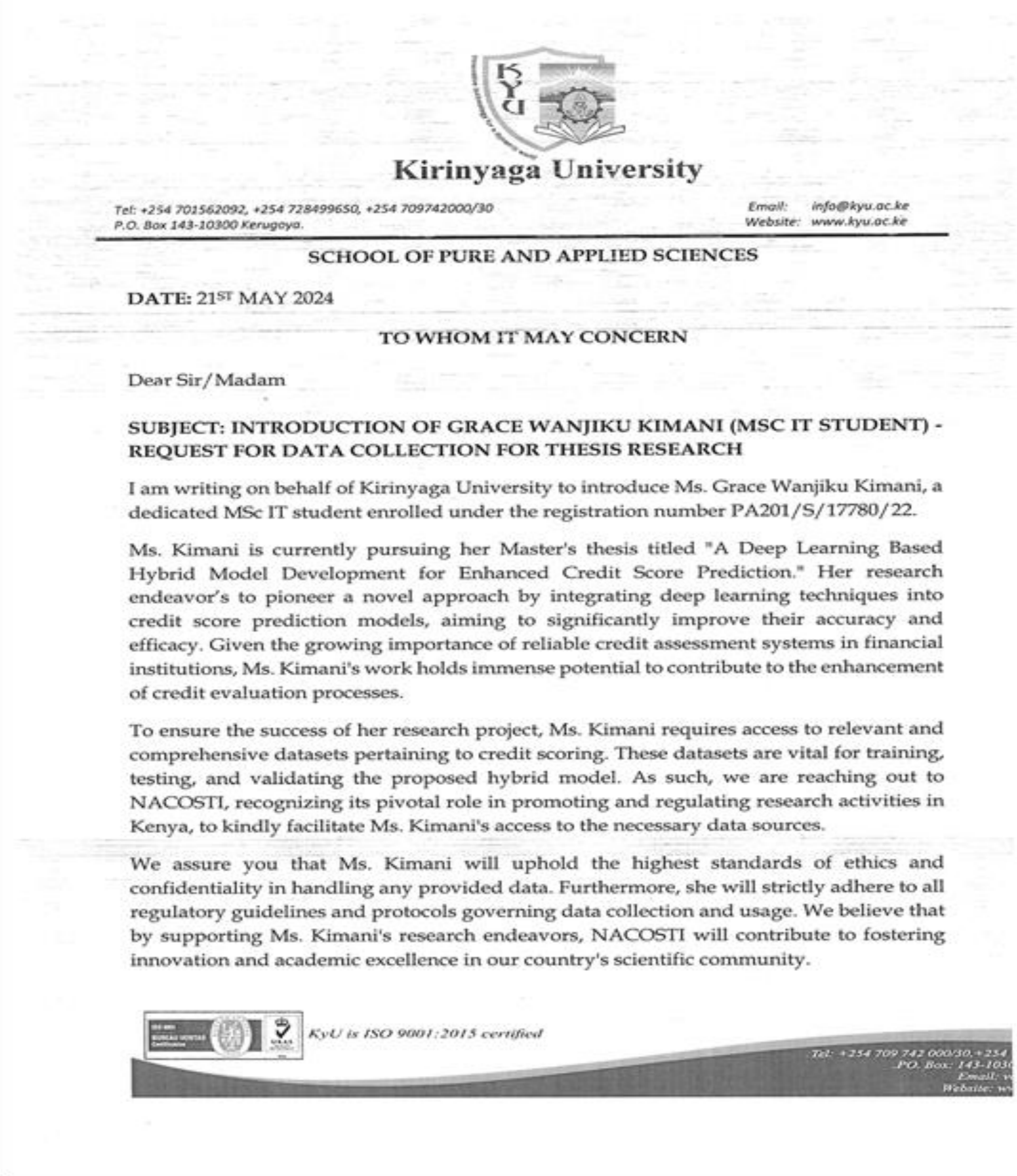
Xia, Y., Guo, X., Li, Y., He, L., & Chen, X. (2022). Deep learning meets decision trees: An application of a heterogeneous deep forest approach in credit scoring for online consumer lending. Journal of Forecasting, 41(8), 1669-1690.

Yeboah, E., & Oduro, I. M. (2018). Determinants of loan defaults in some selected credit

unions in Kumasi Metropolis of Ghana. *Open Journal of Business and Management*, *6*(3), 778-795.

Zhang, W., Yang, D., & Zhang, S. (2021). A new hybrid ensemble model with voting-based outlier detection and balanced sampling for credit scoring. Expert Systems with Applications, 174, 114744.

APPENDICES I: Approval letter from KyU to carry out this research



Kirinyaga University

Tel: +254 701562092, +254 728499650, +254 709742000/30
P.O. Box 143-10300 Kerugoya.

Email: info@kyu.ac.ke
Website: www.kyu.ac.ke

SCHOOL OF PURE AND APPLIED SCIENCES

DATE: 21ST MAY 2024

TO WHOM IT MAY CONCERN

Dear Sir/Madam

SUBJECT: INTRODUCTION OF GRACE WANJIKU KIMANI (MSC IT STUDENT) - REQUEST FOR DATA COLLECTION FOR THESIS RESEARCH

I am writing on behalf of Kirinyaga University to introduce Ms. Grace Wanjiku Kimani, a dedicated MSc IT student enrolled under the registration number PA201/S/17780/22.

