A Tool to Predict Mortgage Default and Recommend Mortgage Amount Using Convolution Neural Networks

By

Okola Dan Naftali

Student ID: 095873

Submitted in partial fulfilment of the requirements for the Degree of Master of Science in Information Technology at Strathmore University

School of Computing and Engineering Sciences

Strathmore University

Nairobi, Kenya

June 2024

Declaration

The thesis documentation is my original work and has not been submitted to any other institution for the award of a degree.

Signature:  Date: 07.04.2024

Student's name: Okola Dan Naftali

Registration Number: 095873

I confirm that the thesis presented was carried out by the student under my supervision and has been submitted for examination with my approval.

Signature: ..... Date: .....07/04/2024.....

Supervisor's name: Professor Ismail Ateya

School of Computing and Engineering Sciences,
Strathmore University.

Abstract

The mortgage sector is vital to the financial services industry and the Kenyan economy in general. In the period preceding March 2021, mortgage defaults surged by 48 percent to reach Sh70.5 billion in Kenya, signalling widespread distress within the real estate sector in the aftermath of economic challenges triggered by the Covid-19 pandemic. This surge was accompanied by a notable increase in property auctions. According to the latest data released by the Central Bank of Kenya (CBK), mortgages experienced the most substantial rise in non-performing loans (NPLs), soaring from Sh47.5 billion in March 2020. Overdue mortgages witnessed a staggering increase of Sh9.1 billion, equivalent to 14.8%, within the three-month period leading up to March, surpassing the default rates observed in other sectors such as manufacturing (3%), agriculture (10.7%), and personal loans (3%). As businesses adopt stringent cost-cutting measures to safeguard profits, mortgage holders find themselves grappling with financial strain in an economy marred by widespread job losses across various sectors since the onset of the Covid-19 pandemic in Kenya. Consequently, individuals who secured mortgages based on their employment income are now facing challenges in meeting their repayment obligations. The downturn in the real estate market poses significant challenges for property developers, who find themselves unable to offload units constructed using loans. This research led to the development of a tool that aids banks and other lending institutions in predicting the likelihood of a client defaulting on their mortgages using convolutional neural networks. The developed tool further recommends mortgage amount to the lenders to minimize the risk of defaulting. The developed model attained an impressive accuracy rate of 97.14%, surpassing the accuracy scores of Gradient Boosting and only slightly behind KNN model, which achieved 88.49% and 99.24%, respectively. The Agile Methodology was selected as the preferred approach owing to its emphasis on collaboration and facilitation of continuous improvement processes.

Keywords: non-performing loans, mortgage default, convolutional neural networks.

Table of Contents

List of Figures

List of Tables

List of Equations

Abbreviations and Acronym

AUC	Area Under the Curve
CPU	Central Processing Unit
CNN	Convolutional Neural Networks
CBK	Central Bank of Kenya
DL	Deep Learning
DPNN	Deep Learning Neural Network
GPU	Graphics Processing Unit
KNN	K Nearest Neighbor
ML	Machine Learning
NPL	Non-Performing Loans
OOB	Out-of-bag
RF	Random Forest
ROC	Receiver Operating Characteristic Curve
SVM	Support Vector Machine

Acknowledgements

I would like to thank my supervisor, Professor Ismail Ateya of Strathmore University's School of Computing and Engineering Sciences (SCES), for his consistent guidance, insightful comments, and unwavering forbearance throughout this research. Additionally, I would like to thank my family, colleagues, and classmates for their consistent encouragement and support.

Definition of Terms

Convolutional Neural Networks	The Convolutional Neural Network (CNN) represents a sophisticated deep learning model tailored for analyzing data exhibiting a grid-like structure, notably images. Drawing inspiration from the organization of the animal visual cortex, CNNs are designed to autonomously and flexibly acquire spatial hierarchies of features, progressing from rudimentary to intricate patterns (Yamashita et al., 2018).
Mortgage	It is a loan that is secured by real property (Fabozzi et al., 2010)
Mortgage Default	Neglecting to take action, particularly in meeting a financial obligation, constitutes a failure (Siaw et al., 2014).
Non-Performing Loan	A bank loan is considered non-performing when more than 90 days pass without the borrower paying the agreed instalments or interest (Cheruiyot, 2015).

Chapter 1: Introduction

1.1 Background

A mortgage represents a financial obligation secured by collateral, wherein the debtor commits to repaying the creditor according to a predetermined schedule. As posited by (Object, 2017), mortgage financing plays a pivotal role in bolstering the housing market and broader economy, fostering transactions directly and enhancing the transactional landscape indirectly. Moreover, mortgages serve as valuable collateral assets. They emerge as the most cost-effective means for households to obtain financing for various purposes, including consumption, non-real estate investments, and business ventures. By leveraging mortgages, individuals and businesses can undertake substantial real estate acquisitions without the need to make full upfront payments.

According to mathematical standards, mortgages are among the most complex securities. A multitude of factors underlie these intricacies, encompassing diverse repayment options accessible to homeowners, the challenge of accurately modeling mortgage behaviors across varying economic scenarios, the heterogeneous nature of loan-level behaviors, and the inherent complexity of analyzing path-dependent instruments like mortgages over discrete time periods. Moreover, it's noteworthy that the conduct of an individual loan may vary significantly based on prevailing economic conditions. Similarly, various loan types respond differentially to identical economic conditions.

Assessing the resilience of a financial institution often entails scrutinizing the performance of its loan portfolio, particularly under stressed conditions, gauging the likelihood of default. This process, integral to risk management and credit risk analysis, underscores the importance of estimating default probabilities accurately. Historically, in the early 1980s, mortgage default assessments relied heavily on heuristic approaches and credit risk ratings derived from professional expertise (Akindaini & Juhola, 2017). These ratings, predicated on four key variables—debt-to-income ratio (DTI), monthly repayment-to-income ratio, loan-to-value ratio (LTV), and property value-to-income ratio—proved inadequate in quantifying default risk for two primary reasons.

First, risk ratings cannot be used to ascertain the quantitative assessment of the default probability. Secondly, traditional methodologies overlooked the temporal dimension of default occurrences, as they predominantly gauged the likelihood or probability of default within the mortgage's active period. Eventually, conventional econometric models, such as linear regression and generalized linear regression, were employed to scrutinize mortgage defaults. These approaches primarily sought to ascertain causality among specific predictors based on established theoretical frameworks, with minimal consideration given to out-of-sample accuracy. In contrast, the advent of machine learning represents a burgeoning paradigm in financial modeling. Machine learning methodologies offer the advantage of quantitatively assessing defaults, scrutinizing outcomes in terms of out-of-sample accuracy through delineating test and training sets and analyzing default timing dynamics.

The year 2007 witnessed a surge in subprime mortgages, triggering a cascade of financial crises marked by a soaring rate of mortgage defaults. Previously, the mortgage-backed securities market was predominantly governed by non-governmental entities, with a considerable portion of mortgages adhering to the guidelines set forth by Fannie Mae and Freddie Mac. Subsequently, the market underwent a speculative bubble, heavily influenced by consumer sentiments and economic oscillations. This bubble stemmed from the relaxation of mortgage regulations and guidelines by non-governmental entities. For banks and other lending institutions, the ability to distinguish bad consumers from good ones is crucial. A modest increase in mortgage prediction accuracy may result in a substantial profit increase. Early identification of high-risk clients can prevent mortgage defaults and help consumers better manage their finances (Kvamme et al., 2018).

This study presents a novel approach to predict mortgage default and recommend mortgage amount using consumer transaction data and convolutional neural networks (CNNs). The developed tool integrates a range of financial metrics, encompassing checking account balance, savings account balance, and credit card balance. Additionally, it incorporates transactional data, comprising the daily frequency of transactions on the checking account and the volume of funds deposited into the checking account. The goal of the study was to leverage the rich information available in consumer transaction data to develop an accurate and reliable model for predicting

mortgage default. By analyzing the financial behavior of individuals, the tool identifies patterns and trends that may indicate a higher risk of defaulting on mortgage payments.

1.2 Problem Statement

Providing individuals and families with access to homeownership, the mortgage industry plays a crucial role in the global economy. However, mortgage default, which occurs when borrowers fail to repay their mortgage obligations, presents significant challenges for lenders and has negative repercussions for financial institutions, borrowers, and the economy as a whole (Kvamme et al., 2018). It is essential for lenders to accurately foresee mortgage default in order to effectively manage risks, allocate resources, and make informed lending decisions.

Traditional methods for predicting mortgage default rely on manual assessment and historical data analysis, which are typically time-consuming, labour-intensive, and susceptible to human biases. In addition, they may not completely capture the dynamic and complex nature of mortgage default risk, resulting in suboptimal forecasts (Martha & Daniel, 2014). To address these limitations, there is an urgent need for a sophisticated instrument that employs machine learning to improve the precision, efficiency, and dependability of mortgage default prediction.

This study presents a novel approach to predict mortgage default and recommend mortgage amount using consumer transaction data and convolutional neural networks (CNNs). The developed tool utilizes various financial indicators such as checking account balance, savings account balance, and credit card balance, along with transactional data including the daily number of transactions on the checking account and the amount transmitted into the checking account. The goal of the study was to leverage the rich information available in consumer transaction data to develop an accurate and reliable model for predicting mortgage default. By analyzing the financial behaviour of individuals, the tool aimed at identifying the patterns and trends that may indicate a higher risk of defaulting on mortgage payments.

1.3 Aim

To develop a tool to predict mortgage default and recommend mortgage amount using convolutional neural networks.

1.4 Specific Objectives

i. To identify the factors that leads to mortgage default prediction.
ii. To review the techniques that have been used for mortgage default prediction.
iii. To review the models and frameworks that are used for mortgage default prediction.
iv. To develop a tool that can predict mortgage default and recommend mortgage borrowing amount.
v. To test and evaluate the developed system.

1.5 Research Questions

i. What are the major factors that causes mortgage default?
ii. What are the techniques currently being used for mortgage default prediction?
iii. What models and frameworks are being used for mortgage default prediction?
iv. How can a tool to predict mortgage default and recommend mortgage amount using convolutional neural networks be developed?
v. How can the developed system be tested and validated?

1.6 Study Justification

Despite the crucial role of mortgages in Kenya's financial landscape, there have been few studies exploring mortgage prediction specifically in this context. This research opens the door to address this knowledge gap and gain deeper insights into Kenyan credit scoring. In contemporary research, mortgage assessment heavily relies on heuristic, manually designed features, leading to challenges in selecting the most appropriate features or covariates. This research seeks to revolutionize the mortgage assessment process by employing a convolutional neural network (CNN) to systematically automate feature engineering. Harnessing the capabilities of artificial intelligence and deep learning, this research endeavours to diminish human biases, thereby enhancing the precision and efficacy of mortgage evaluations.

Kenya's credit scoring landscape presents a unique opportunity for research, as there have been relatively few studies conducted in the domain of mortgage prediction. The lack of comprehensive research opens the door for this study to fill the research gap and contribute valuable insights to the field of credit scoring within the Kenyan context. While credit scoring studies have been conducted, many have overlooked the value of incorporating additional explanatory variables to enhance prediction models. Instead, researchers often focus on comparing various scoring models without exploring the potential benefits of including new variables. This study aimed to explore the impact of introducing more explanatory variables in mortgage prediction, thereby contributing to a more comprehensive and accurate credit scoring process.

Furthermore, despite the extensive research in credit scoring, a consumer-facing tool that empowers individuals to understand and improve their creditworthiness remains largely absent. To bridge this gap, this research did not only focus on developing an automated mortgage assessment system but also led to the design of a user-friendly tool that provides consumers with insights into the factors influencing their mortgage applications.

1.7 Scope and Limitations

The study has some inherent limitations that should be acknowledged. Firstly, it solely concentrates on mortgage assessment and the recommendation of loan amounts, disregarding other types of loans. This narrow scope might restrict the generalizability of the findings to other loan categories, and further research would be needed to encompass a broader range of loan products. The study's location-specific focus on Kenya also poses limitations to its external validity. The findings may not be directly applicable to other countries or regions with different economic, regulatory, and cultural contexts. Care should be taken when extrapolating the results beyond the Kenyan credit market.

Chapter 2: Literature Review

2.1 Introduction

The realm of mortgage default prediction has garnered substantial attention across diverse academic disciplines, revealing a complex landscape of factors influencing this critical phenomenon. Researchers from fields spanning economics, finance, data science, and sociology have dedicated substantial effort to unravelling the intricacies of mortgage default, resulting in a plethora of valuable insights. In light of the evolving mortgage landscape, this review aims to distill the salient threads of mortgage default prediction literature. By synthesizing findings from diverse disciplinary perspectives and emphasizing the pertinence of context, this review endeavours to contribute to a more nuanced understanding of mortgage default prediction, thereby informing prudent lending practices in an ever-changing financial environment.

2.2 Empirical Literature

Empirical literature refers to research based on empirical data, facts, or evidence collected through observation or experimentation. Empirical research is typically based on the scientific method, where a hypothesis is tested through the collection and analysis of data. Empirical literature can include research articles, case studies, and primary data sources such as survey reports or datasets. Mortgage default prediction has received much attention from researcher with various empirical studies conducted to analyse consumer patterns and predict the likelihood of default.

Akindaini and Juhola's (2017) research on mortgage default prediction adopted a dual focus, comprising both data-centric and methodological approaches. The data-centric aspect centred on the utilization of mortgage datasets sourced from Fannie Mae, while the methodological aspect involved the application of diverse machine learning techniques for classifying mortgages into default, paying, and prepaid categories. Employing exploratory analysis, the researchers scrutinized the dataset's structure and investigated the correlation between default rates and specific variables. Additionally, economic indicators such as the unemployment rate and rent ratio were incorporated into the dataset, drawing from prior knowledge suggesting their relevance to default rates.

In their research, Akindaini and Juhola (2017) employed a repertoire of five distinct machine learning methodologies to tackle classification tasks. These methodologies encompassed simple logistic regression, multinomial-multiclass logistic regression, naive Bayes classifier, random forest model, and KNN classifier. The researchers assessed the efficacy of each model, gauging their performance through measures such as overall accuracy and pertinent test statistics. Additionally, they scrutinized the influence and contribution of individual variables towards the accuracy of each model. Results revealed a spectrum of performances, with the Random Forest model emerging as a frontrunner, boasting an impressive accuracy of 95.68%, while the Naïve Bayes classifier lagged behind with a modest accuracy of 70.74%. Notably, the simple logistic regression model exhibited an overall accuracy of 95.14%, albeit constrained to binary classification tasks. Despite this limitation, most variables demonstrated statistical significance within the model. The multinomial logistic regression model yielded marginal improvement over Naïve Bayes, achieving a model accuracy of 74%, which was further enhanced to 84.81% through strategic variable elimination using backward elimination technique. Regarding the KNN model, varying accuracies were observed, with the highest accuracy of 83.14% achieved at $k = 15$ and the lowest at 39% for $k = 3$.

Zhang (2015) aimed to predict the probability of default (PD) within a mortgage portfolio in their study. Employing logistic regression and survival analysis techniques on a substantial dataset sourced from a prominent national bank in China, the researcher explored the comparative efficacy of these methodologies. Survival analysis, previously underutilized in this domain, was presented as a promising alternative to traditional logistic regression. The results of both approaches demonstrated closely matched performances, particularly evident in the receiver operating characteristic curve (ROC). Notably, the survival model exhibited marginally superior performance in the training dataset, while achieving near parity with logistic regression in the testing dataset. Although logistic regression excelled in predicting defaulted and non-defaulted mortgage portfolios in the training dataset, the survival model showcased superior performance in the testing dataset, underscoring its viability as a competitive alternative.

In the study conducted by Kvamme et al. (2018), an intriguing approach was undertaken to forecast mortgage default utilizing convolutional neural networks (CNNs) on consumer transaction data. The intricacy of their method lay in the utilization of specific financial indicators from each consumer, including the balances within their checking and savings accounts, along with credit card data. Furthermore, the researchers incorporated the daily count of transactions linked to the checking account and the volume of funds transferred into it. Remarkably, this predictive model operated solely on these financial facets, without delving into additional consumer particulars. The outcomes were remarkably promising, showcasing the potency of their approach. The convolutional neural networks attained an impressive ROC AUC score of 0.918, which indicated the model's capability in distinguishing between mortgage defaulters and non-defaulters. Moreover, when these CNNs were integrated with a random forests classifier, the predictive power soared even higher, yielding a ROC AUC of 0.926. This innovative fusion of machine learning techniques not only underscores the potential for accurate mortgage default prediction but also attests to the formidable synergy that can be harnessed through hybrid models. Just as the examined financial data interweaved to provide a comprehensive picture, so too did the amalgamation of CNNs and random forests, harmonizing their strengths and magnifying their predictive prowess.

The research on credit scoring on personal loans has also garnered much interest from scholars. Li (2022) conducted a study primarily focused on the development of a network loan default prediction model with DPNN. The initial step was an analysis of the issues and potential hazards associated with the peer-to-peer (P2P) online lending platform. Subsequently, a comprehensive explanation was provided about the principles and distinguishing features of the Backpropagation Neural Network (BPNN). Finally, a credit risk rating procedure for online lending was established by leveraging the BPNN methodology. Utilizing advanced data analysis and processing software, a suite of online lending default risk assessment models is developed employing Backpropagation Neural Network (BPNN) techniques. These models are constructed based on credit customer data obtained from lending clubs, which has thorough cleaning and variable selection processes. The experimental findings indicate that the BPNN model achieved a maximum accuracy rate of 98.01% and a maximum recall rate of 99.82%, exceeding the performance of the remaining two models. Furthermore, the AUC value of the BPNN model was

0.79, demonstrating a statistically significant improvement compared to the support vector machine and regression models.

Table 2.1 shows a summary of the empirical studies conducted on mortgage default predictions.