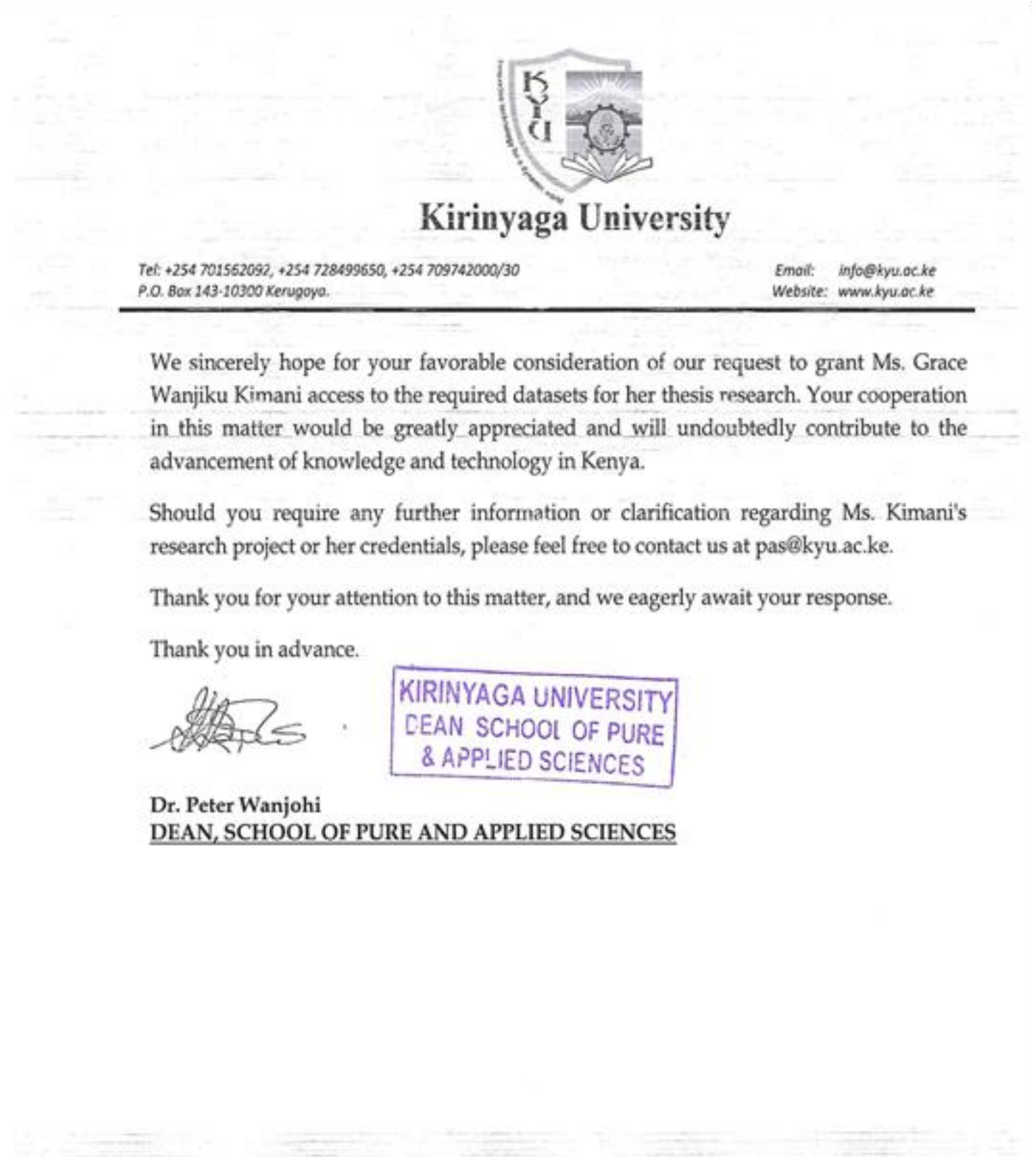
Ms. Kimani is currently pursuing her Master's thesis titled "A Deep Learning Based Hybrid Model Development for Enhanced Credit Score Prediction." Her research endeavor's to pioneer a novel approach by integrating deep learning techniques into credit score prediction models, aiming to significantly improve their accuracy and efficacy. Given the growing importance of reliable credit assessment systems in financial institutions, Ms. Kimani's work holds immense potential to contribute to the enhancement of credit evaluation processes.

To ensure the success of her research project, Ms. Kimani requires access to relevant and comprehensive datasets pertaining to credit scoring. These datasets are vital for training, testing, and validating the proposed hybrid model. As such, we are reaching out to NACOSTI, recognizing its pivotal role in promoting and regulating research activities in Kenya, to kindly facilitate Ms. Kimani's access to the necessary data sources.

We assure you that Ms. Kimani will uphold the highest standards of ethics and confidentiality in handling any provided data. Furthermore, she will strictly adhere to all regulatory guidelines and protocols governing data collection and usage. We believe that by supporting Ms. Kimani's research endeavors, NACOSTI will contribute to fostering innovation and academic excellence in our country's scientific community.

KyU is ISO 9001:2015 certified

Tel: +254 709 742 000/30,+254
P.O. Box: 143-103
Email: w
Website: w



Kirinyaga University

Tel: +254 701562092, +254 728499650, +254 709742000/30
P.O. Box 143-10300 Kerugoya.