Author	Techniques and Accuracy	Loan Type	Limitations
(Akindaini and Juhola, 2017)	-simple logistic regression-95.14% -multinomial-multiclass logistic regression-74% -Naive Bayes -70.74% -random forest - 95.68% -KNN - 83.14%	Mortgage	- The models proposed in this study are broad in scope, deriving outcomes from the variables contained within the dataset. However, to enhance the robustness of the model, it is imperative to integrate additional variables that provide insights into the characteristics of the mortgage owner. These supplementary factors may encompass gender, income level, occupation, and the degree of occupational volatility.
(Zhang ,2015)	-logistic regression	Mortgage	-This research used mortgage loan data obtained during the year 2004. This makes it difficult to apply the same results to the current economic environment.
(Kvamme et al., 2018)	-CNN – ROC 0.91	Mortgage	-This research failed to add more explanatory variables that are crucial while determing loan default. -The study was conducted in India which has a different

			economic landscape compare to Kenya. -The study did not develop a consumer facing tool.
(Li ,2022)	-BPNN – 98.01%	Personal	-Study conducted on personal loans hence not suitable for mortgage assessment.

2.3 Theoretical Framework

Theoretical literature refers to research based on abstract ideas and concepts and is often used to develop or test theories. It often involves deductive reasoning, where general principles are used to derive specific predictions or explanations. Theoretical literature can include philosophical or theoretical papers, review articles, and conceptual models. Three predominant theories elucidate mortgage default phenomena: strategic default, characterized by negative equity as the driving force; cash-flow default, precipitated by adverse life events; and double-trigger default, necessitating the occurrence of both negative triggers.

2.3.1 Arbitrage Theory

The theoretical framework posits that the optimal investor within a specific asset market functions as a specialized arbitrageur (Gabaix, 2005). Consequently, the constraints encountered by this arbitrageur, such as capital limitations, manifest in asset pricing dynamics. This notion is examined within the context of the mortgage-backed securities market, where anecdotal evidence suggests a prevalence of specialized investors. Analysis reveals that risks deemed relatively inconsequential in the broader wealth landscape are indeed factored into pricing within the mortgage-backed securities market. A straightforward pricing model based on the aggregate value of mortgage-backed securities effectively integrates risk assessment within this market. Conversely, employing a pricing kernel based on aggregate consumption or wealth yields erroneous implications for pricing mortgage-backed securities risk. Hence, the findings align with the tenets of arbitrage theory, asserting that the marginal investor must possess specialized expertise in mortgage arbitrage.

2.3.2 Ruthless Default Theory

This theory posits that borrowers opt for immediate default as soon as the property value depreciates to match the mortgage value, a phenomenon termed "ruthless default," rather than waiting for further depreciation. The decision to default hinges upon comparing the current mortgage value to the present property value. At each decision juncture, borrowers must assess both values. While property value is readily observable, determining the mortgage value entails forecasting future scenarios, accounting for potential fluctuations in interest rates and property values, and discounting these outcomes to the present risk-free rate. This process involves weighing the likelihood of various outcomes and their respective impacts on financial well-being. According to the "ruthless" default model, borrowers default strategically to maximize their financial wealth. Negative equity serves as a prerequisite for default, but not a standalone determinant. Instead, a critical threshold of negative equity or house price exists, beyond which rational, wealth-maximizing agents opt for default, as elucidated in prior research by Kau, Keenan, and Kim (1994), among others. Notably, this theory presupposes borrowers' access to a flawless credit market for unsecured credit, rendering default decisions independent of liquidity constraints or income fluctuations.

2.3.3 Double Trigger Theory

This theory posits that negative equity serves as a pivotal precursor to default occurrences. However, it attributes default events to the confluence of negative equity and significant life events, such as unemployment. Widely acknowledged in mortgage research, the 16 double-trigger hypothesis is frequently discussed descriptively or through stylized models, as evidenced in works by Gerardi, Shapiro, and Willen (2007), Foote, Gerardi, and Willen (2008), and Foote, Gerardi, Goette, and Willen (2009), among others. Notably, this hypothesis has yet to be formalized into a structural dynamic stochastic model. In contrast, the frictionless theory, while prevalent, exhibits an over-reliance on fluctuations in aggregate house prices and tends to over predict the escalation of default rates. Conversely, the double-trigger hypothesis aligns more closely with empirical evidence. This alignment can be attributed to the economic rationale that default rates tend to escalate in tandem with the prevalence of negative equity among borrowers, as posited by the double-trigger theory. The disparity in predictions between the two theories underscores the significance of this finding, particularly within the scholarly discourse.

Moreover, it represents a pivotal stride toward the development of mortgage default models conducive to policy formulation and risk assessment, as such endeavours necessitate empirically sound frameworks.

2.3.4 Factors that Lead to Mortgage Default

Goodman et al. (2010) delve into the intricate dynamics that underlie the vexing issue of mortgage defaults, shedding light on the multifaceted forces that propel such occurrences. Within the realm of mortgage defaults, two compelling forces emerge, each wielding its own distinct influence on borrower behaviour, thus intricately weaving the narrative of this financial conundrum.

The first compelling force that Goodman et al. (2010) illuminate is the notion of liquidity scarcity. In this context, the homeowners find themselves ensnared in the treacherous grip of financial turmoil, where unforeseen and impactful shocks to their income or expenditures unravel their once-stable financial fabric. As a lamentable consequence, these homeowners grapple with a stark reality: the inability to meet their mortgage obligations. This force underscores the vulnerability of homeowners to external economic fluctuations, amplifying the delicate balance between financial stability and unforeseen adversities.

In parallel, Goodman et al. (2010) navigate the contours of another potent force: negative equity, an enigmatic phenomenon often colloquially dubbed as 'strategic default'. Here, the narrative takes a nuanced twist, as homeowners, despite retaining the financial capability to fulfil their mortgage commitments, choose an alternative path – that of default. This calculated decision stems from the precarious realm of negative equity, a realm wherein homeowners' mortgage obligations far outweigh the current market value of their abodes. In a strategic manoeuvre, these homeowners opt for default as a means of extricating themselves from the clutches of a home steeped in financial disparity. This strategic default, while ostensibly counterintuitive, unveils a complex interplay of financial calculus and homeowner agency, wherein borrowers make calculated decisions to navigate the intricate landscape of indebted homeownership.

2.3.4.1 Negative Equity

Negative equity, a term that strikes fear into the hearts of homeowners and economists alike, has long been a critical issue in the realm of housing finance (Barik, 2019). The intricate interplay between negative equity and mortgage default has been a subject of extensive study, spanning various disciplines such as economics, finance, and sociology. This complex dynamic has significant implications not only for individual homeowners but also for the broader stability of the housing market and the economy at large.

Negative equity arises when the outstanding balance of a mortgage surpasses the current market value of the property it secures. This precarious situation can arise due to a multitude of factors, including a decline in property values, economic downturns, or even overambitious borrowing. As homeowners find themselves trapped in a financial conundrum where their home is worth less than the debt they owe, a cascade of adverse consequences can unfold (Lee et al., 2012).

The link between negative equity and mortgage default is a profound one, rooted in both economic rationality and psychological distress. When a homeowner faces negative equity, the incentive to continue making mortgage payments diminishes. After all, why persistently pour funds into an asset that appears to be losing value? As financial strain mounts, a tipping point may be reached where default becomes an appealing option, leading to a downward spiral of missed payments and potential foreclosure (Bhutta et al., 2010).

Psychologically, negative equity can take a toll on homeowners, eroding their sense of financial well-being and stability. The emotional burden of feeling "underwater" on a mortgage can breed a sense of hopelessness, making the decision to default seem like an escape from an overwhelming situation. This emotional distress can be particularly pronounced when homeowners witness their neighbors or peers defaulting, creating a perceived social norm that further normalizes the behavior.

The contagion effect of negative equity-induced defaults can amplify the broader economic impact. As default rates rise, lenders face heightened risks, prompting them to tighten lending standards. This, in turn, restricts access to credit, making it harder for prospective homeowners to

enter the market. Reduced demand for housing can lead to further declines in property values, exacerbating the negative equity problem and perpetuating a vicious cycle (Foote et al., 2008).

Mitigating the link between negative equity and mortgage default requires a multi-faceted approach. Policy interventions, such as targeted loan modification programs, can provide homeowners with relief by adjusting the terms of their mortgages to align with current market conditions. These efforts aim to restore a sense of balance between the debt burden and the property's value, incentivizing continued payments and reducing the risk of default.

Financial education and counseling also play a crucial role in addressing the negative equity-default nexus. Empowering homeowners with a better understanding of their financial options and the long-term consequences of default can encourage informed decision-making. By promoting financial literacy, homeowners are better equipped to navigate the challenges posed by negative equity and make choices that align with their broader financial goals.

In conclusion, the intricate relationship between negative equity and mortgage default underscores the profound impact that housing market dynamics can have on individual behavior and broader economic outcomes. As homeowners grapple with the daunting prospect of owing more on their mortgages than their homes are worth, the temptation to default can become alluring. By untangling the web of negative equity and default, stakeholders can contribute to a more resilient and stable housing market for all.

2.3.4.2 Unemployment

The intricate web of socioeconomic dynamics reveals a profound link between unemployment and mortgage defaults, encapsulating a sphere where economic upheaval reverberates through households and financial systems. In a contemporary world characterized by interdependence, understanding the intricate dance between these two phenomena holds paramount importance. This discourse delves into the multifaceted nature of how unemployment orchestrates a symphony of mortgage defaults, intertwining economics, psychology, and policy frameworks.

At the heart of this relationship lies the vulnerability of individuals and families when income streams dissipate due to job loss (Lee et al., 2012). The narrative of financial stability begins to falter, posing imminent threats to housing payments. The phenomenon rests on a cascade of interconnected events: a job loss culminates in income reduction, which, in turn, snowballs into an inability to meet mortgage obligations. This complex interplay underscores the dire consequences that unemployment casts upon homeowners, bringing mortgage defaults to the forefront of the economic stage.

Psychological undercurrents also contribute significantly to this nexus. The psychological toll of unemployment weighs heavily on individuals, intertwining self-worth and financial capacity. A loss of employment often corrodes one's self-perception and induces stress, potentially leading to suboptimal decision-making. Coping with this emotional burden, homeowners might prioritize basic necessities over mortgage payments, inadvertently laying the groundwork for defaults. The intricate relationship between psychology and financial behavior underscores the need for holistic support systems during periods of unemployment (Lee et al., 2012).

The intricacies of unemployment's influence on mortgage defaults extend beyond national borders. Global economic interconnections render the phenomenon a universal concern. In an era of increasing mobility and cross-border employment, individuals might find themselves grappling with unemployment in a foreign land, exacerbating the complexity of the issue. The globalized nature of modern economies underscores the need for international cooperation in devising strategies to address this intersection.

The bond between unemployment and mortgage defaults is not solely dictated by financial arithmetic; it is a manifestation of human struggles, policy intricacies, and economic fluctuations. It underscores the necessity of a multidisciplinary approach to address its challenges. From economic policies aimed at job creation and stability to psychological support systems that alleviate the emotional burden of unemployment, the battle against mortgage defaults hinges on a comprehensive understanding of the dynamics at play.

The interwoven threads of unemployment and mortgage defaults paint a multifaceted portrait of vulnerability and resilience. This nexus is a testament to the profound impact of economic shifts on individual lives and societal structures. Unraveling this tapestry necessitates a holistic approach that combines economic insights, psychological understanding, and well-crafted policy frameworks. As we navigate the complexities of a rapidly changing world, acknowledging and addressing the interplay between unemployment and mortgage defaults stands as a pivotal endeavor, offering the promise of more secure financial futures for individuals and nations alike.

2.4 Models and Frameworks

To anticipate and mitigate the potential of mortgage defaults, scholars and experts have turned to the arsenal of machine learning methodologies. Among the array of models employed for this intricate task, certain stalwarts have emerged as cornerstones in predictive analyses.

2.4.1 Models

2.4.1.1 Random Forest Model

The random forest model stands as a machine learning ensemble technique tailored for both classification and regression tasks. It achieves this by assembling numerous decision trees and determining the most frequently occurring class (mode) for classification or the mean prediction for regression. Employing a random selection of features to partition the decision trees, the resultant classifier comprises multiple tree-based classifiers. The formulation of the random forest model can be encapsulated by the equation 2.1 presented below:

$$Space = \{F(X, \alpha i); i = 1,2,3,4, \dots\dots, n\ trees\}$$ *Equation 2.1 Random Forest Model Representation*

In equation 2.1, the variable "i" denotes the count of independent and identically distributed random vectors, with each tree contributing a vote towards the most prevalent category. Crafting the algorithm for this model entails randomly selecting "k" data points from the training set to form a decision tree. Subsequently, a predetermined number of trees (referred to as "ntrees") are chosen, and the aforementioned steps are iterated. When classifying each new data point, the

collective predictions of the "ntrees" are utilized to determine its category, with the majority vote dictating the classification. This iterative process commences with a single tree and progressively incorporates additional trees based on subsets of the data.

The primary advantage of the random forest lies in its ability to assess the significance of each variable by ranking their performance. This assessment entails estimating the predictive value of variables and subsequently perturbing them to gauge the resulting degradation in the model's performance.

In their study, Akindaini and Juhola (2017) employed a random forest model constructed from a random sample comprising one million observations, encompassing twelve predictors within a mortgage dataset. Utilizing the "random Forest" R package, they partitioned the data into a 60% training set and a 40% test set. This package facilitated the evaluation of variable significance and importance, with two key metrics: Mean Decrease in Accuracy (MDA) and Mean Decrease in Gini.

MDA quantifies the impact of omitting a particular variable from the model by measuring the percentage of misclassified observations. This metric is computed for each tree by permuting the out-of-bag (OOB) data and assessing the resulting prediction error, which is then averaged and normalized. Conversely, Mean Decrease in Gini assesses the increase in purity achieved by splitting a variable. A significant variable facilitates the separation of nodes into singular class nodes, enhancing model performance.

The model exhibited high performance in predicting both paying and prepaid classes, demonstrating sensitivity and specificity levels surpassing 90%. However, its performance in predicting the default class was moderate, with sensitivity marginally exceeding 50%. Overall, the random forest model demonstrated exceptional performance, achieving accuracy levels exceeding 95%.

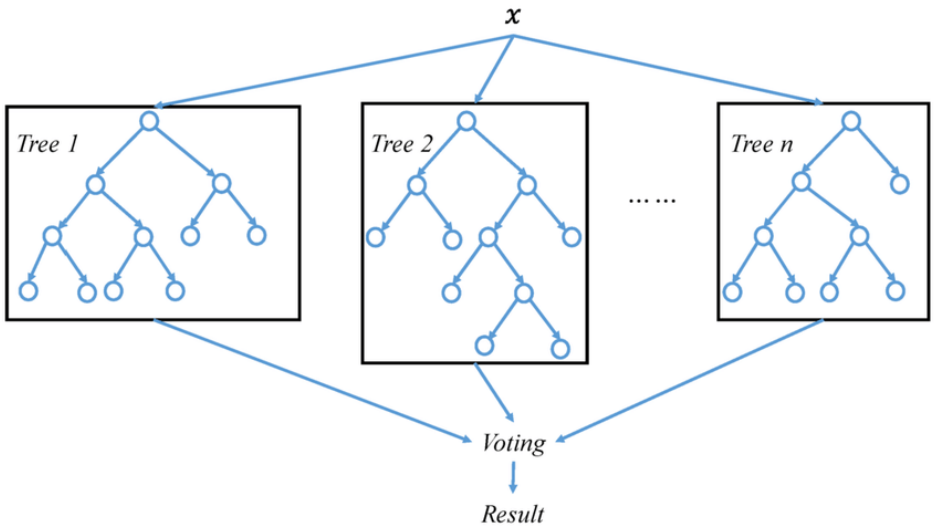


Figure 2.1 Architecture of Random Forest Model (Krauss, 2014)

2.4.1.2 KNN Model

The K Nearest neighbor classifier, often referred to as KNN, exemplifies a nonparametric statistical model, devoid of explicit presumptions regarding parameter form or distribution. Operating on a basis of distance, KNN determines outcomes by taking a majority vote from the k nearest data points. Diverse distance metrics, such as Euclidean, Manhattan, Chebyshev, and Hamming distances, can be employed in the KNN model. For the present study, we exclusively utilize the Euclidean distance measure.

The K-nearest neighbor (KNN) algorithm operates as follows: upon receiving parameters including a positive integer K, a specified distance metric d, and an unknown observation x, the model executes the following steps:

a) The algorithm proceeds through the entire training set, assessing the distance d between x and every data point contained within it. It discerns the K data points nearest to x, constituting the set W, with K chosen as an odd number to avert any potential ties.
b) Following this, the algorithm proceeds to calculate the percentage of points within the designated set W that correspond to a particular class label. This calculation yields the conditional probability assigned to each class, as outlined in equation 2.2:

$$P(y = i|X = x) = 1 / K \sum I(y(j) = i)\, W\, j \in W$$ *Equation 2.2 KNN Model*

In the above equation, I function as an indicator that yields 1 when true and 0 when false.

c) Ultimately, x is assigned to the class with the highest probability. The choice of K is of paramount importance in this context. Within the framework of K-Nearest Neighbors (KNN), K acts as a hyper parameter dictating the shape of the decision boundary, thus requiring meticulous tuning to ensure optimal alignment with the dataset. A smaller K constricts the prediction region, inducing higher variance and lower bias. In contrast, larger K values encompass a greater number of neighbouring points, leading to a smoother decision boundary characterized by reduced variance but heightened bias.

It is worth highlighting that the training phase of the K Nearest Neighbors (KNN) algorithm entails both memory and computational expenses. Memory cost arises due to the necessity of storing an extensive dataset, as the algorithm essentially memorizes training observations for classification. Consequently, the algorithm relies solely on these training observations to generate predictions when queried. Predicting the class of an individual observation demands traversing the entire dataset, thus introducing computational cost as a significant factor.

Akindaini and Juhola (2017) put the KNN classifier into practice in their study. Utilizing a randomized dataset comprising 120,000 data points, partitioned into 80,000 for training and 40,000 for testing, researchers embarked on the creation of 50 distinct K-Nearest Neighbors (KNN) models, with K values ranging from 1 to 50. Evaluation of each model's accuracy entailed predicting mortgage defaults within the test dataset. Impressively, the KNN model demonstrated robust performance in forecasting both paying and prepaid classes, yielding positive predicted values of 76% and 94% respectively. However, its efficacy in predicting the default class proved less formidable. Despite exhibiting a positive predicted value of 56%, slightly surpassing the average threshold, the model's performance in this regard was notably less pronounced.

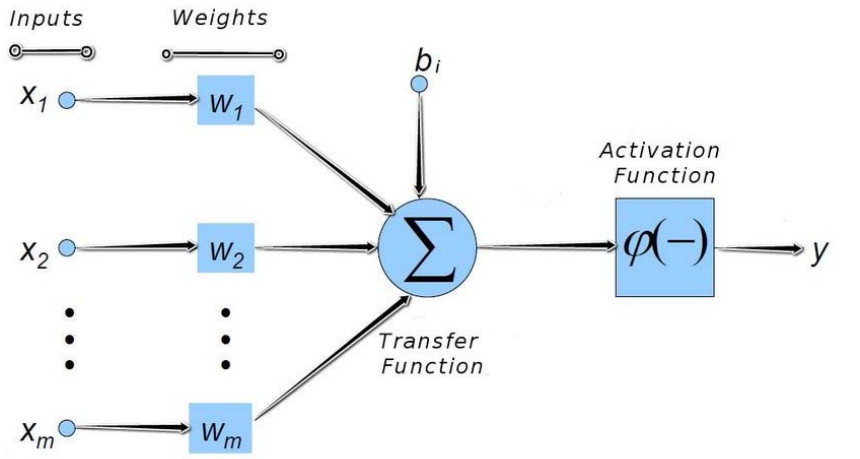


Figure 2.2 KNN Model Architecture (Kutlu & Turan, 2018)

2.4.1.3 Logistic Regression

Logistic regression stands as a versatile technique within the realm of generalized linear models, facilitating the prediction of discrete outcomes. In logistic regression, the response variable assumes the form of a Bernoulli variable, capable of assuming the value 1 with a probability of success represented by θ, or the value 0 with a complementary probability of failure, denoted as 1-θ. In the context of credit risk analysis, let us delineate a random variable D, taking on values of 1 and 0, where D = 1 signifies a defaulted loan, while D = 0 indicates a non-defaulted loan. Thus, the probability of default corresponds to the probability of success for the random variable D, expressed as $\theta = P(D = 1)$.

2.4.1.4 Naïve Bayes

The inception of Naive Bayes can be traced back to the 1950s, where it emerged as a foundational probabilistic classifier known by various names, including simple Bayes and independence Bayes. Operating on the principles of the Bayes theorem, this classifier hinges on the assumption of attribute independence given the knowledge of class. Within a broader framework, Naive Bayes embodies a conditional probabilistic model, often employing decision rules such as the maximum a posteriori (MAP) rule.

Consider a scenario within supervised learning wherein the objective revolves around approximating a target function P(X/Y). This formulation can be represented in the Bayes format as follows:

$$P(Y = yi\,|x1, x2, \dots, xj) = P(X = x1, x2, \dots, xj \mid Y = yj) * P(Y = yi) \sum P(X = x1, x2, \dots, xj \mid Y = yj) * P(Y = yi)\; j\; 1$$

Equation 2.3 Naïve Bayes Format

Focusing on the outcome of the classification task, the ultimate objective is to assign the instance Y the class with the highest probability. This entails computing:

$$Y \leftarrow arg\ max\ P(Y = yi) \prod P(xi\,|Y = yi)\; j.$$

Equation 2.4 Probability Computation

2.4.1.5 Convolutional Neural Networks (CNN)

CNN represents a biologically inspired variant of traditional neural networks. It operates akin to standard neural networks but distinguishes itself through its ability to harness three-dimensional neurons, facilitating robust feature extraction from input data. Structurally, CNNs comprise three main components: convolutional layers, pooling layers, and fully connected layers, each serving distinct purposes:

i). Convolution layers: These layers constitute the core of CNNs, where input data undergoes feature extraction using three-dimensional neurons.

ii). Pooling layer: Positioned after convolution layers, the primary function of the pooling layer entails down-sampling spatial dimensions derived from the convolutional layer, thereby reducing data complexity.

iii). Fully connected layers: Serving as the final stage of CNN architecture, these layers generate target outputs. In CNNs, propagation is unidirectional, with one or more convolutional layers linked to subsequent pooling layers. Nonetheless, bidirectional propagations occur as outputs from convolution layers are transmitted to fully connected layers within the neural network.

2.4.2 Frameworks

2.4.2.1 TensorFlow

TensorFlow stands as an open-source framework pivotal for numerical computation via data flow graphs (TensorFlow, 2018). Developed and meticulously maintained by the Google Brain team within Google's Machine Intelligence research division, TensorFlow caters to the realms of Machine Learning (ML) and Deep Learning (DL). Its current iteration operates under the Apache 2.0 open-source license.

The framework is meticulously crafted for facilitating large-scale distributed training and inference tasks. Within its architecture, nodes within the graph represent mathematical operations, while the graph edges symbolize the flow of multidimensional data arrays (tensors) between these operations. The distributed TensorFlow framework encompasses distributed master and worker services, each equipped with kernel implementations. These kernels encompass a broad spectrum of standard operations—approximately 200 in total—including mathematical, array manipulation, control flow, and state management operations, predominantly coded in C++.

TensorFlow's versatility extends across the spectrum from research and development environments to production systems. It exhibits compatibility with a diverse range of computing setups, accommodating single CPU systems, Graphics Processing Units (GPUs), mobile devices, and extensive distributed systems spanning hundreds of nodes.

Furthermore, TensorFlow Lite emerges as a lightweight variant tailored for mobile and embedded devices (TensorFlowLite, 2018). It facilitates on-device ML inference characterized by low latency and a compact binary size, albeit with a constrained set of operator coverage. Moreover, TensorFlow Lite leverages hardware acceleration through the Android Neural Networks API, enhancing its performance on mobile platforms.

TensorFlow boasts an array of programming interfaces, primarily encompassing APIs for Python and C++, with ongoing developments aimed at extending support to Java, GO, R, and Haskell.

Notably, TensorFlow finds robust support within prominent cloud environments, including those offered by Google and Amazon.

2.4.2.2 Keras

Keras, the dynamic Python wrapper framework, emerges as a potent conduit that establishes seamless connections to a kaleidoscope of cutting-edge Deep Learning (DL) tools. Among these, it forges strong alliances with industry stalwarts such as TensorFlow, CNTK, and Theano. Moreover, it eagerly embraces the burgeoning potential encapsulated within its beta version, weaving a harmonious integration with MXNet. A nod to the future can be seen in its roadmap, with the promise of integration with Deeplearning4j announced within the expansive Keras framework (Keras 2018).

At its core lies an unwavering commitment to catalyzing swift and decisive experimentation. Keras unfurls its wings under the sheltering canopy of the MIT license, a symbol of freedom that resonates through every line of its meticulously crafted code. From the steadfast embrace of Python 2.7 to the forward-facing momentum of Python 3.6, Keras remains an unwavering presence, a steadfast companion across the arc of Python's evolution.

In the grand symphony of hardware, Keras conducts a harmonious ballet, effortlessly gliding between the resounding cadence of Graphics Processing Units (GPUs) and the measured rhythm of Central Processing Units (CPUs). It does so while remaining in perfect resonance with the chosen backdrop of the underlying DL frameworks. This symposium of compatibility, efficiency, and openness paints Keras as a spellbinding instrument—one that not only erects bridges to the future but also navigates the diverse landscapes of today's computational prowess with consummate finesse.

2.4.2.3 PyTorch

PyTorch, a robust and dynamic Python framework designed for accelerating deep learning computations on GPUs (PyTorch 2018), stands as a testament to the power of seamless

integration between the convenience of Python and the optimization capabilities of C libraries, in the same vein as its predecessor, Torch. Emerging under the nurturing guidance of Facebook's AI research group in 2016, PyTorch emerges as an intricate fusion of Python, C, and CUDA, leveraging a confluence of programming languages to deliver an unparalleled deep learning experience. Anchored at its core is a symphony of optimization libraries, including Intel MKL and NVIDIA's cuDNN and NCCL, playing in unison to orchestrate an eloquent ballet of acceleration.

The essence of PyTorch is encapsulated within its manipulation of tensors and neural networks, supported by the orchestration of a tape-based autograd system, akin to a finely tuned orchestra conductor guiding its performers. This autograd symphony harmonizes tensor computations with vigorous GPU acceleration, breathing life into complex architectures as effortlessly as a maestro bringing forth melodies from a well-tuned instrument. The secret behind this melodic transformation lies in PyTorch's adept use of reverse-mode auto-differentiation, a technique akin to a composer's pen, allowing a network's behaviour to be elegantly reimagined with the mere stroke of logic. The result: a dynamic computational graph (DCG) that readily adapts to the evolving narrative of the data.

Inspired by the virtuosos of autograd and Chainer (Chainer 2018), PyTorch finds itself bridging the chasm between the scientific and industrial realms. Within the grand stage of innovation, Uber engineers masterfully crafted Pyro, a universal probabilistic programming language that takes PyTorch as its backend, casting probabilistic incantations that dance on the edges of the unknown. Fast.ai, the trailblazers of deep learning education, have cast their vote of confidence, migrating their courses to the PyTorch ecosystem, anointing it as the canvas upon which future deep learning maestros shall paint.

In the realm of digital landscapes, PyTorch stands unwavering, a flagbearer of open-source prowess, unfurling the banner of its BSD license to foster a collective symphony. Amongst its patrons stand the titans of the virtual world: Facebook, Twitter, NVIDIA, and a pantheon of other organizations, united in their reverence for PyTorch's harmonious serenade. As it strides forth, PyTorch encapsulates the spirit of a dynamic sonnet, resonating through scientific

laboratories, industrial citadels, and innovation sanctuaries, etching its melodious tale across the annals of deep learning history.

2.5 Architectures and Designs

2.5.1 CNN Architecture

Zhou et al. (2020) employed a convolutional neural network (CNN) framework in their investigation to develop a predictive model for personal credit default. Assessing the model's efficacy entailed two key metrics: accuracy (ACC) and the area under the receiver operating characteristic (ROC) and area under the curve (AUC). Results from the experimentation showcased a remarkable accuracy level, denoted as ACC, exceeding 95%. Moreover, the area under the receiver operating characteristic (ROC) curve, designated as AUC, surpassed the 99% threshold. Notably, the model's performance demonstrated a notable superiority over conventional algorithms such as support vector machine (SVM), Bayes, and random forest (RF). The CNN architecture utilized in the study is illustrated in Figure 2.1.

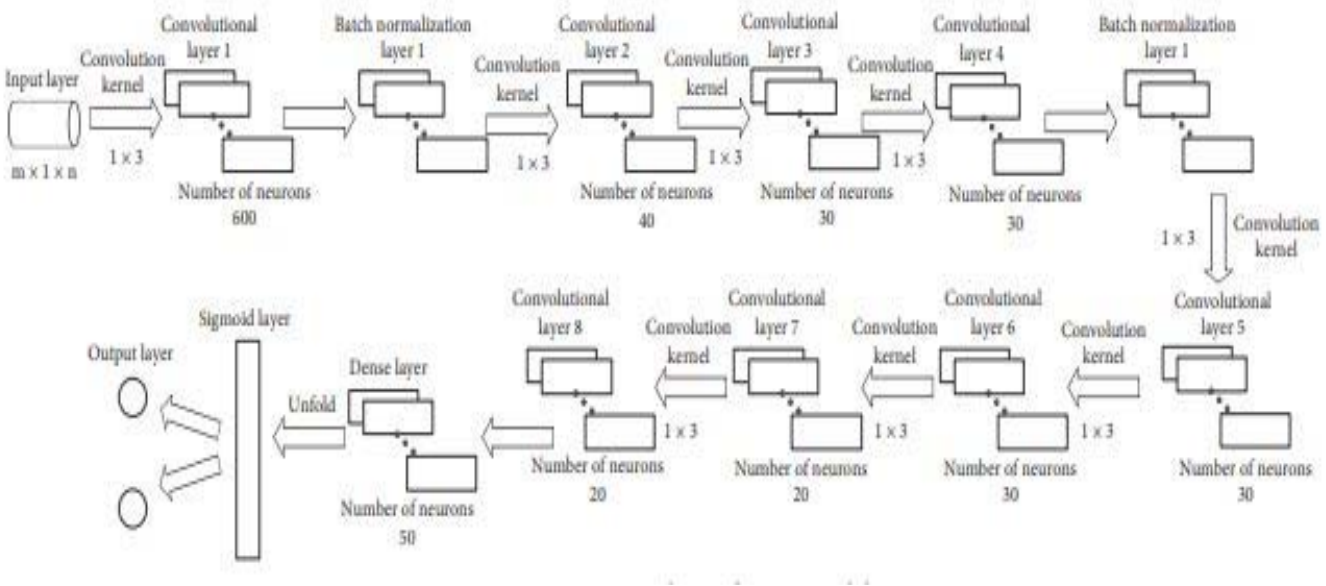


Figure 2.3 CNN Architecture (Zhou et al, 2020)

2.5.2 Bayesian Inference Architecture

Zaharieva and Ignatov (2019) adopted the Bayesian inference architecture in modelling of default mortgage portfolios. The pre-existing beliefs concerning the unknown parameter vector β within a single model are encapsulated by the probability density function π(β). Meanwhile, the likelihood function is denoted as π(y|β). The amalgamation of the likelihood function and the prior beliefs yields a proportionate representation of the posterior distribution of the parameters, conditioned on the available data as shown below:

$$\pi(\beta|y) = \frac{\pi(y|\beta)\pi(\beta)}{\int \pi(y|\beta)\pi(\beta)d\beta} \propto \pi(y|\beta)\pi(\beta) \quad \textit{Equation 2.5 Bayesian Inference}$$

The equilibrium within lies in the application of the law of total probability, denoted as π(y) = Z π(y|β)π(β)dβ. Termed as the marginal likelihood, it serves as the linchpin for Bayesian model comparison and, consequently, variable selection. The essence of proper density functions, ensuring integration to 1, is pivotal for unbiased model comparisons. Deriving the marginal likelihood allows for the transformation of the likelihood function multiplied by a proper prior into a suitable density function. Figure 2.4 summarizes the Bayesian inference architecture.

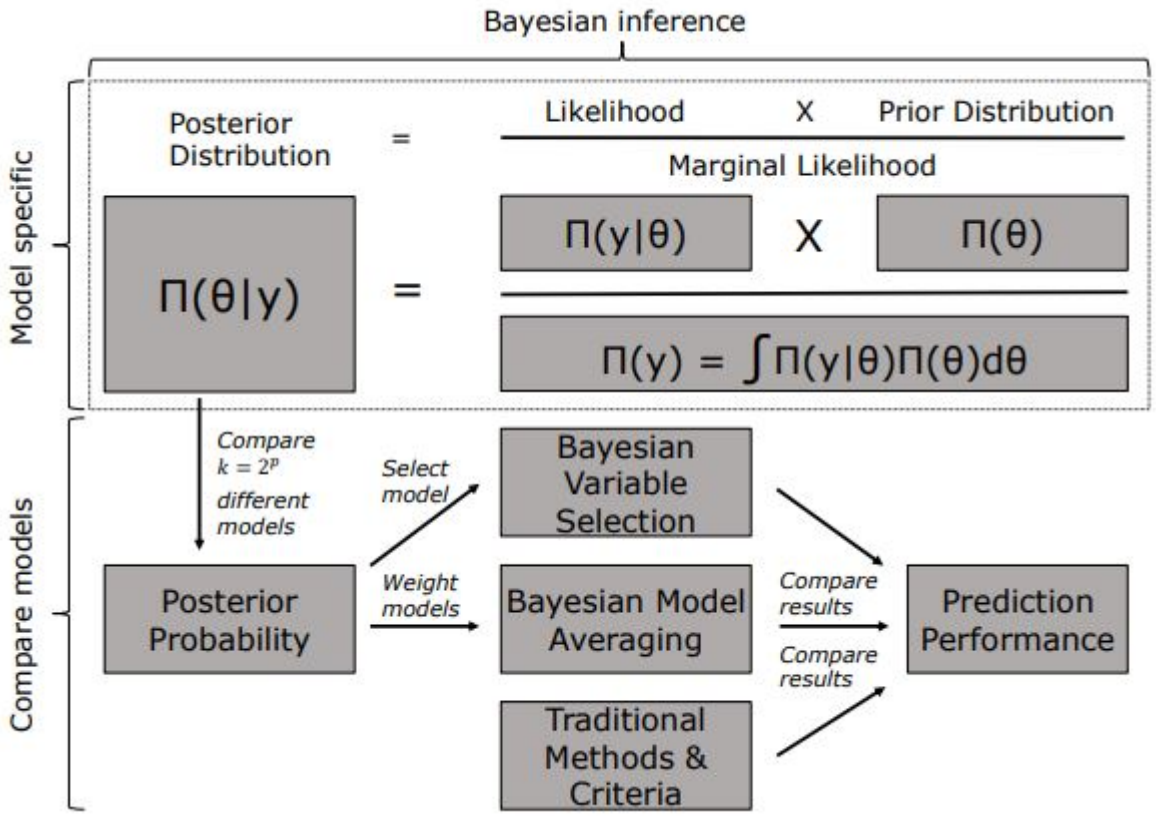


Figure 2.4 Bayesian Inference Architecture (Zaharieva and Ignatov, 2019)

Mortgage default prediction architectures and designs are summarized in Table 2.2.