Email: info@kyu.ac.ke
Website: www.kyu.ac.ke

We sincerely hope for your favorable consideration of our request to grant Ms. Grace Wanjiku Kimani access to the required datasets for her thesis research. Your cooperation in this matter would be greatly appreciated and will undoubtedly contribute to the advancement of knowledge and technology in Kenya.

Should you require any further information or clarification regarding Ms. Kimani's research project or her credentials, please feel free to contact us at pas@kyu.ac.ke.

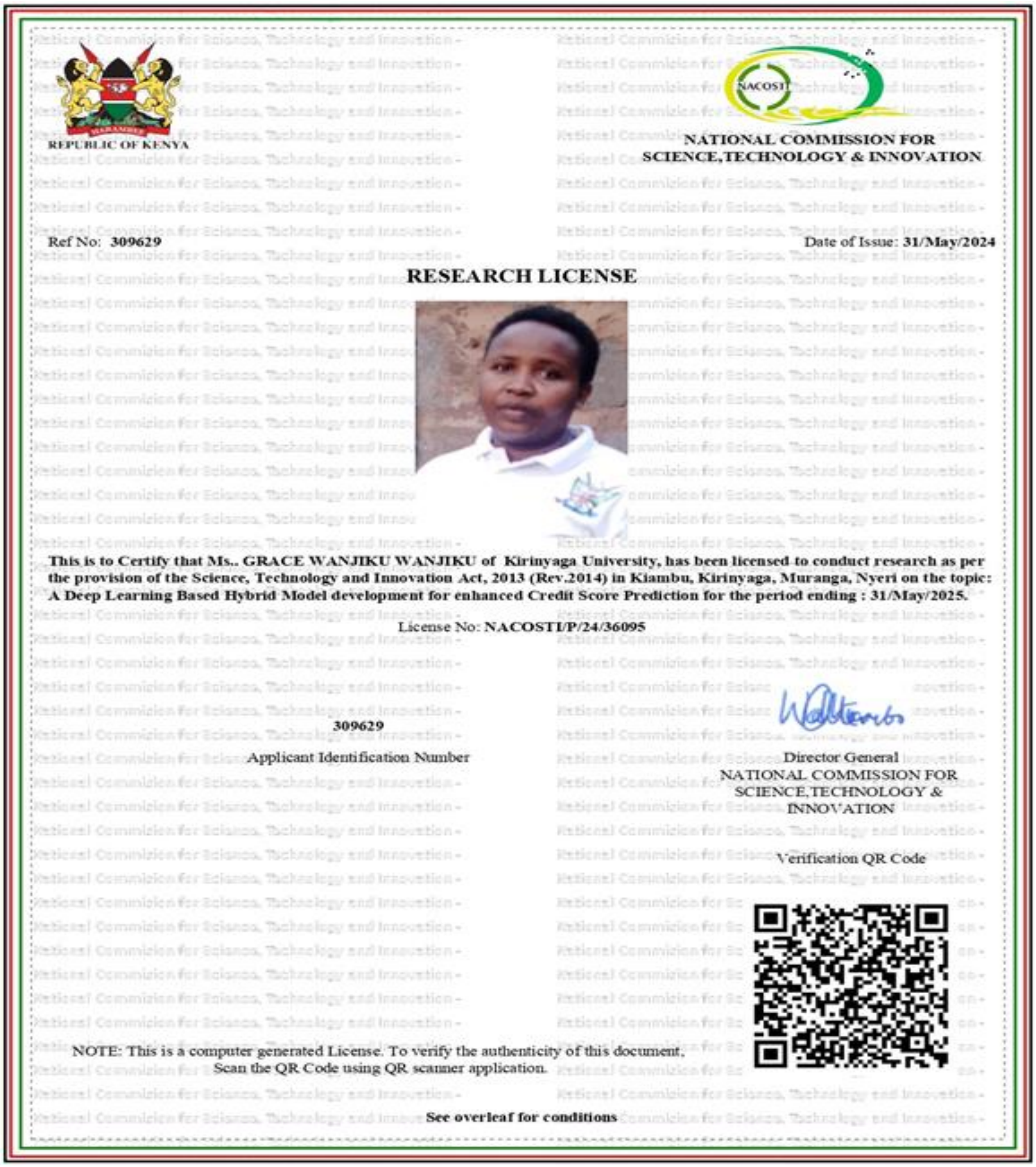
Thank you for your attention to this matter, and we eagerly await your response.

Thank you in advance.

KIRINYAGA UNIVERSITY
DEAN SCHOOL OF PURE
& APPLIED SCIENCES

Dr. Peter Wanjohi
DEAN, SCHOOL OF PURE AND APPLIED SCIENCES

APPENDICES II : Nacosti Research Permit



REPUBLIC OF KENYA

NATIONAL COMMISSION FOR
SCIENCE,TECHNOLOGY & INNOVATION

Ref No: **309629**

Date of Issue: **31/May/2024**

RESEARCH LICENSE

This is to Certify that Ms.. GRACE WANJIKU WANJIKU of Kirinyaga University, has been licensed to conduct research as per the provision of the Science, Technology and Innovation Act, 2013 (Rev.2014) in Kiambu, Kirinyaga, Muranga, Nyeri on the topic: A Deep Learning Based Hybrid Model development for enhanced Credit Score Prediction for the period ending : 31/May/2025.