Table 2.1 Architectures Summary

Author	Architecture	Strengths	Limitations
(Zhou et al., 2020)	CNN Architecture	The architecture incorporates a Convolutional Neural Network (CNN) model, designed to autonomously discern and map relationships between input and output pairs. This	This architecture demands significant computational resources.

		capability is achieved through exposure to a substantial volume of known data pairs, enabling the model to learn intricate patterns without requiring precise mathematical expressions delineating the input-output relationship.	
(Zaharieva and Ignatov, 2019)	Bayesian Inference Architecture	-The architecture compares various models used for predicting mortgage default.	-The architecture relies on the Naïve Bayes algorithm which is sensitive to irrelevant features, potentially diluting predictive accuracy.

2.6 Algorithms

2.6.1 Machine Learning

Machine learning stands as a subset within the expansive realm of artificial intelligence, characterized by machines' ability to emulate intelligent human behaviour (Sarker, 2021). These algorithms operate by ingesting and analyzing data to discern underlying patterns across various domains, encompassing individuals, business processes, transactions, and events. Categorically, machine learning algorithms are broadly classified into four distinct types: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning.

2.6.1.1 Supervised Learning

In machine learning, supervised learning encompasses the task of training models to discern patterns that map input data to corresponding outputs, drawing insights from labelled training datasets. It operates under a task-driven paradigm, wherein specific objectives guide the learning process. The primary supervised learning tasks include classification, which involves segregating

data into distinct categories, and regression, which entails fitting data to predictive models. For instance, predicting the sentiment or class label of textual data, such as tweets or product reviews, exemplifies the application of supervised learning techniques (Sarker, 2021).

2.6.1.2 Unsupervised Learning

Unsupervised learning, a data-driven process, entails the analysis of unlabelled datasets devoid of human intervention. This approach finds extensive utility in extracting generative features, discerning meaningful trends and structures, identifying groupings in results, and facilitating exploratory endeavours. Predominantly, unsupervised learning encompasses tasks such as clustering, density estimation, feature learning, dimensionality reduction, association rule discovery, anomaly detection, among others (Sarker, 2021).

2.6.1.3 Semi Supervised Learning

Semi-supervised learning represents a synthesis of the supervised and unsupervised methodologies, effectively harnessing both labelled and unlabelled data (Sarker, 2021). Positioned between the realms of "unsupervised" and "supervised" learning, it navigates the terrain of limited labelled data juxtaposed against abundant unlabelled data, offering practical utility in various real-world scenarios. The core objective of semi-supervised learning lies in surpassing the predictive efficacy attainable solely through reliance on labelled data. Noteworthy application domains where semi-supervised learning finds application encompass diverse fields such as machine translation, fraud detection, data labelling, and text classification.

2.6.1.4 Reinforcement Learning

Reinforcement learning stands out as a pivotal branch within machine learning, empowering software agents and machines to autonomously discern optimal behaviours within specific contexts or environments, thereby enhancing efficiency through an environment-centric approach. This learning paradigm hinges on the principles of reward and penalty, striving to leverage insights gleaned from environmental interactions to drive actions aimed at maximizing rewards or mitigating risks. Reverberating with potential, reinforcement learning serves as a potent instrument for training AI models poised to elevate automation levels and streamline the operational efficacy of intricate systems, spanning domains like robotics, autonomous driving,

manufacturing, and supply chain logistics. However, its utility is less favoured for addressing rudimentary or straightforward challenges.

2.6.2 K Nearest Neighbor

The K-nearest neighbors (KNN) algorithm functions by classifying incoming observations based on the target values of their nearest neighbors within the feature space. Proximity, a pivotal element of this algorithm, is shaped by its hyperparameters. For instance, specifying the number of neighbors, such as setting it to 10, dictates the classification process. Here, the new observation is assigned to the class prevalent among its closest 10 neighbors. Typically, the algorithm adopts a weighted scheme, assigning greater significance to the nearest neighbors. Moreover, the concept of "distance" often relies on the Euclidean distance metric. Determining the optimal number of neighbors, a crucial parameter influencing the algorithm's efficacy, entails empirical exploration involving experimentation with diverse values.

2.6.3 Gradient Boosting

The Gradient Boosting algorithm endeavours to approximate a weighted function utilizing weaker classifiers, notably Decision Trees, with the aim of minimizing the loss function. Initiated with arbitrary weights, the algorithm proceeds iteratively in a "greedy" manner. A plethora of Gradient Boosting algorithms draw inspiration from the recursive partitioning algorithm delineated by Friedman (2001) and Friedman (2002), employing Decision Trees as the primary weak classifiers, thereby adopting a strategy termed "Tree boosting." This methodology entails the generation of a sequence of decision trees from data samples (SAS, 2017). The requisite hyperparameters stem from the aforementioned approach. For instance, the parameter "13 Iterations" denotes the number of trees to be grown, while "Train proportion" signifies the percentage of data allocated for training each tree. Moreover, additional hyperparameters pertinent to Decision Trees, such as "Maximum branch" and "Maximum depth," necessitate specification, as elucidated earlier.

2.6.4 Convolutional Neural Network

Convolutional Neural Networks (CNNs) constitute a subset of deep learning models utilized for the analysis of gridded data, particularly images. They are engineered to autonomously and dynamically discern spatial hierarchies of features, progressing from rudimentary to intricate patterns (Yamashita et al., 2018). A typical CNN architecture encompasses three distinct layers: convolution, pooling, and fully connected. These layers serve as mathematical constructs whereby convolution and pooling layers extract features, while fully connected layers facilitate mapping these features to the ultimate output, such as classification. The graphical representation of a CNN is depicted in Figure 2.5.

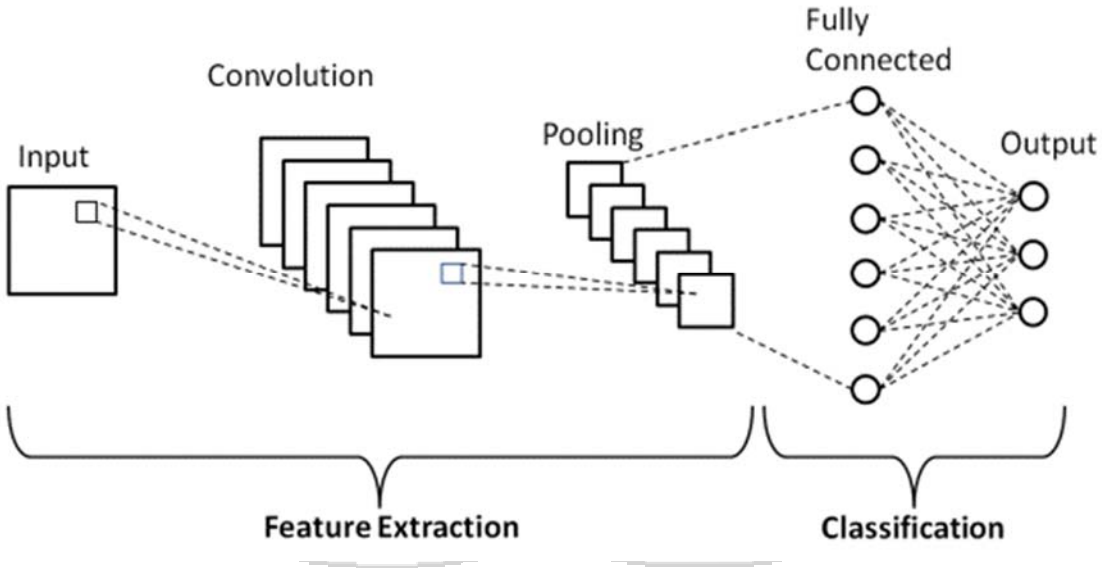


Figure 2.5 Structure of CNN (Gurucharan, 2020).

Table 2.2 shows the summary of the algorithms that have been used predicting mortgage default.

Table 2.2 Summary of Algorithms used in Mortgage Default Prediction.

Algorithm	Strengths	Limitations
K Nearest Neighbor	-KNN leverages the proximity of data points to make predictions. In the context of mortgage default predictions, this implies that the algorithm can identify patterns and trends specific to certain neighbourhoods or regions, accounting for localized economic conditions and property market dynamics. - KNN operates as a non-parametric method, refraining from assuming any specific underlying distribution of the data. This attribute is particularly advantageous when dealing with complex and evolving socioeconomic variables that might not adhere to traditional statistical distributions.	-KNN entails an inherent computational load, as it necessitates the calculation of distances between data points to determine neighbors. This characteristic can hinder scalability and real-time processing, potentially impeding the algorithm's efficiency with large datasets. -Mortgage default datasets often exhibit class imbalances, where instances of default are considerably fewer than non-default instances. KNN's reliance on neighbouring points can result in biased predictions towards the majority class, diminishing its ability to predict rare events accurately.
Gradient Boosting	-Gradient boosting excels in capturing intricate relationships between predictors and mortgage defaults. It iteratively constructs an ensemble of weak learners, refining predictions at each step, culminating in a robust and accurate model.	-Without proper regularization, gradient boosting can overfit smaller mortgage datasets, leading to poor generalization to new data. - Noisy or irrelevant features in the mortgage dataset can mislead the boosting process, impacting prediction quality.
Convolutional Neural Networks	-CNNs excel in automatically extracting intricate patterns and	- The high complexity of CNNs can lead to overfitting, especially with

	features from raw data, enabling them to capture complex relationships within mortgage datasets that might be overlooked by traditional models. -	smaller mortgage datasets. Rigorous regularization techniques are necessary to mitigate this risk.

2.7 Gaps in the Existing Systems

A significant portion of existing research revolves around heuristic and manually crafted feature design. The task of identifying suitable features or covariates for mortgage evaluation poses a challenge for human decision-making. Hence, the primary objective of this research entails the utilization of a convolutional neural network, systematically harnessing its capabilities to automate the process of feature engineering.

In the context of mortgage prediction, there exists a noticeable scarcity of research within the Kenyan domain, which thereby presents a compelling avenue for exploring this specific domain. Interestingly, this gap in knowledge serves as an invitation to delve into uncharted territory and contribute meaningful insights. Within the domain of credit scoring, the focus has often been misplaced, with a majority of studies failing to introduce additional explanatory variables. Rather than scrutinizing the potential benefits of incorporating novel explanatory variables, researchers frequently direct their attention toward comparing distinctions among various scoring models. This approach overlooks a crucial aspect of credit assessment.

Moreover, a substantial number of prior studies have neglected the development of a tool accessible to consumers themselves. The lack of emphasis on creating a consumer-oriented tool has left a void in empowering individuals to actively engage with and comprehend the intricacies of credit scoring systems. The prevailing research landscape is ripe for a shift towards automated feature engineering through advanced neural networks, an exploration of underexplored terrains such as mortgage prediction in Kenya, and a reorientation of focus towards comprehensive credit scoring models and user-friendly tools.

2.8 Conceptual framework

The mortgage dataset underwent preprocessing, ensuring its suitability for training of the model. The preprocessing phase, encompassing data cleaning and transformation, elevates the dataset's quality, aligning it with subsequent stages of the modeling process. To establish a robust training and evaluation setup, the dataset was partitioned into two subsets: 80% for training and 20% for testing. In the feature extraction stage, automated feature engineering is deployed alongside Convolutional Activation and Max Pooling layers. These layers enhance the model's ability to discern patterns within the mortgage data. Convolutional Activation layers apply non-linearities to the convolution operation, allowing the model to capture complex relationships in the data. Max Pooling layers then down-sample the spatial dimensions, preserving the most relevant information. The feature extraction process leverages the power of CNN's hierarchical feature learning. This allows the model to capture nuanced relationships within the mortgage dataset, enhancing its ability to discern intricate patterns and representations. Moving to the classification stage, Fully Connected layers and Softmax activation are employed. These layers facilitate the learning of complex interactions between features extracted in the previous stage. Fully Connected layers connect every neuron in one layer to every neuron in the next layer, enabling the model to capture intricate relationships within the feature space. Softmax activation is applied for multi-class classification, providing probabilities for each class. The conceptual framework's objectives remain dual-fold: predicting mortgage defaults with precision, aiding risk assessment, and providing informed recommendations for optimal mortgage amounts. This framework, integrating Convolutional Activation, Max Pooling, Fully Connected layers, and Softmax activation, stands at the forefront of advanced technology in mortgage risk management, contributing to the evolution of predictive analytics in the financial sector.

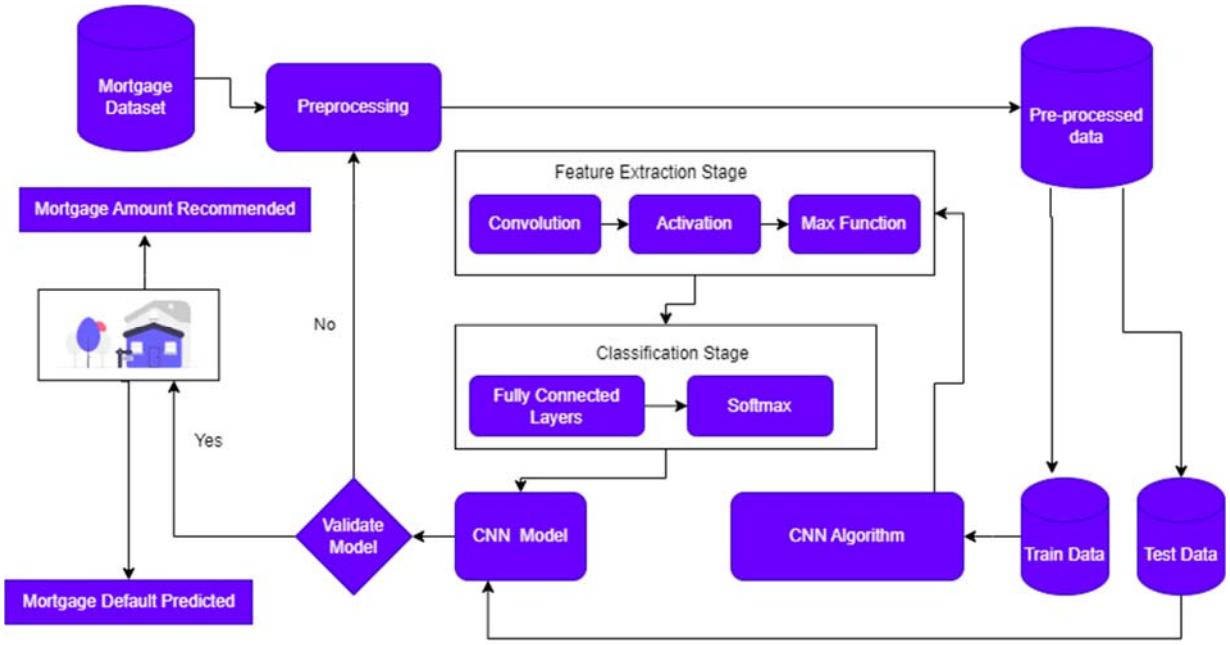


Figure 2.6 Conceptual Model

Chapter 3: Research Methodology

3.1 Introduction

Research methodology refers to studying different methods and analysing how the available methods can be applied to solving a particular problem in specific research (McConnell, 2010). This section describes the research design and philosophy used during the research. In addition, the section elaborates on the development method that will be used in the latter stages of systems development. The agile methodology was preferred for the systems development because it leads to creating a user-centered system. The section also explains the population and sampling techniques to be used during the study. Furthermore, the strategies used to ensure the research results' quality and reliability are explained in detail. Lastly, ethical considerations during the research are also presented.

3.2 Research Design and Philosophy

This research used applied research, focusing on finding solutions to real-world issues rather than expanding one's knowledge base (Lawrence Berkeley National Laboratory, 2014). The first step was a detailed discussion and definition of the research problem. The researcher examined literature that discussed the recommended solutions in terms of concepts and theories, and empirical studies similar to the ones the researcher was conducting. The knowledge gained from this helped clarify the nature of the issue and formed the basis for creating potential remedies. The right questions were asked, the data was analyzed properly, and reasonable conclusions were drawn.

3.3 Population and Sampling

3.3.1 Population

A research population refers to the huge number of individuals or entities that are the focus of a study. The target population for this study is the mortgage borrowing data from lending institutions in Kenya.

3.3.2 Sampling

A sample serves as a representative subset encompassing all qualities or characteristics present within a population. The primary aim of sampling is to acquire precise empirical data while

minimizing costs compared to examining every possible case. Utilizing a sizable sample size diminishes sampling variability and mitigates the likelihood of biased sampling. In the context of model development, 80% of the data was allocated for training purposes, while the remaining 20% was reserved for testing and validation. The utilization of an 80% portion of data for training and a 20% subset for testing in machine learning methodology is underscored by its solid foundation in empirical research across diverse disciplines. By setting aside a sizable testing subset, the efficacy and adaptability of models to real-world scenarios are rigorously assessed. This training-to-testing ratio, long validated in multifaceted reviews and meta-analyses, ensures models' generalization and applicability, enhancing their utility for practical implementation.

3.4 Data Collection Methods and Analysis

3.4.1 Data Collection

This study used secondary data obtained from Kaggle. This is because lending institutions are very strict with providing their consumer data.

3.4.1.1 Model Construction

The model construction process encompassed several key steps. It commenced with the collection of the HMEQ dataset from Kaggle, followed by data pre-processing to ensure its suitability for subsequent analysis. Subsequently, exploratory analysis was conducted to gain insights into the essential characteristics of the dataset's entities. The subsequent steps included model training and testing, performed on both the training and testing subsets.

3.4.1.2 Data Pre-Processing

The initial requirement comprised obtaining a dataset in CSV format. Prior to its utilization in model development, the data necessitated pre-processing, encompassing the following sequential steps:

i. Feature selection

Many techniques, such as feature selection and dimensionality reduction, are employed to enhance machine learning models' efficacy. In this research, we'll build five different training sets, each with its own distinct set of characteristics. T1 is the complete set, and the subsequent

sets are all subsets of T1. T2 relies on attributes with a strong correlation to default. Characteristics with significant positive correlations to default are located in T5, while features with high correlations to other features in the core set are located in T3.

ii. Feature scaling

In most mortgage datasets, features have multiple scales. A big scale gap between features frequently slows down optimization procedures. Feature scaling may aid in improving classification performance and learning efficiency in various ML algorithms. Following the data conversion stage, standardization and normalization will be conducted on the original data.

iii. Data Cleaning

The process involved eliminating statistically insignificant variables, such as the ID variable, as well as instances with missing or undefined values for categorical variables, as outlined in Section 5.3.1. Additionally, outlier removal was conducted in accordance with the guidelines provided in Section 5.3.2.

3.4.1.3 Model Training

To train and evaluate the model, the data was split into training and validation sets using the train-split-test function from the Scikit-learn library. This ensured that the model was trained on a subset of the data and evaluated on a separate subset to assess its performance. The input data was reshaped as needed to match the expected input shape of the model, which included converting the DataFrame to a NumPy array and reshaping it accordingly.

Augmentation techniques such as random shift and noise addition were applied to the training data to enhance model generalization and robustness. The augmented data was then used to train the model using the fit function, specifying the number of epochs and batch size. To experiment with different batch sizes, a loop was implemented, iterating over a predefined range of batch sizes. Within each iteration, a Convolutional Neural Network (CNN) model architecture was constructed using the Sequential API from Keras, comprising convolutional and pooling layers followed by fully connected layers. The model was compiled with the Adam optimizer and binary cross-entropy loss function. The training process was performed for each batch size, and the model's performance was evaluated on the validation set using the validation data argument.

3.4.1.4 Model Testing and Evaluation

The model was evaluated on the test set using the evaluate function, which computed the test loss and accuracy metrics. These metrics provided insights into the model's generalization and effectiveness in classifying the test data. The model achieved a top accuracy of 97.14%.

3.4.2 Data Analysis

In the data analysis phase, attention was dedicated to the process of data cleaning and transformation. This crucial step aimed to prepare the raw data, ensuring its quality and compatibility for the subsequent stages of model development. Just as a skilled craftsman refines raw materials before crafting a masterpiece, data analysis involved refining and organizing the data to enable the creation of robust and accurate models. Furthermore, the journey through data exploration was undertaken, delving into the depths of the dataset's nuances to unearth hidden insights and patterns. This exploratory analysis acted as a compass to guide the researcher through the uncharted territories of information, revealing potential variables, relationships, and outliers that might hold the key to unravelling the core questions of our research.

3.5 Research Quality and Reliability

Reliability, or the degree to which results are stable over time, is the primary yardstick by which studies are evaluated for quality (Goodfriend, 2015). The second requirement for good research is validity, meaning studies must provide plausible explanations for their findings. All of the criteria mentioned above are met in this study thanks to its well-thought-out topic, defined objectives, focused research questions responsive to the literature review, lack of research bias, appropriate analytical methods, and logical, consistent conclusions and recommendations based on the study's findings.

3.6 Systems Development Methodology

Agile methodology was used as the systems development methodology. Agile methodology represents an iterative approach to project management and software development, aiming to expedite the delivery of value to consumers (Shankarmani et al., 2012). An agile team spreads feature releases over a longer period rather than risking everything on a single large release.

Teams have an inbuilt mechanism for responding to changing conditions thanks to constant evaluation of requirements, plans, and results. Figure 3.1 shows the process flow used by the Agile methodology.

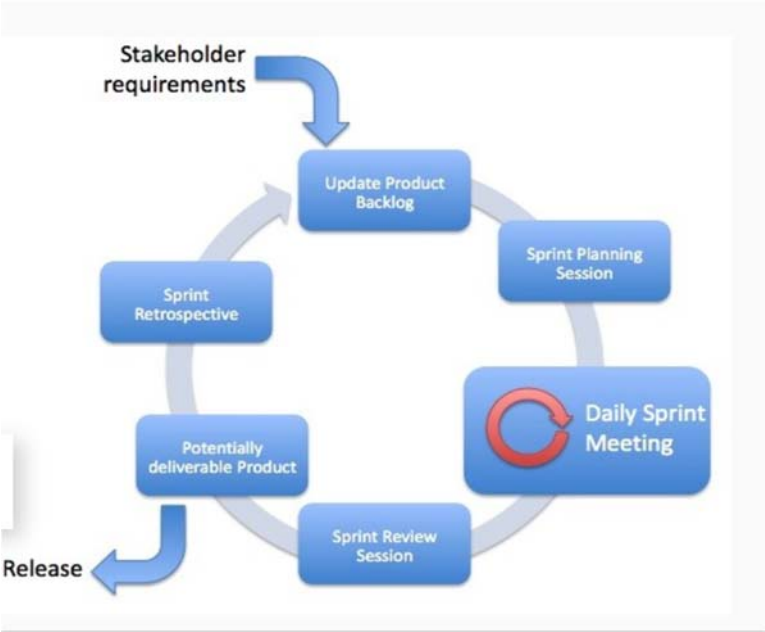


Figure 3.1 Agile Methodology (Martin, 2019)

3.6.1 Stakeholder Requirements

The project's product requirements are the stakeholders' needs. A new software development project's owner is the ultimate decision-maker. Beginning the requirement engineering process is one of the duties. It entails documenting the most important use cases and conversing about them with user representatives and other interested parties. The Agile development team needs these specifications.

3.6.2 Update Product Backlog

A product backlog was created, listing all the tasks and deliverables required for the thesis. The research, data collection, analysis, and writing tasks was broken down into manageable units.

3.6.3 Sprint Planning Session

Work was planned in time-boxed iterations called sprints. The length of each sprint was determined based on the project's timeline and complexity. During sprint planning, tasks from the product backlog was selected for completion in the upcoming sprint.

3.6.4 Daily Sprint Meeting

Regular brief meetings were held with the thesis supervisor. Progress, challenges, and roadblocks was discussed, and daily activities planned. This promoted transparency, accountability, and collaboration.

3.6.5 Sprint Review Session

At the end of each sprint, progress and accomplishments were reviewed. Findings were presented, research insights shared, and any necessary adjustments or changes discussed. Feedback from the supervisor was used to improve and refine the work.

3.6.6 Potential Deliverable Product

After the first iteration, the Agile Product Owner collects the customer-ready product. This product may be insufficient, but it can still be used to satisfy some requirements.

3.7 Utilization and Dissemination of Research Results

The findings of this research will be disseminated through various channels. They will be made accessible in Strathmore University's digital repository, ensuring wide accessibility to interested parties. Furthermore, the findings will be presented to policymakers, contributing to a well-rounded understanding of the research's implications in the broader societal context. This research's significance extends beyond academia. Its insights hold substantial value for financial institutions, including banks and SACCOs. By offering the ability to anticipate mortgage default probabilities among their clients, these entities can make more informed decisions. Moreover, the research provides recommendations for appropriate mortgage amounts based on client profiles, enhancing the risk assessment process. In addition to its immediate applications, this study's contributions will resonate with future scholars delving into the mortgage default realm.

By expanding the knowledge base, this research opens avenues for further investigation and the refinement of strategies aimed at mitigating mortgage defaults.

3.8 Ethical Considerations and Issues

The researcher diligently navigated the procedural channels, seeking the indispensable ethical clearance from the institutional ethics committee. With unwavering commitment, the researcher is poised to safeguard the sanctity of privacy and the cloak of confidentiality enveloping the amassed data. Adherence to the most rigorous ethical standards was the cornerstone of this research, meticulously upheld throughout the research journey. The researcher sought clearance from the Strathmore University's institutions ethical review committee and National Commission for Science, Technology and Innovation (NACOSTI) to ensure that the researcher complied with the necessary requirements for research.

Chapter 4: System Analysis and Design

4.1 Introduction

Systems analysis entails the examination and exploration of genuine operational procedures. A comprehensive examination of various systems, including entities, organisms, organizations, beings, and objects, is necessary. In order to establish a conceptual understanding of a given situation, the process of systems analysis involves breaking down the complexities associated with various entities into their individual components and examining the interconnections between them. The primary aim of this study is to acquire a thorough understanding of the most effective strategies for addressing complex challenges commonly encountered by executives (Conger & Mason, 2013). The resultant conceptual definition is often operationalized in the form of a website, process-support system, or mathematical model. The objective of this chapter is to enhance understanding of the system by considering the requirements put forth by various possible users of the system. Additionally, this chapter seeks to determine the essential functioning of the system by examining its interactions with similar tools. The system requirements encompass both functional and non-functional needs. This chapter examines the aspects of system design and architecture in connection to its execution.

4.2 System Analysis

4.2.1 Requirement gathering

The researcher deemed it imperative to gather requirements in order to understand the users' demands and, of greater importance, to establish the goal of the system. The aforementioned needs were predominantly obtained using qualitative research methods, including observation, analysis, and document examination. To acquire primary experiential knowledge, the researcher engaged in direct observation of the practical implementation of pre-existing models for predicting mortgage defaults. Through the application of qualitative methodologies, such as observation and comprehensive analysis, the researcher successfully unveiled the complex network of user requirements. By conducting an analysis of established models inside real-life scenarios, the researcher explored their feasibility and constraints.

4.2.2 Functional Requirements

Functional requirements refer to the necessary procedures and responsibilities that a system must undertake in order to achieve its intended objective. The system should allow users to:

i) Create an account on the system.
ii) Login to the system after successful authentication.
iii) Predict mortgage default of a selected user.
iv) Recommend mortgage amount based on user's profile.
v) View reports
vi) Access their profiles and edit them.

4.2.3 Non-functional Requirements

In contrast to the functional requirements, the non-functional needs of the system are unrelated to the forecast of mortgage defaults. However, they influence the system's expectations with regards to its use and effectiveness. These factors are responsible for ensuring the seamless execution of system deliverables, namely the system requirements.

i) The system must be simple to maintain.
ii) The system must always be accessible to end users.
iii) The system should be simple to use and require little training. This is necessary so users can operate it with little or no training.
iv) Because it deals with critical life challenges, the system must provide consistent and reliable information.

4.3 System Architecture

The system architecture defines the framework for the different components comprising the mortgage default prediction system. The inputted consumer data constitutes the initial point of interaction with the tool. The data is processed through an API endpoint, during which the default value and the client-recommended quantity are predicted as depicted in Figure 4.1.

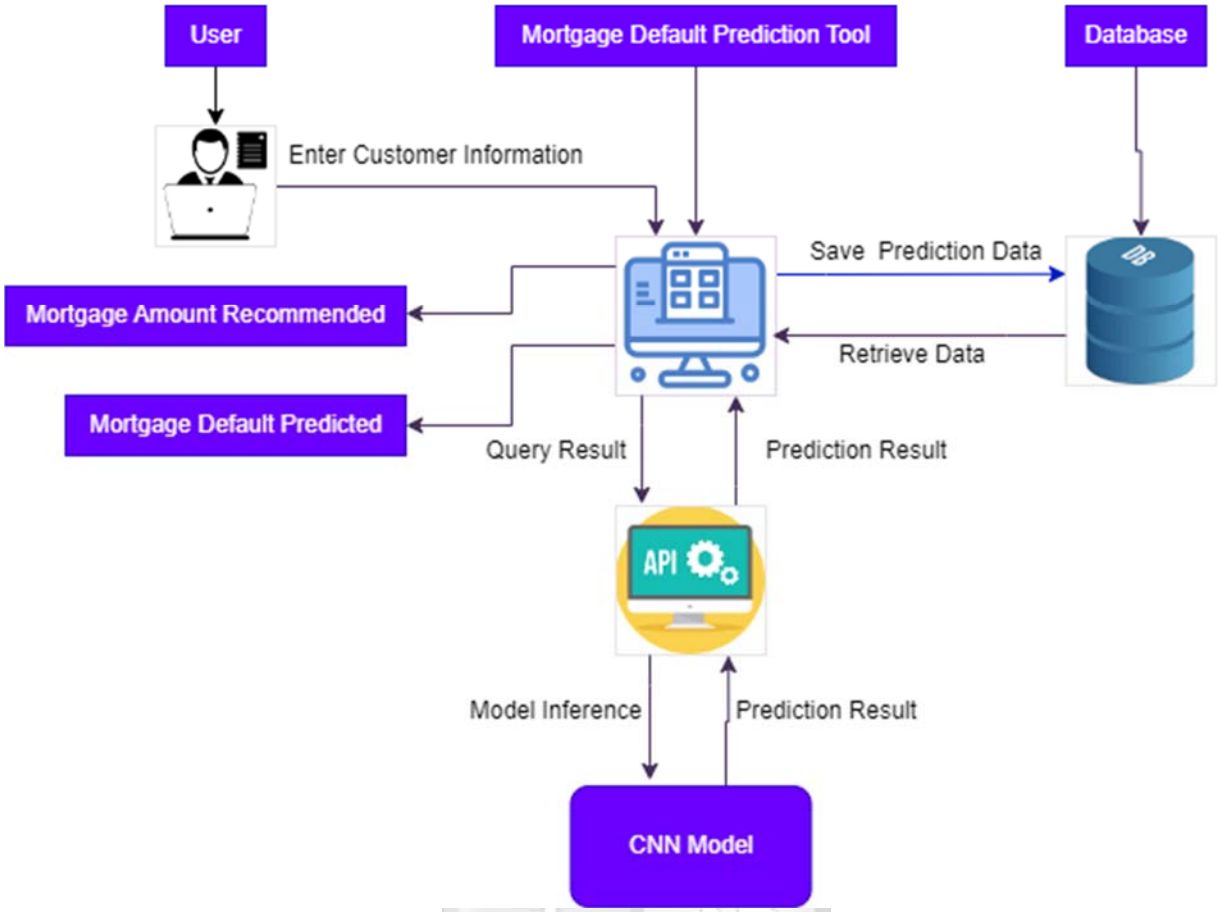


Figure 4.1 System Architecture.

4.4 System design

The system design encompasses the comprehensive plan for executing the implementation process, commencing from the first stages and concluding with the final stages. This section provides an analysis of the diagrams that serve as a foundation for the object-oriented implementation technique. The utilization of Object-Oriented Analysis and Design (OOAD) was employed throughout the process of system analysis, design, and development. Object-Oriented Analysis and Design (OOAD) stands as a pivotal software engineering methodology, encompassing the formulation of object-oriented models delineating the fundamental components of a software system. These models are then used as a framework to guide the development process (Rumbaugh, 2013). The model concepts and notation aim to represent design choices that exert a substantial influence on the ultimate system.

4.4.1 Use Case diagram

A use case diagram elucidates the interaction between a user and a system, elucidating their engagement with various use cases. Figure 4.5 presents a visual depiction delineating how actors interact with the mortgage default prediction tool.

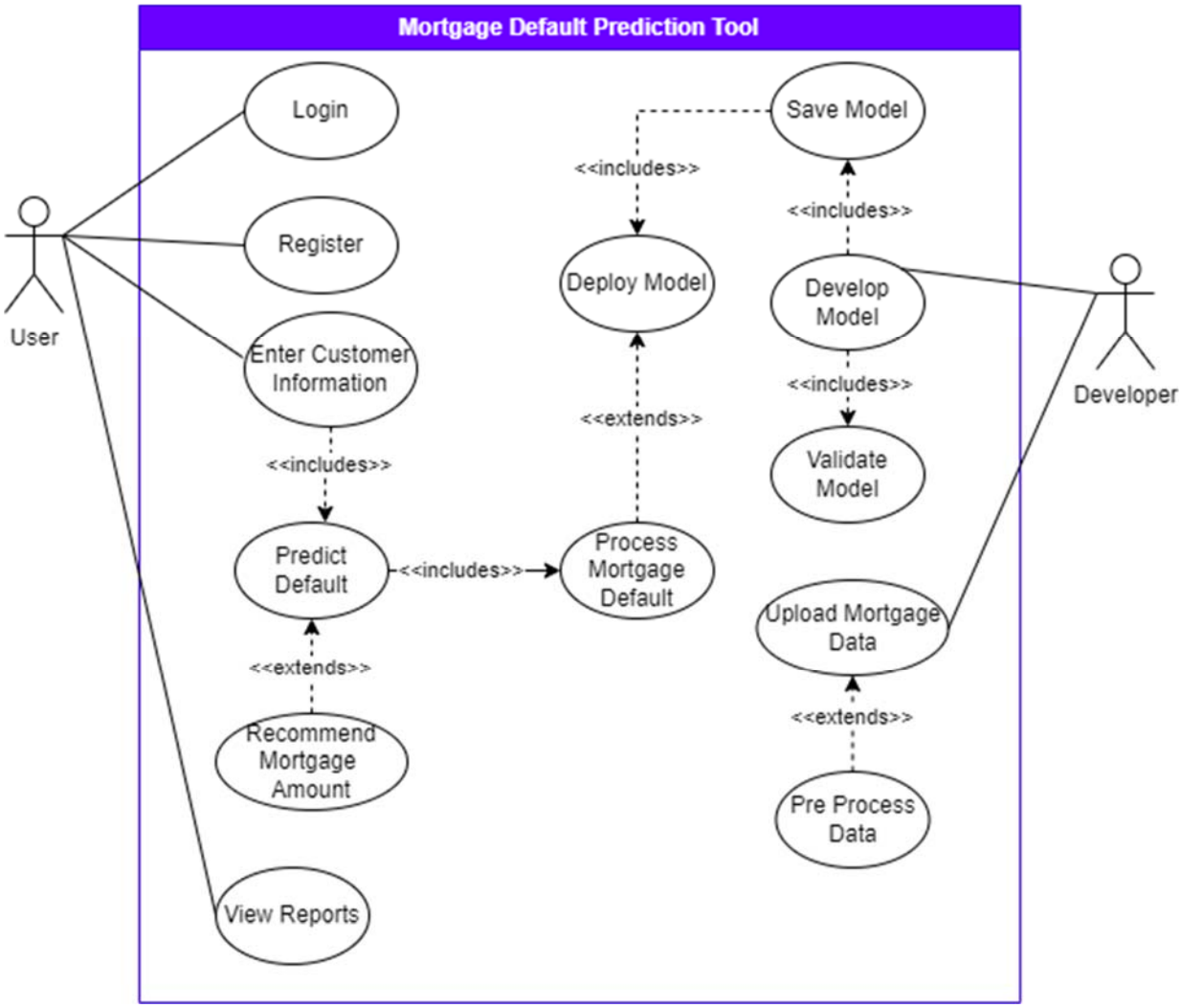


Figure 4.2 Use Case diagram.