License No: **NACOSTI/P/24/36095**

309629

Applicant Identification Number

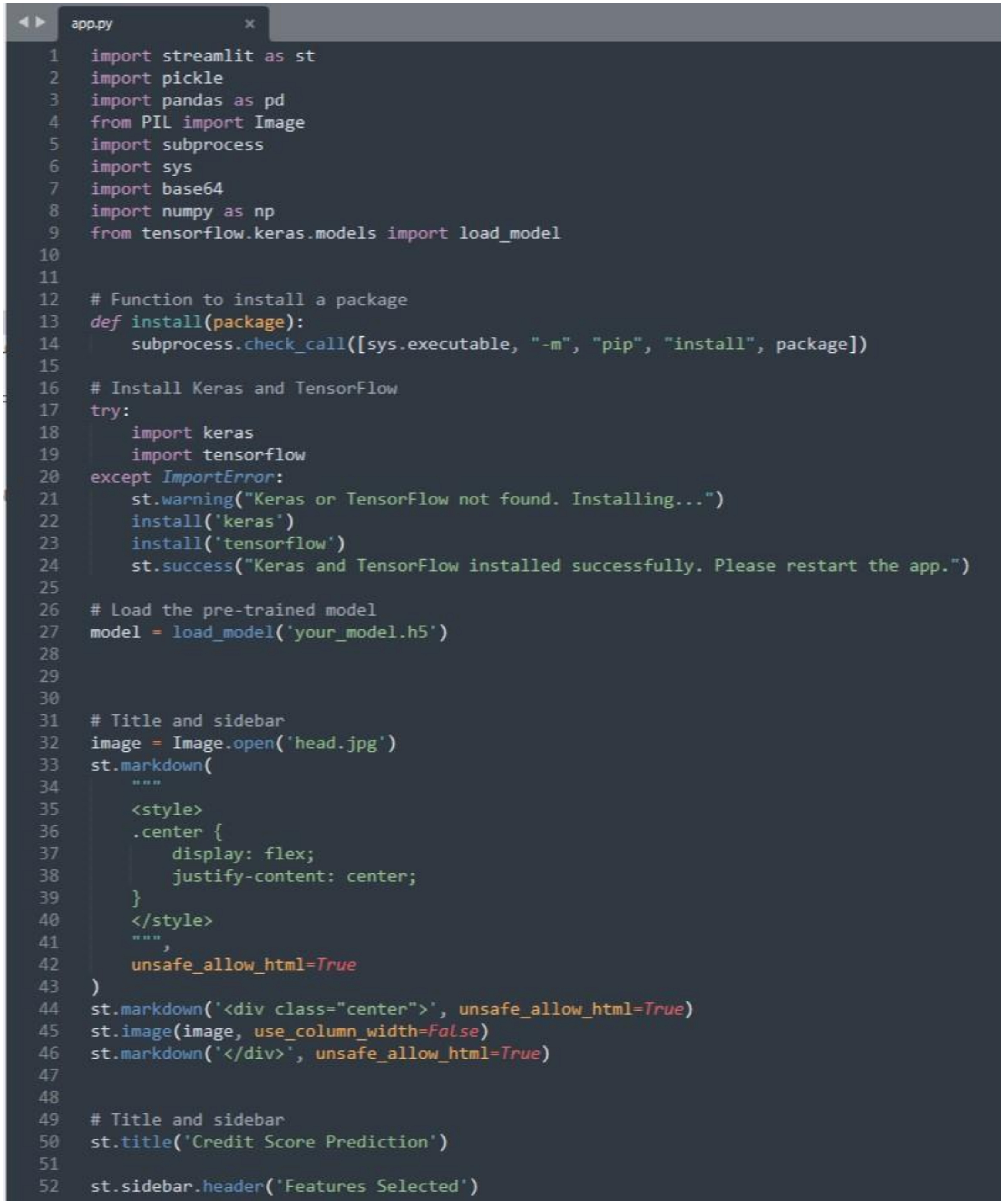
Director General
NATIONAL COMMISSION FOR
SCIENCE,TECHNOLOGY &
INNOVATION