4.4.2 Detailed use case descriptions

Table 4.1 gives a detailed description of the use cases in Figure 4.2.

Table 4.1 Use Case Descriptions

Use Case	Preconditions	Main Success Scenario	Post Conditions
User Registration	The user is connected to the internet	i). The user enters details in the registration form. ii). The system saves user details.	None
User Login	User is Registered	The user is logged in two the system	None
Predict Mortgage Default	The user is authenticated	i). The user enters customer information. ii). The system predicts mortgage default.	None
Recommend Mortgage Amount	The system has predicted mortgage default	i). User gets the recommended mortgage amount for a customer	Recommended amount displayed
Edit Profile	The user must be logged in	i). User edits their profile ii). The profile is edited successfully	.
Logout	user must be logged in	i). User clicks on the logout button	The user logs out of the system

4.4.3 Sequence diagram

Figure 4.3 depicts a sequence diagram that offers a thorough portrayal of the user's interaction with the system. It showcases the complicated interplay among the many components of the system and the precise orchestration of requests and responses. The process of verifying the identity of the user is an essential need that must be fulfilled prior to engaging in the analysis of mortgage default prediction.

The diagram represents the various components inherent to the system, each fulfilling a unique function in facilitating user interactions and eventually achieving the goals of the tool. The

journey commences as users navigate through the user interface object, which subsequently initiates communication with the API in order to provide prediction results. These results are then sent to the user, offering them useful insights.

The Convolutional Neural Network (CNN) model undertakes a multifaceted process behind the scenes. The procedure commences with data preprocessing and feature extraction, which are fundamental prerequisites for the succeeding convolutional pooling stage. Once this carefully selected model achieves its highest point, it is stored in a secure manner, ready to fulfil its crucial role in the field of mortgage default prediction and the subtle provision of mortgage amount recommendations.

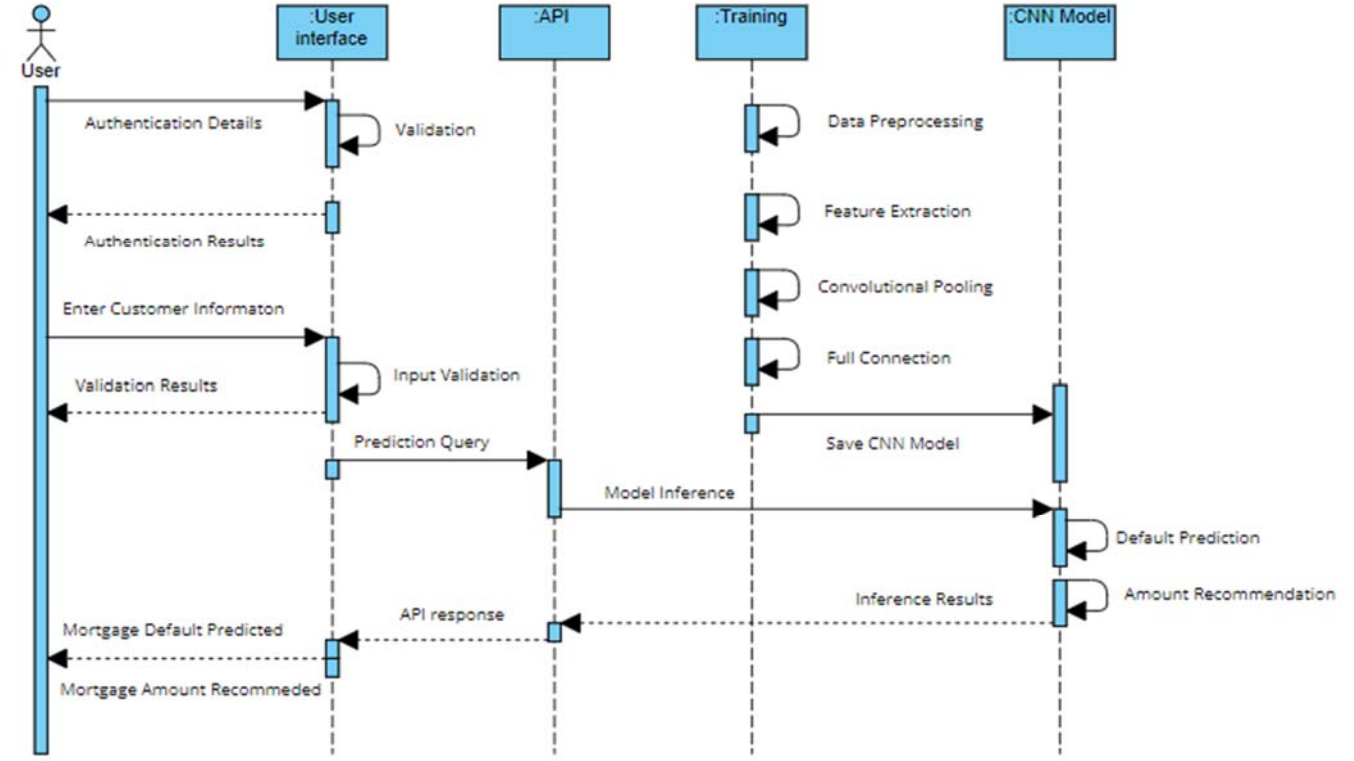


Figure 4.3 Sequence diagram.

4.4.4 Database Schema

The database schema serves as a representation of the conceptual data model of the produced system. The text illustrates the process of constructing the database. Additionally, it portrays the diverse entities inside the system. Figure 4.4 illustrates the database schema of the proposed system following the process of normalization.

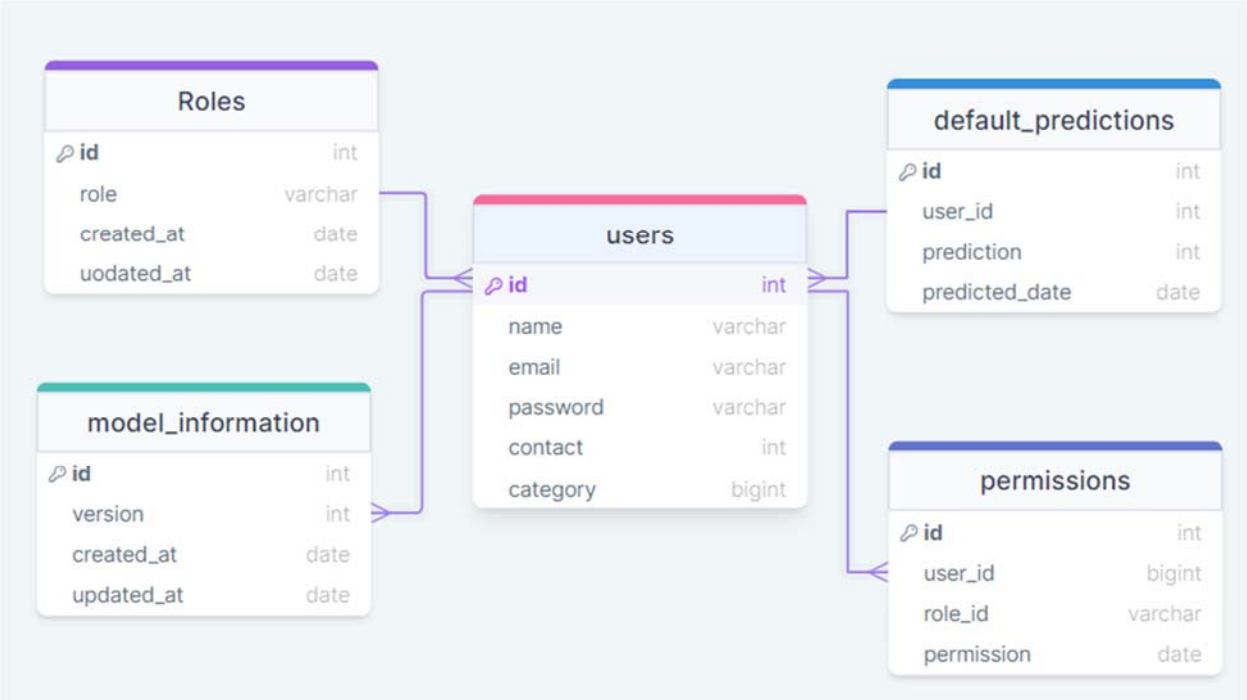


Figure 4.4 Database Schema

4.4.5 Class diagram

Class diagrams serve as a means to illustrate the architectural structure of a system by visually representing the associations between classes, as well as the characteristics of each class, such as attributes, operations, and the interactions that exist between objects. Figure 4.5 depicts a visual representation of the class diagram.

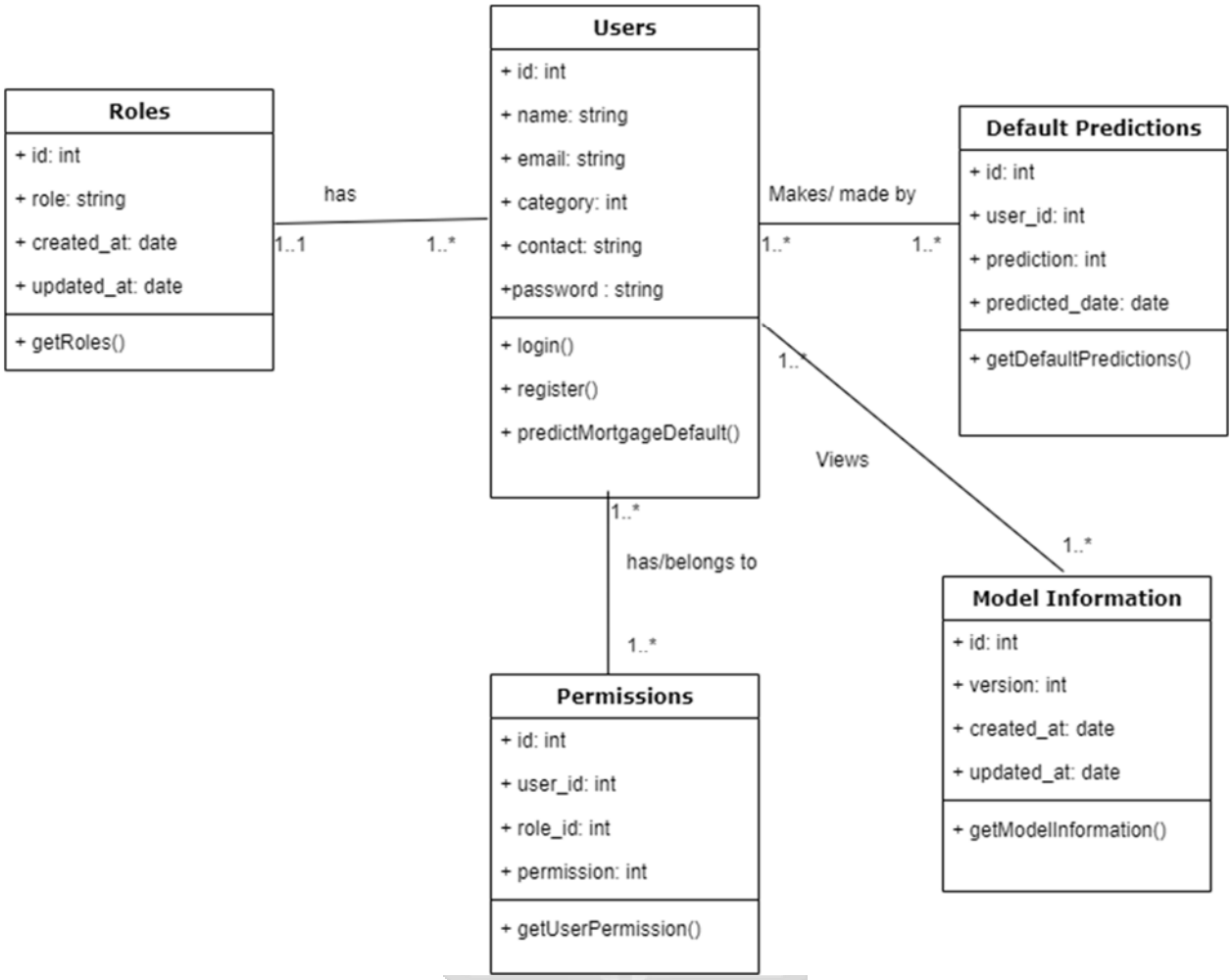


Figure 4.5 Class diagram.

4.5 Wireframes

A wireframe is a design used to inform the implementation of a system by rendering key intended features of the system based on user requirements or the developer's perspective.

4.5.1 Home

The landing page wireframe is shown below in Figure 4.6.

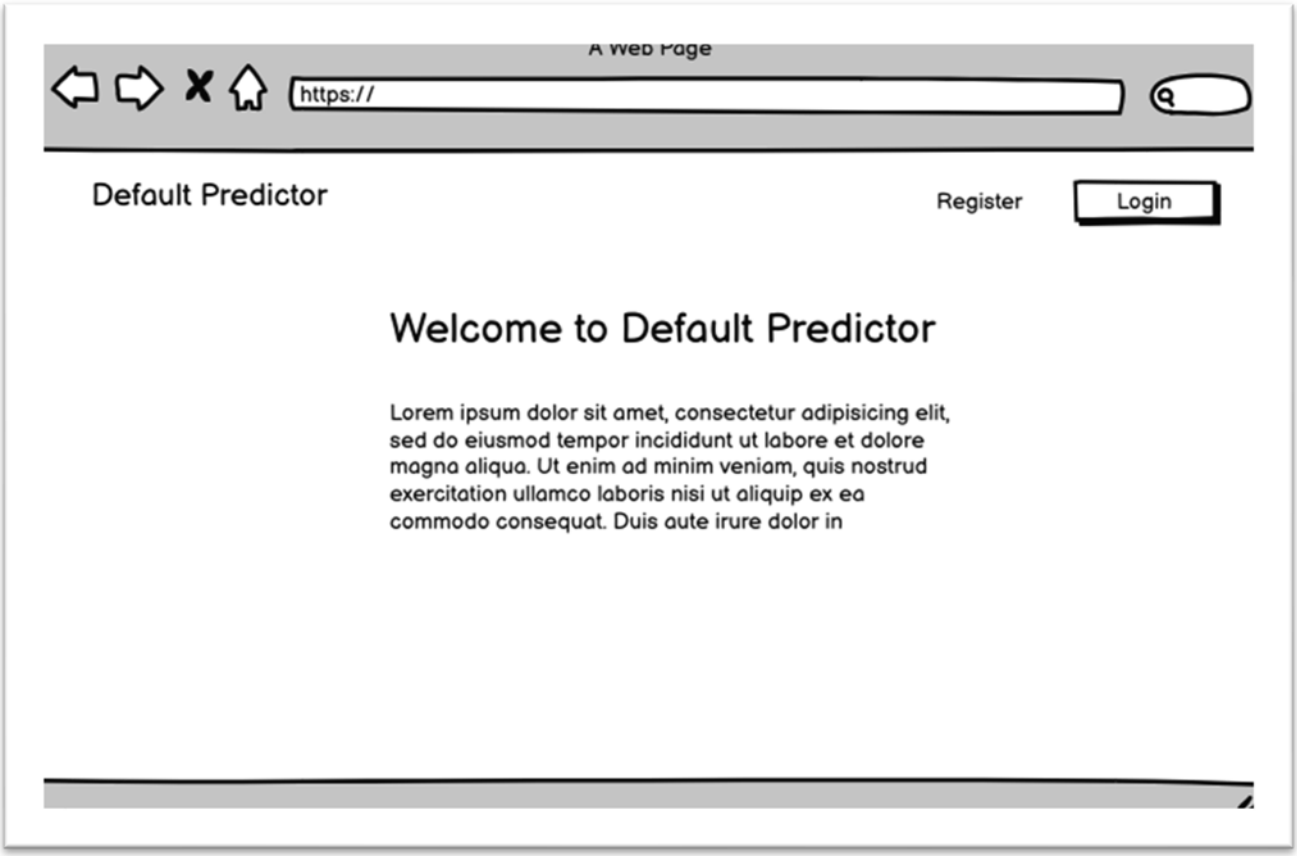


Figure 4.6 Home Wireframe

4.5.2 Register

Figure 4.7 shows the wireframe of the registration screen. Users are required to fill in all the fields to register.

Figure 4.7 Signup Wireframe

4.5.3 Login

Figure 4.8 Login Wireframe

Figure 4.8 above shows the login wireframe, which requires users to provide their login credentials authentication.