Verification QR Code

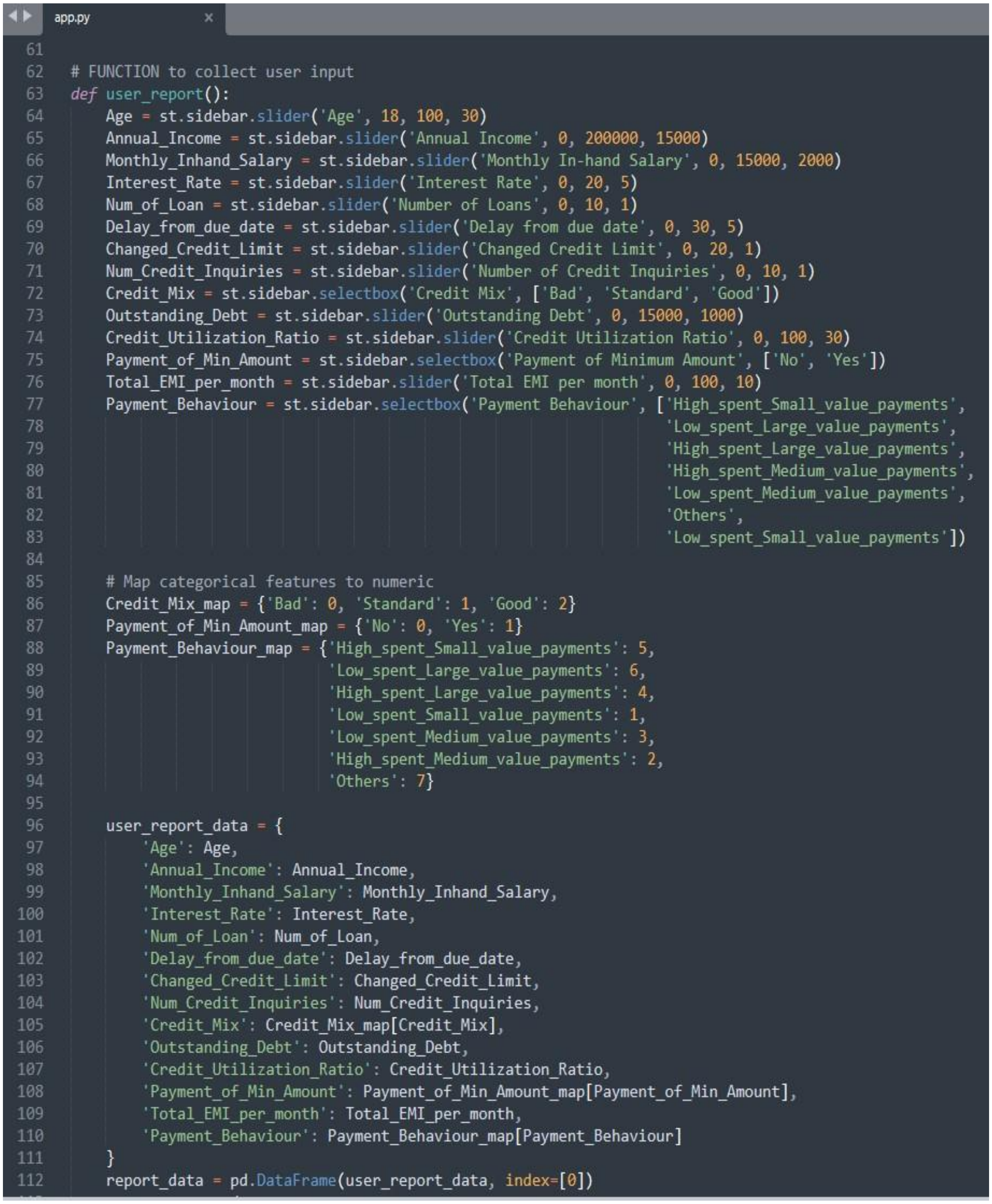
NOTE: This is a computer generated License. To verify the authenticity of this document,
Scan the QR Code using QR scanner application.

See overleaf for conditions

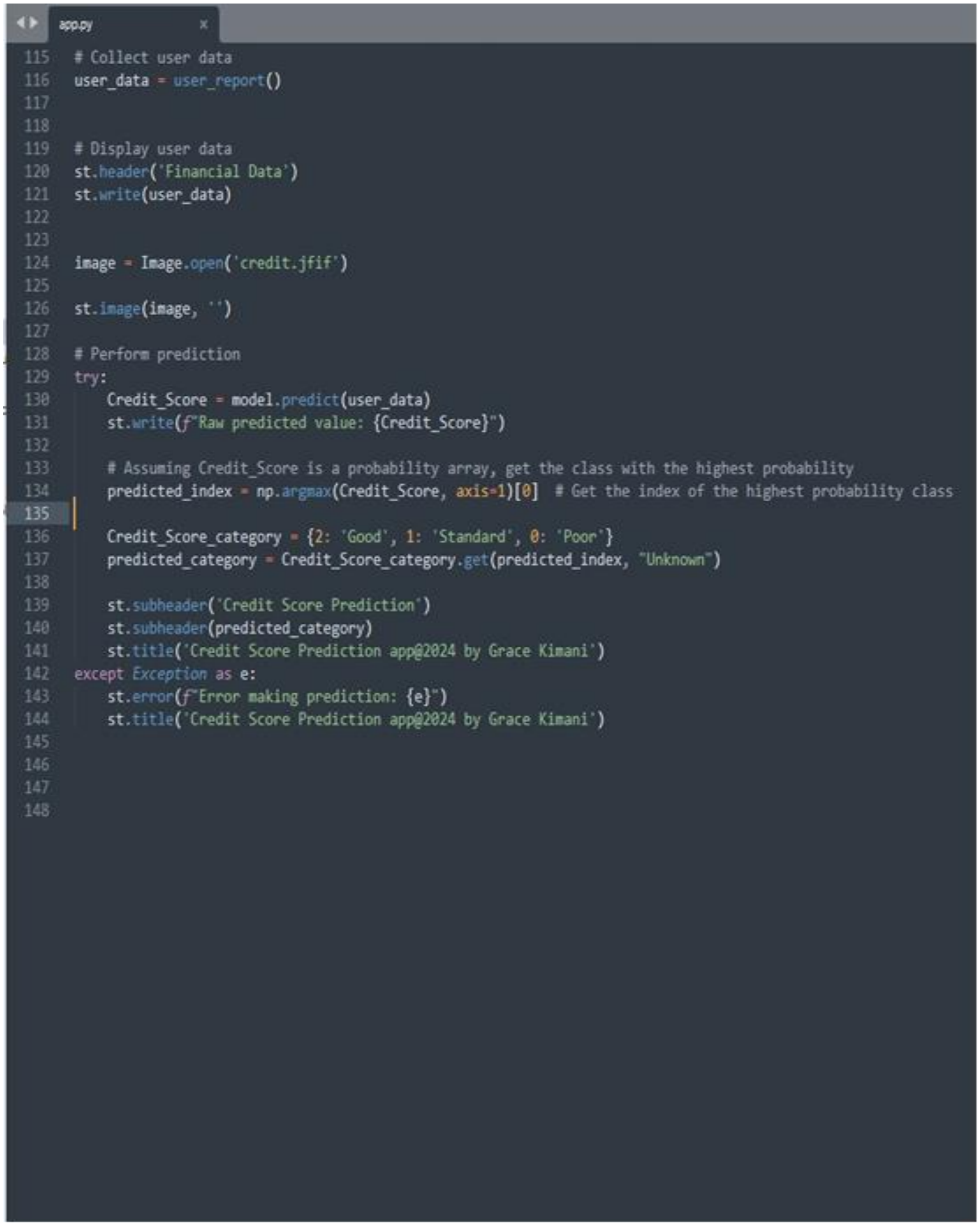
APPENDICES III: Web-Based Tool Code



```python
import streamlit as st
import pickle
import pandas as pd
from PIL import Image
import subprocess
import sys
import base64
import numpy as np
from tensorflow.keras.models import load_model


# Function to install a package
def install(package):
    subprocess.check_call([sys.executable, "-m", "pip", "install", package])

# Install Keras and TensorFlow
try:
    import keras
    import tensorflow
except ImportError:
    st.warning("Keras or TensorFlow not found. Installing...")
    install('keras')
    install('tensorflow')
    st.success("Keras and TensorFlow installed successfully. Please restart the app.")

# Load the pre-trained model
model = load_model('your_model.h5')



# Title and sidebar
image = Image.open('head.jpg')
st.markdown(
    """
    <style>
    .center {
        display: flex;
        justify-content: center;
    }
    </style>
    """,
    unsafe_allow_html=True
)
st.markdown('<div class="center">', unsafe_allow_html=True)
st.image(image, use_column_width=False)
st.markdown('</div>', unsafe_allow_html=True)


# Title and sidebar
st.title('Credit Score Prediction')

st.sidebar.header('Features Selected')
```

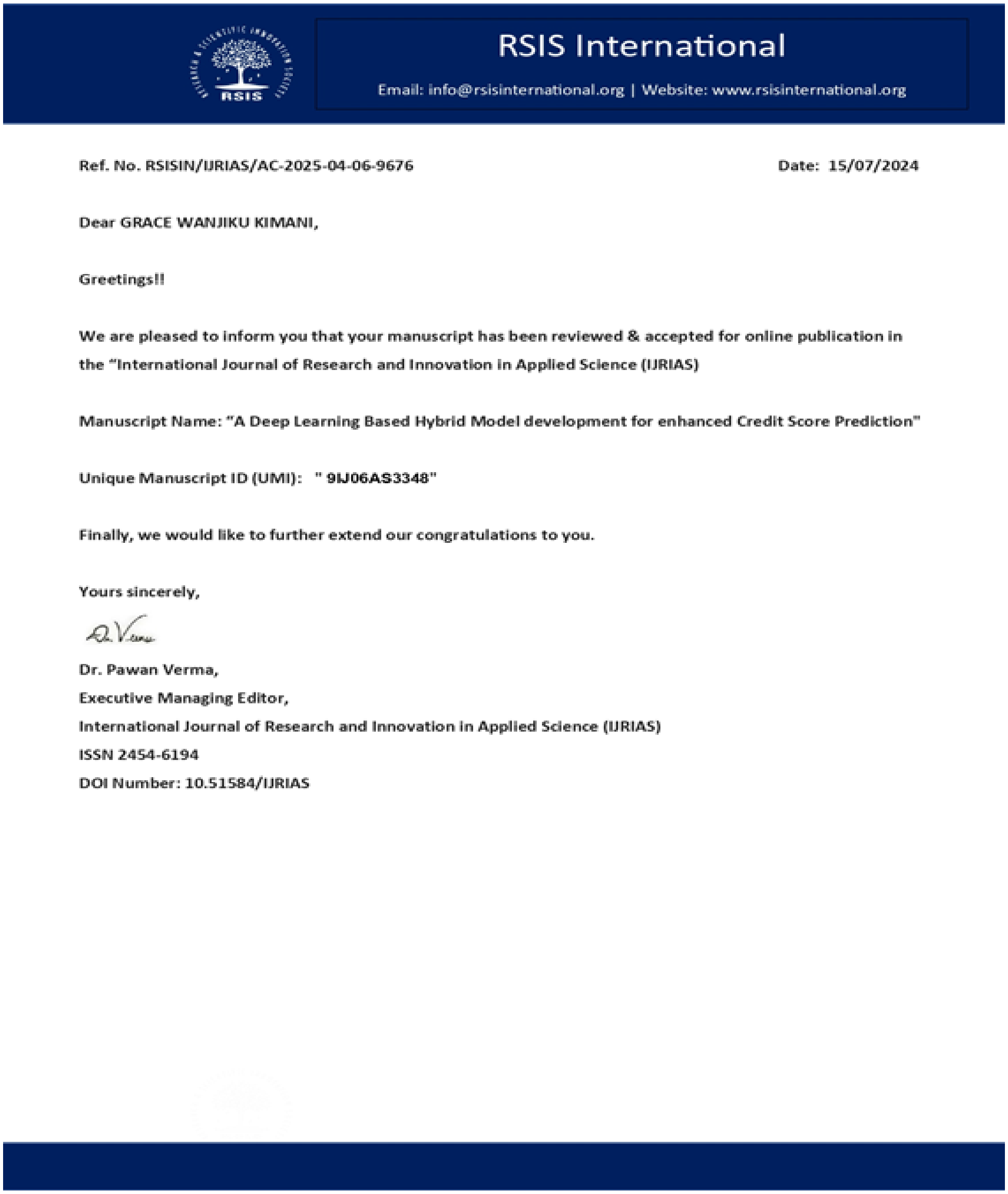


```python

# FUNCTION to collect user input
def user_report():
    Age = st.sidebar.slider('Age', 18, 100, 30)
    Annual_Income = st.sidebar.slider('Annual Income', 0, 200000, 15000)
    Monthly_Inhand_Salary = st.sidebar.slider('Monthly In-hand Salary', 0, 15000, 2000)
    Interest_Rate = st.sidebar.slider('Interest Rate', 0, 20, 5)
    Num_of_Loan = st.sidebar.slider('Number of Loans', 0, 10, 1)
    Delay_from_due_date = st.sidebar.slider('Delay from due date', 0, 30, 5)
    Changed_Credit_Limit = st.sidebar.slider('Changed Credit Limit', 0, 20, 1)
    Num_Credit_Inquiries = st.sidebar.slider('Number of Credit Inquiries', 0, 10, 1)
    Credit_Mix = st.sidebar.selectbox('Credit Mix', ['Bad', 'Standard', 'Good'])
    Outstanding_Debt = st.sidebar.slider('Outstanding Debt', 0, 15000, 1000)
    Credit_Utilization_Ratio = st.sidebar.slider('Credit Utilization Ratio', 0, 100, 30)
    Payment_of_Min_Amount = st.sidebar.selectbox('Payment of Minimum Amount', ['No', 'Yes'])
    Total_EMI_per_month = st.sidebar.slider('Total EMI per month', 0, 100, 10)
    Payment_Behaviour = st.sidebar.selectbox('Payment Behaviour', ['High_spent_Small_value_payments',
                                                                   'Low_spent_Large_value_payments',
                                                                   'High_spent_Large_value_payments',
                                                                   'High_spent_Medium_value_payments',
                                                                   'Low_spent_Medium_value_payments',
                                                                   'Others',
                                                                   'Low_spent_Small_value_payments'])


    # Map categorical features to numeric
    Credit_Mix_map = {'Bad': 0, 'Standard': 1, 'Good': 2}
    Payment_of_Min_Amount_map = {'No': 0, 'Yes': 1}
    Payment_Behaviour_map = {'High_spent_Small_value_payments': 5,
                             'Low_spent_Large_value_payments': 6,
                             'High_spent_Large_value_payments': 4,
                             'Low_spent_Small_value_payments': 1,
                             'Low_spent_Medium_value_payments': 3,
                             'High_spent_Medium_value_payments': 2,
                             'Others': 7}

    user_report_data = {
        'Age': Age,
        'Annual_Income': Annual_Income,
        'Monthly_Inhand_Salary': Monthly_Inhand_Salary,
        'Interest_Rate': Interest_Rate,
        'Num_of_Loan': Num_of_Loan,
        'Delay_from_due_date': Delay_from_due_date,
        'Changed_Credit_Limit': Changed_Credit_Limit,
        'Num_Credit_Inquiries': Num_Credit_Inquiries,
        'Credit_Mix': Credit_Mix_map[Credit_Mix],
        'Outstanding_Debt': Outstanding_Debt,
        'Credit_Utilization_Ratio': Credit_Utilization_Ratio,
        'Payment_of_Min_Amount': Payment_of_Min_Amount_map[Payment_of_Min_Amount],
        'Total_EMI_per_month': Total_EMI_per_month,
        'Payment_Behaviour': Payment_Behaviour_map[Payment_Behaviour]
    }
    report_data = pd.DataFrame(user_report_data, index=[0])
```



```python
# Collect user data
user_data = user_report()


# Display user data
st.header('Financial Data')
st.write(user_data)


image = Image.open('credit.jfif')

st.image(image, '')

# Perform prediction
try:
    Credit_Score = model.predict(user_data)
    st.write(f"Raw predicted value: {Credit_Score}")

    # Assuming Credit_Score is a probability array, get the class with the highest probability
    predicted_index = np.argmax(Credit_Score, axis=1)[0]  # Get the index of the highest probability class

    Credit_Score_category = {2: 'Good', 1: 'Standard', 0: 'Poor'}
    predicted_category = Credit_Score_category.get(predicted_index, "Unknown")

    st.subheader('Credit Score Prediction')
    st.subheader(predicted_category)
    st.title('Credit Score Prediction app@2024 by Grace Kimani')
except Exception as e:
    st.error(f"Error making prediction: {e}")
    st.title('Credit Score Prediction app@2024 by Grace Kimani')
```