4.5.4 Dashboard

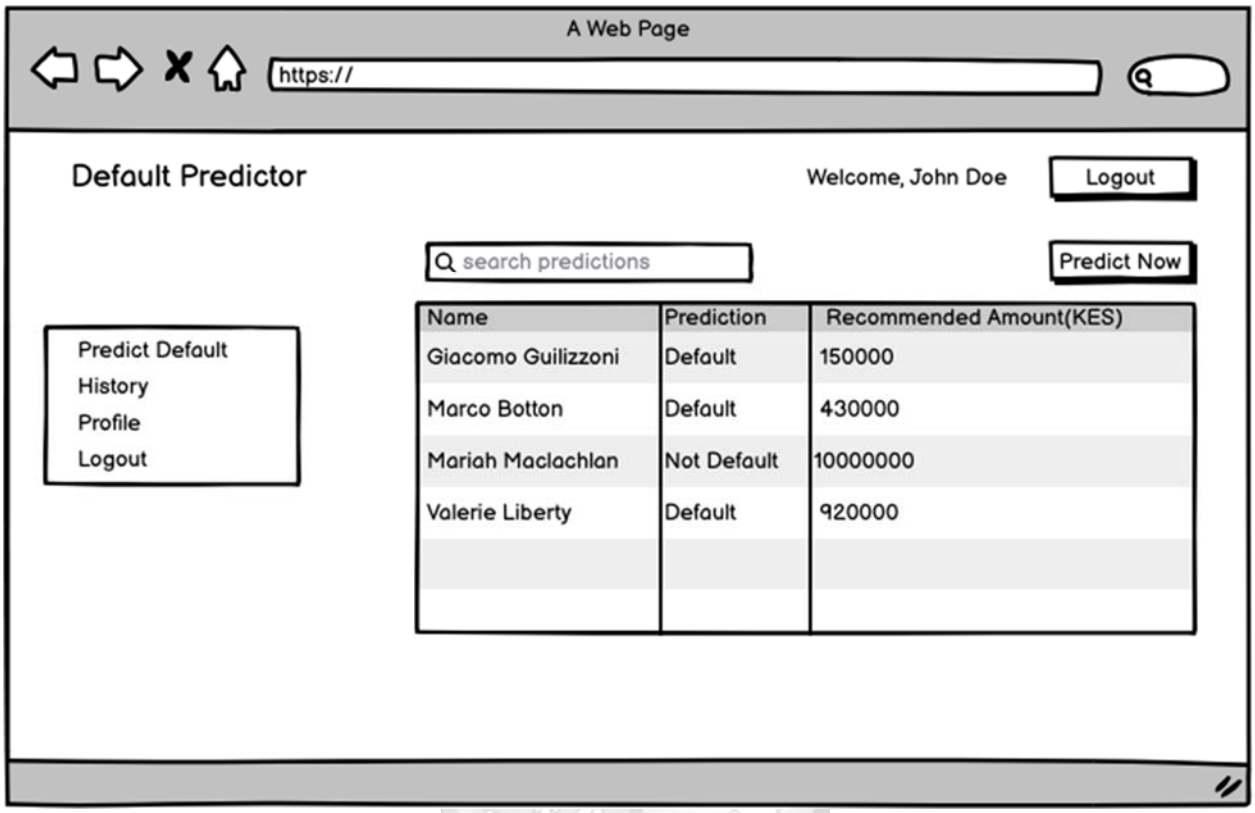


Figure 4.9 Dashboard Wireframe

4.5.6 Mortgage Default Prediction Wireframe

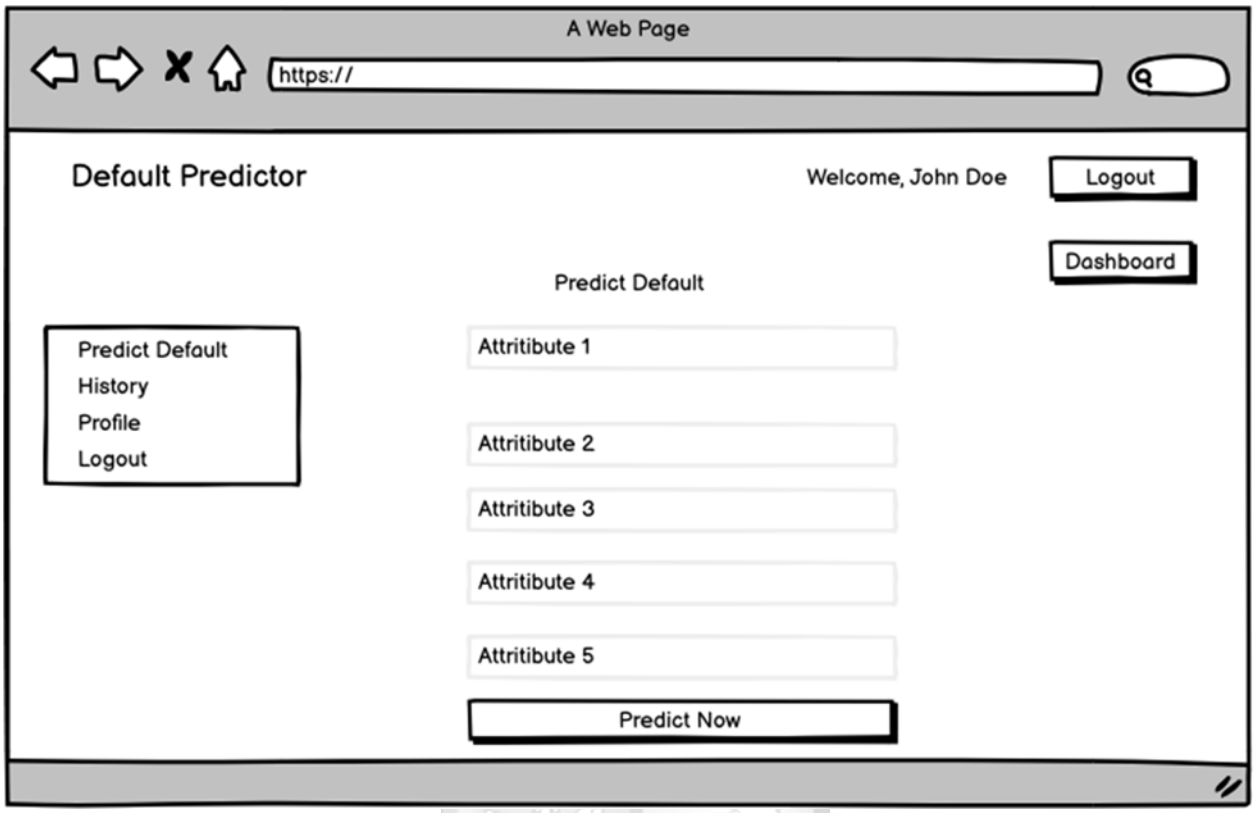


Figure 4.10 Mortgage Default Prediction Wireframe

Chapter 5: System Implementation and Testing

5.1 Introduction

The system architecture comprised a hierarchical structure, consisting of a high-level module interfacing with a subsystem housing the prediction algorithm. The high-level module facilitated user management functionalities, data uploading, preliminary data validation, interaction with the data storage engine, integration with the default prediction model, and provision of the graphical user interface. This system operated as a web application developed using the Laravel framework. The mortgage default prediction model was seamlessly integrated into the system as a dynamically linked library, implemented using Python. Data storage was facilitated through a MySQL database. The web application empowered users to log in, input customer datasets, and request default predictions and recommended amounts. This process involved invoking a call to the Flask API, which in turn loaded the default prediction model.

5.2 Software and Hardware Requirements

Table 5.1 Hardware Requirements

Hardware	**Specifications**	
HP Laptop	Processor	Core i7
	RAM	8GB

Table 5.2 Software Requirements

Software	**Specifications**	
Google Collaboratory	GPU	12GB (11.439GB Usable), GDDR5 VRAM; 2496 CUDA cores; Compute Version 3.7; 1xTesla K80
	CPU	1xhyper threaded single core
VS Code	Version 1.87	
PHP	Version 8.1.0	

5.3 Model Development

The model was developed employing advanced deep learning methodologies within the realm of machine learning. Subsequent sections delineate the systematic procedure undertaken to execute the model.

5.3.1 Data Preprocessing

The initial requirement comprised obtaining a dataset in CSV format. Prior to its utilization in model development, the data necessitated pre-processing, encompassing the following sequential steps:

i. Loading the Dataset

To load data, the process begins by importing the necessary module from Google Colab, utilizing the command "drive.mount('/content/drive')" to mount the Google Drive. This action establishes a connection between the code environment and the Google Drive storage. Subsequently, a path is defined ("/content/drive/MyDrive/MortgageDefaultPrediction") where the hmeq dataset is stored. Finally, the data is loaded into the code environment using the "pd.read_csv()" function from the pandas library.

```
from google.colab import drive
drive.mount('/content/drive')


path = "/content/drive/MyDrive/MortgageDefaultPrediction"

hm=pd.read_csv(path +"/hmeq.csv")
```

Figure 5.1 Loading Dataset

ii. Feature Selection

Many techniques, such as feature selection and dimensionality reduction, are employed to enhance machine learning models' efficacy. In this research, we'll build five different training sets, each with its own distinct set of characteristics. T1 is the complete set, and the subsequent sets are all subsets of T1. T2 relies on attributes with a strong correlation to default.

Characteristics with significant positive correlations to default are located in T5, while features with high correlations to other features in the core set are located in T3.

iii. Feature scaling

In most mortgage datasets, features have multiple scales. A big scale gap between features frequently slows down optimization procedures. Feature scaling may aid in improving classification performance and learning efficiency in various ML algorithms. Following the data conversion stage, standardization and normalization will be conducted on the original data.

iv. Data Cleaning

The process involved eliminating statistically insignificant variables, such as the ID variable, as well as instances with missing or undefined values for categorical variables. The following method was employed for data cleaning:

```
#creating a list of bool features
bool_cols = df.select_dtypes(['bool']).columns.tolist()

#droping missing value flags
X = df.drop(columns = bool_cols)
```

Figure 5.2: Data Cleaning

v. Data Splitting

The process of partitioning a dataset into multiple subsets for model training, validation, and testing purposes is commonly referred to as data splitting. Typically, this involves dividing the dataset into three primary subsets: the training set, validation set, and test set. The training set, being the largest subset, is utilized to train the model. Meanwhile, the validation set serves to assess the model's performance during the training phase and adjust its hyperparameters accordingly. Finally, the test set is employed to evaluate the model's overall performance post-training and validation. The primary objective of data splitting is to ensure that the model is trained on a diverse range of data and subsequently evaluated using unseen data, thereby

mitigating the risk of overfitting. Overfitting occurs when a model performs well on training data but poorly on new data. The process of data splitting is illustrated in Figure 5.3.

```
# Splitting the data into training and test set

x_train, x_test, y_train, y_test = train_test_split(X_scaled, Y, test_size = 0.2, random_state = 1, stratify = Y)
```

Figure 5.3: Data Splitting

vi. Conversion of individual variables to required model inputs.

This involved transforming variables like "BAD" into categorical formats, generating new variables such as the "TARGET" variable, and converting categorical variables into the 'factor' data type. The "TARGET" variable served as the focal point for prediction, enabling the assessment of the probability of customer default on their mortgage for any given month based on factors such as their occupation, debt-to-income ratio, and other pertinent attributes.

5.3.2 Exploratory Data Analysis (EDA) and Visualization

Exploratory Data Analysis (EDA) is an analysis approach that identifies general patterns in the data. These patterns include outliers and features of the data that might be unexpected. EDA is an important first step in any data analysis. The data underwent exploratory analysis to identify the patterns in the HMEQ dataset.

5.3.2.1 Univariate Analysis

Univariate analysis explores each variable in a data set, separately. It looks at the range of values, as well as the central tendency of the values. It describes the pattern of response to the variable. It describes each variable on its own. Descriptive statistics describe and summarize data.

i). Histogram boxplot for Loan

The variable LOAN exhibits a right-skewed distribution, characterized by a multitude of large outliers in comparison to the mean.

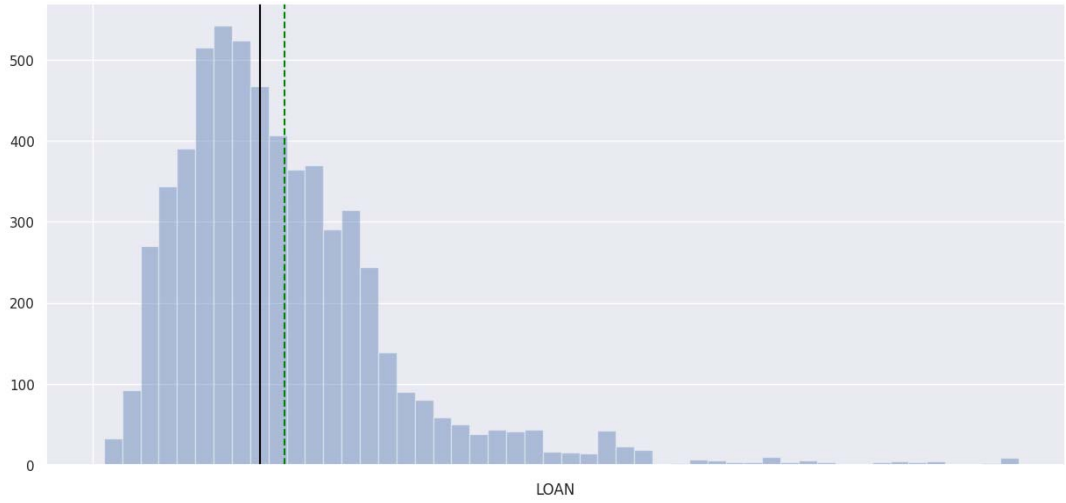


Figure 5.4 Histogram Plot for Loan

ii). Analyzing Bar Plot for Job

The variable JOB appears to exhibit a normal distribution.

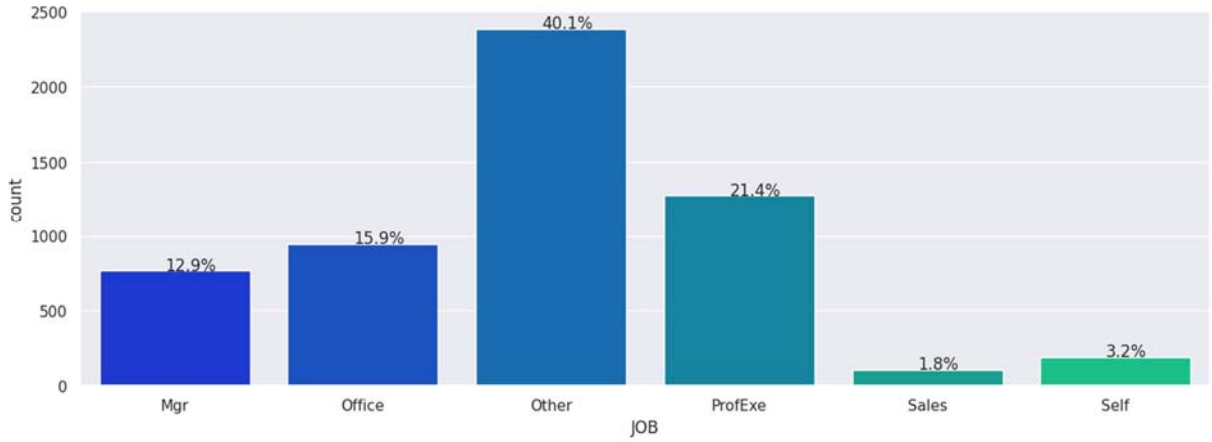


Figure 5.5 Bar Plot for Job

5.3.2.2 Bivariate Analysis

i). Analyzing Amount of Loan V Bad (Defaulted or Not)

Regardless of the client's type, whether defaulted or not, it appears that they receive an equivalent amount of loans.

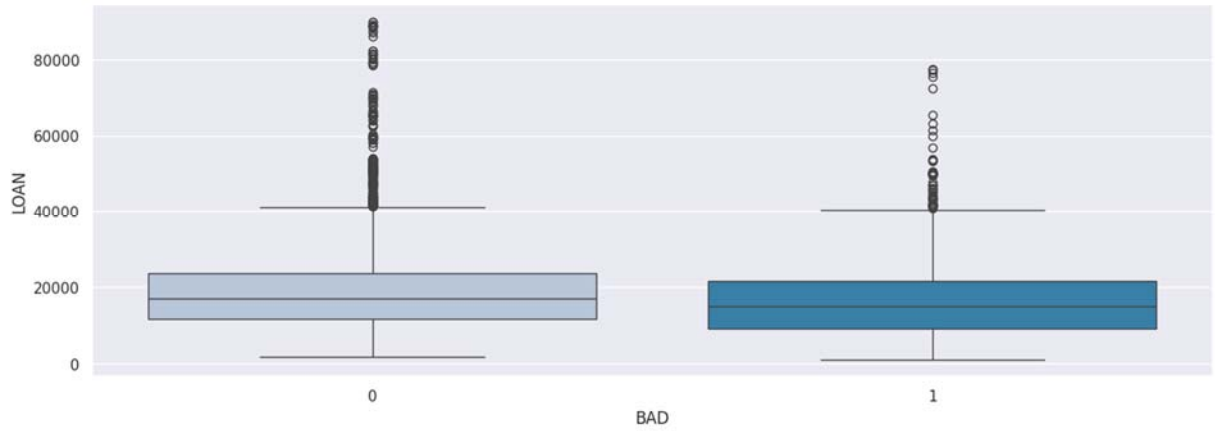


Figure 5.6 Amount of Loan V Bad

ii). Analyzing MORTDUE V Bad (Defaulted or Not)

Regardless of the client's status, whether defaulted or not, it appears that they owe the same amount on their existing mortgage.