APPENDICES IV: Publication evidence



RSIS International

Email: info@rsisinternational.org | Website: www.rsisinternational.org

Ref. No. RSISIN/IJRIAS/AC-2025-04-06-9676 Date: 15/07/2024

Dear GRACE WANJIKU KIMANI,

Greetings!!

We are pleased to inform you that your manuscript has been reviewed & accepted for online publication in the "International Journal of Research and Innovation in Applied Science (IJRIAS)

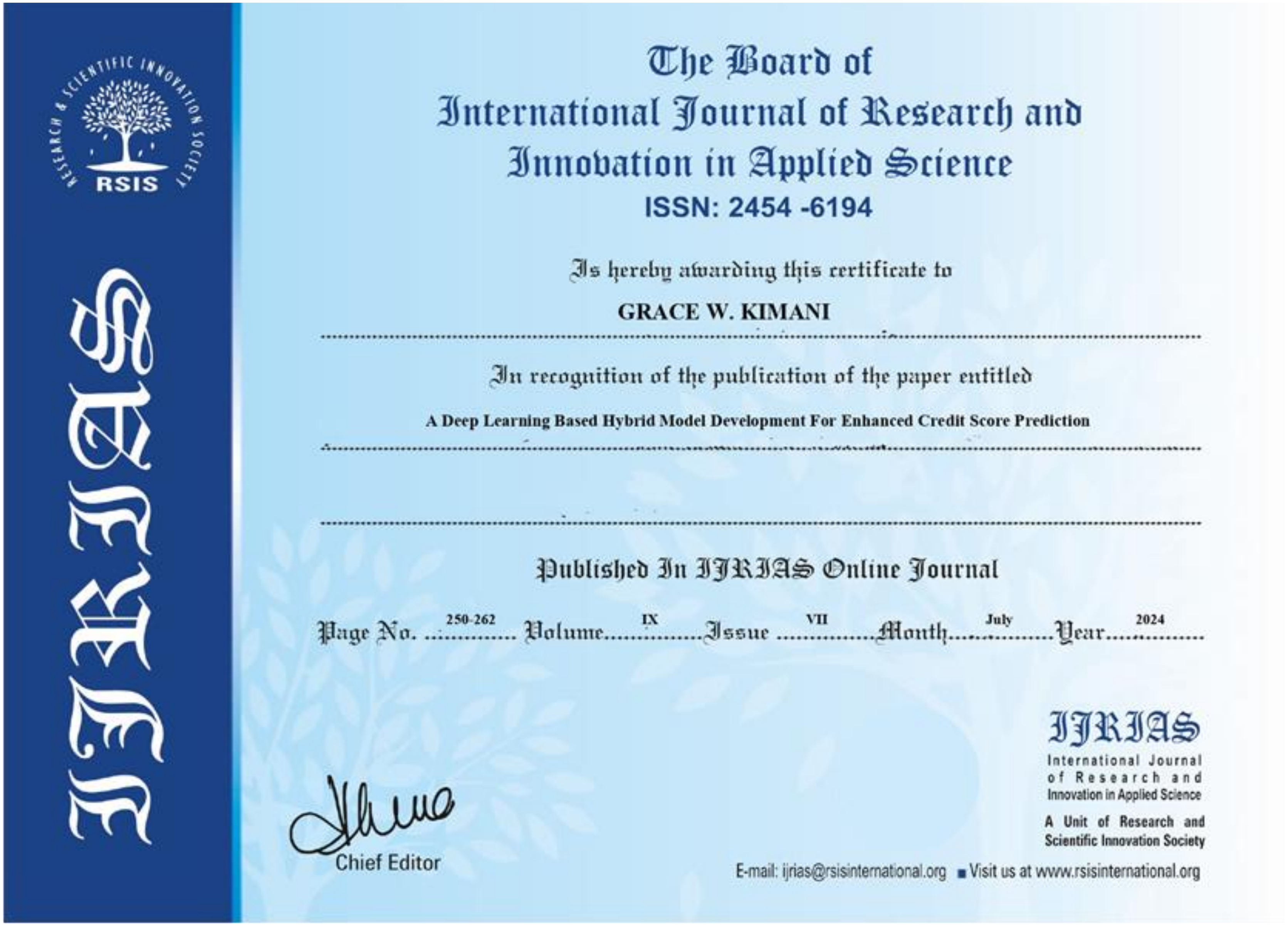
Manuscript Name: "A Deep Learning Based Hybrid Model development for enhanced Credit Score Prediction"

Unique Manuscript ID (UMI): " 9IJ06AS3348"

Finally, we would like to further extend our congratulations to you.

Yours sincerely,

Dr. Pawan Verma,
Executive Managing Editor,
International Journal of Research and Innovation in Applied Science (IJRIAS)
ISSN 2454-6194
DOI Number: 10.51584/IJRIAS



IJRIAS

RESEARCH & SCIENTIFIC INNOVATION SOCIETY

RSIS

The Board of
International Journal of Research and
Innovation in Applied Science
ISSN: 2454 -6194

Is hereby awarding this certificate to

GRACE W. KIMANI

In recognition of the publication of the paper entitled

A Deep Learning Based Hybrid Model Development For Enhanced Credit Score Prediction

Published In IJRIAS Online Journal

Page No. ...250-262... Volume...IX...Issue ...VII...Month...July...Year...2024...

Chief Editor

IJRIAS
International Journal
of Research and
Innovation in Applied Science

A Unit of Research and
Scientific Innovation Society

E-mail: ijrias@rsisinternational.org ■ Visit us at www.rsisinternational.org