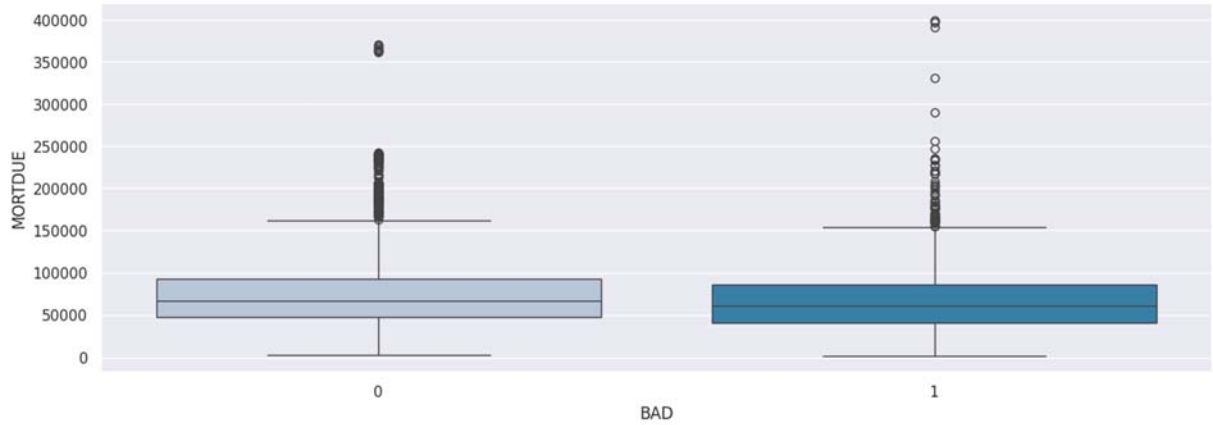


Figure 5.7 MORTDUE V BAD

iii). Analyzing DEBTINC V Bad (Defaulted or Not)

Regardless of the client's type, whether defaulted or not, there appears to be no discrepancy in the current value of the property.

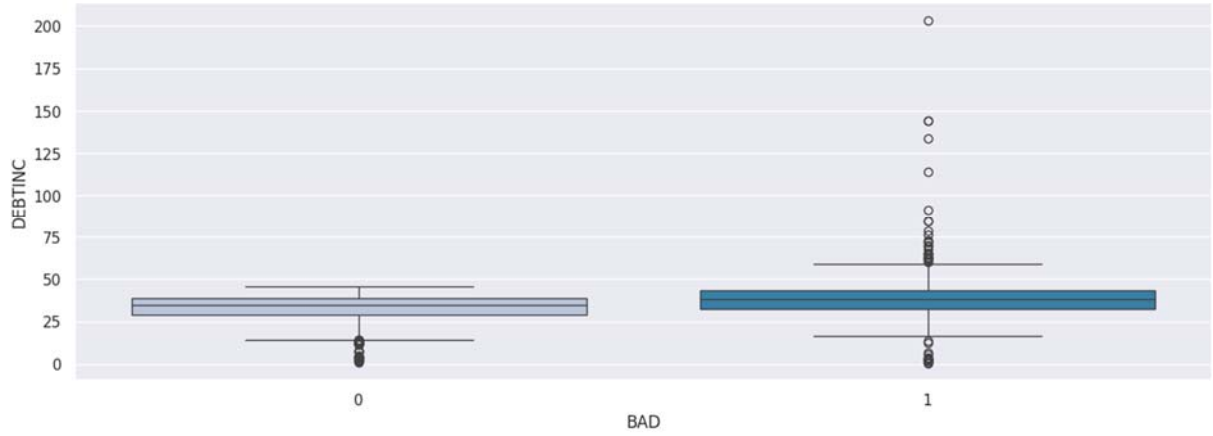


Figure 5.8 DEBTINC V BAD

iv). Analyzing Value V Bad (Defaulted or Not)

Both types of clients exhibit remarkably similar ratios.

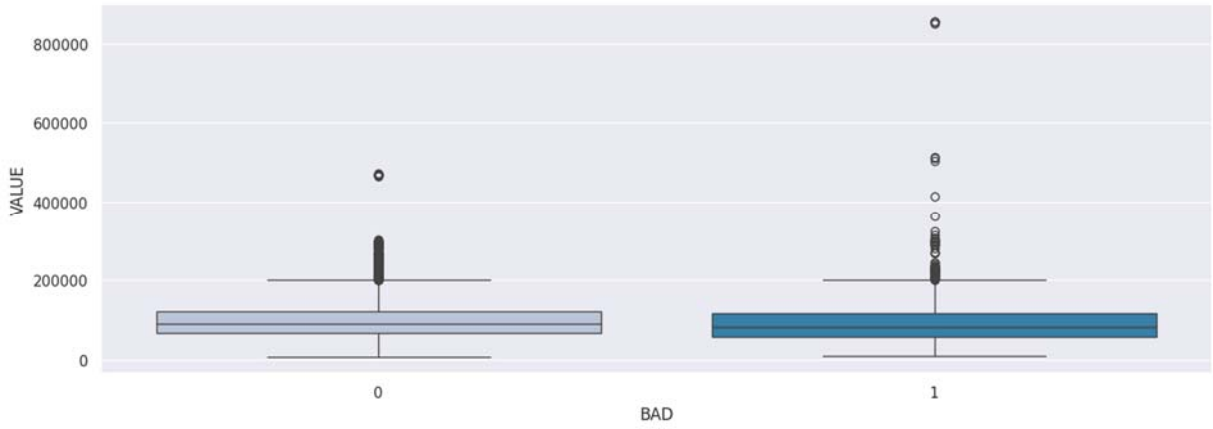


Figure 5.9 Histogram Plot for Loan

v). Analyzing Reason V Bad

In Figure 5.10, it is evident that irrespective of the loan's purpose, the majority, accounting for 80%, represents non-defaulted clients, while the remaining 20% corresponds to defaulted clients.

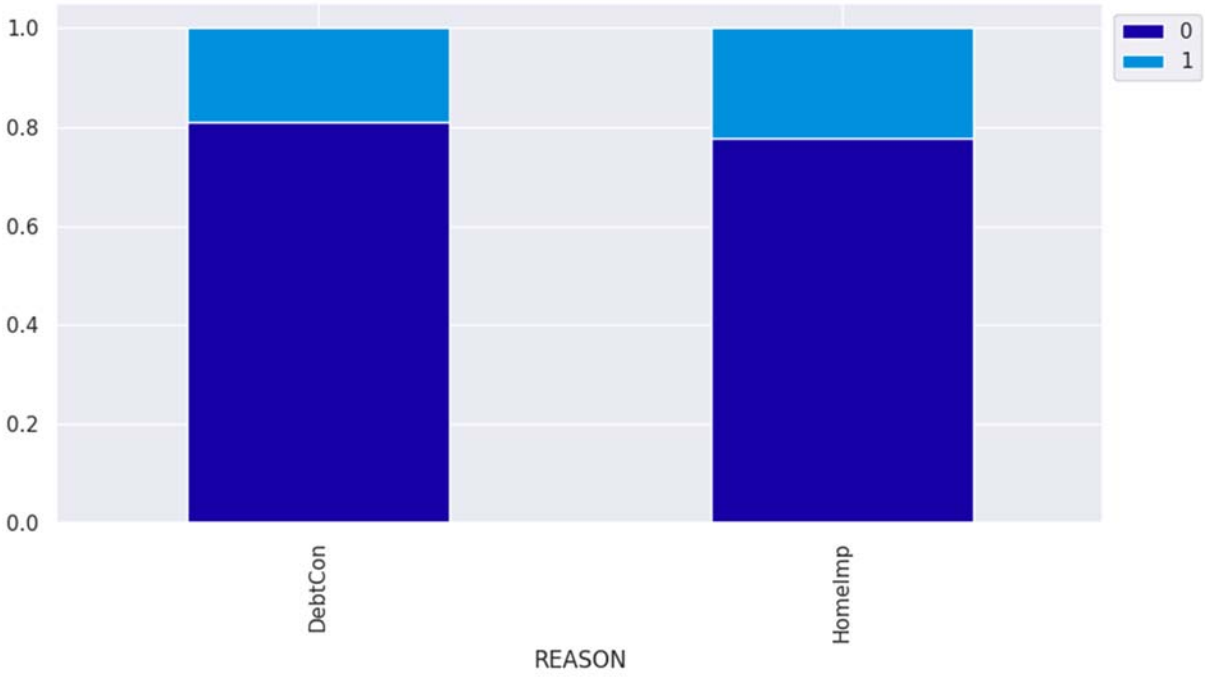


Figure 5.10 Reason V Bad

vi). Analyzing Job and Defaulted

In Figure 5.11, it is evident that individuals in the sales profession exhibited the highest percentage of defaulted loans.

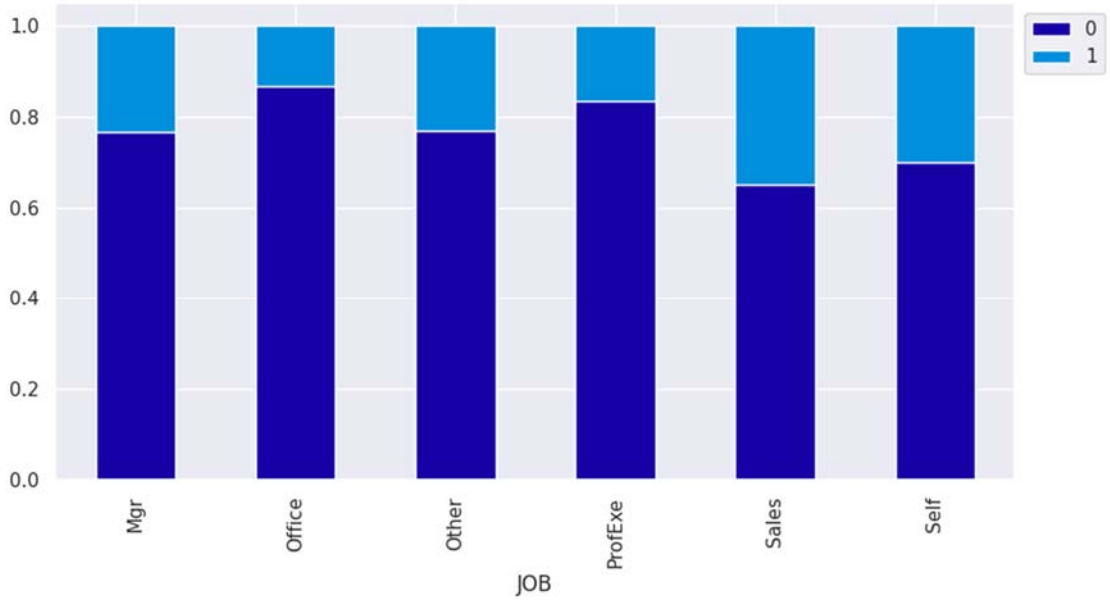


Figure 5.11 Histogram Plot for Job v Defaulted.

5.3.2.3 Correlational Analysis

The correlation heatmap shows the relationship between features. Darker colours signify stronger correlations, while lighter colours suggest weaker correlations. Positive correlations, where an increase in one variable corresponds with an increase in the other, typically manifest as warm colours like red or orange. Conversely, negative correlations, where an increase in one variable corresponds with a decrease in the other, are often depicted by cool colours such as blue or green. Figure 5.12 shows a correlation heatmap for the features used in the model construction.

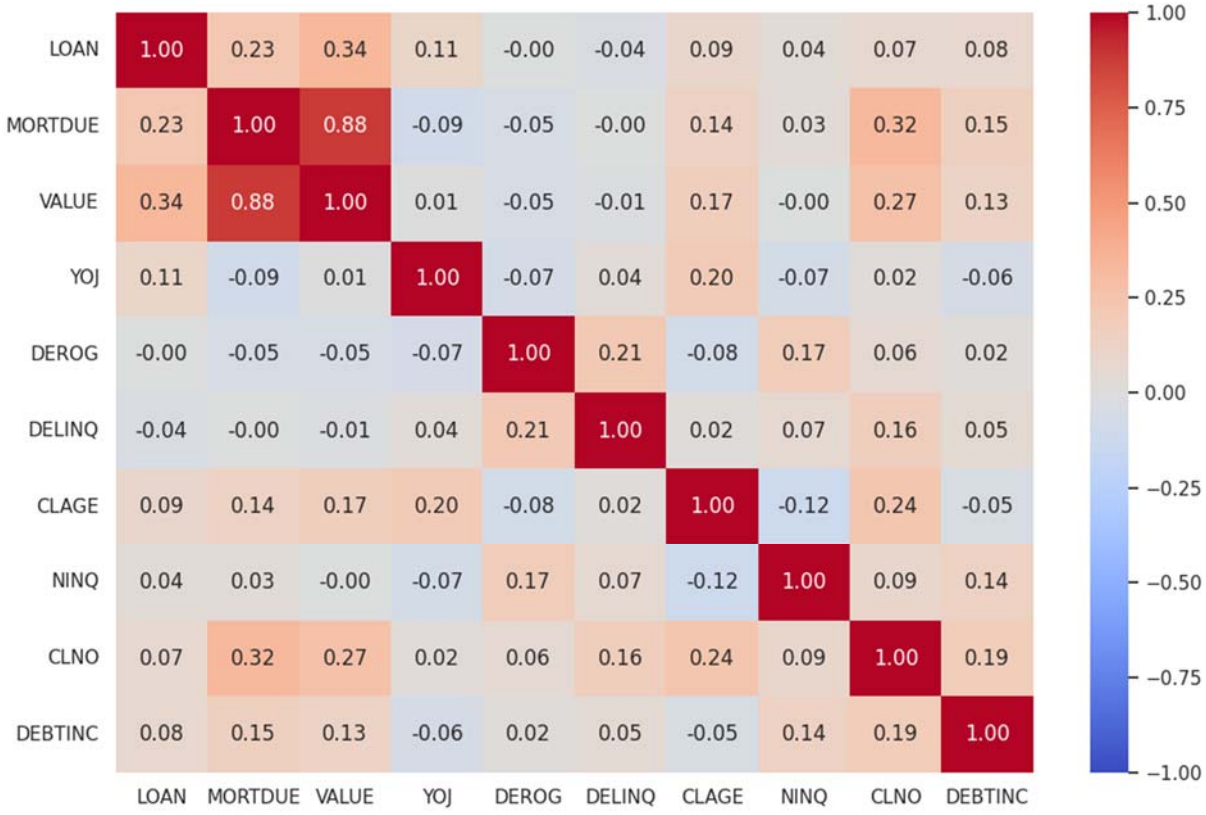


Figure 5.12 Histogram Plot for Loan

5.3.2 CNN Model

The code excerpt in Figure 5.13 delineates the construction and training process of a Convolutional Neural Network (CNN) model employing the Keras library alongside a TensorFlow backend. Initially, the CNN model architecture is established using the `Sequential` model, facilitating a sequential arrangement of neural network layers. The model commences with a convolutional layer, instantiated by `Conv1D`, featuring 32 filters with a kernel size of 3

and employing the Rectified Linear Unit (ReLU) activation function to extract pertinent features from the input data. Subsequently, a max-pooling layer is introduced utilizing `MaxPooling1D`, effectively reducing the dimensionality of the extracted features by selecting the maximum value from a pool of size 2. Another convolutional layer ensues, integrating 64 filters and a kernel size of 3, once again utilizing the ReLU activation function, followed by another max-pooling layer.

Following the convolutional and pooling layers, a global average pooling layer is added to compute the average value of each feature map across all spatial dimensions, thereby generating a fixed-length vector representation. Additional layers are appended to the model for the incorporation of fully connected neural networks. These layers comprise a dense layer housing 128 neurons, activated by ReLU, succeeded by a single-neuron dense layer employing sigmoid activation, deemed suitable for binary classification tasks. Upon defining the model architecture, it is compiled utilizing the Adam optimizer and binary cross-entropy loss function, standard choices for binary classification problems. Furthermore, the accuracy metric is specified to gauge model performance throughout the training process.

The model undergoes training using the provided training data (`x_train` and `y_train`) over 10 epochs, employing a batch size of 32. To assess its performance during training, a validation split of 20% is employed, evaluating the model's predictive capabilities on a subset of the training data. Leveraging the CNN classifier, the model ingests customer characteristics to forecast the probability of mortgage default.

```
from keras.models import Sequential
from keras.layers import Dense, Conv1D, MaxPooling1D, GlobalAveragePooling1D
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report, precision_recall_curve, recall_score, f1_score

# Build a CNN model
model = Sequential()
model.add(Conv1D(32, 3, activation='relu', input_shape=(x_train.shape[1], 1)))
model.add(MaxPooling1D(2))
model.add(Conv1D(64, 3, activation='relu'))
model.add(MaxPooling1D(2))
model.add(GlobalAveragePooling1D())
model.add(Dense(1, activation='sigmoid'))

# Add fully connected layers
model.add(Dense(128, activation='relu'))
model.add(Dense(1, activation='sigmoid'))  # Binary classification, so using sigmoid activation

# Compile the model
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
model.fit(x_train, y_train, epochs=10, batch_size=32, validation_split=0.2)
```

Figure 5.13: CNN Classifier

5.3.2 Model Results

Three models were tested, and their performances were compared to each other. The CNN model achieved an impressive performance with 97.14%, followed by Gradient Boosting at 88.49%, and KNN boosting a top performance at 99.25%, as shown in the snapshots below.

```
Test Loss: 0.08947115391492844, Test Accuracy: 0.9713587164878845
```

Figure 5.14 CNN Accuracy

```
Accuracy: 0.9924903946908837
```

Figure 5.15 KNN Accuracy

```
Accuracy: 0.8849109325881942
              precision    recall  f1-score   support

           0       0.88      0.90      0.89      2923
           1       0.89      0.87      0.88      2803

    accuracy                           0.88      5726
   macro avg       0.89      0.88      0.88      5726
weighted avg       0.89      0.88      0.88      5726
```

Figure 5.14 Gradient Boosting Accuracy

5.4 System Implementation

The system was engineered to offer a comprehensive suite of user functionalities, seamlessly integrating with the mortgage default prediction model to facilitate predictive analysis. Facilitating user interaction, a web application served as the interface through which banking institutions could input customer data and receive predictions regarding default probabilities, along with recommended loan amounts. Leveraging a Convolutional Neural Network (CNN) model, the system undertook model training and prediction tasks. To manage user data

effectively, a MySQL database was employed to store user information, including user groups and permissions. The development environment for the system encompassed the following technologies:

i. Laravel – PHP framework for development of the web application

ii. MySQL database

iii. Machine Learning (Model Development)

iv. Flask (API development)

The software workflow for the system commences with a user authentication procedure, wherein mortgage officers input their login credentials to gain access to the dashboard. Upon successful authentication, mortgage officers are granted access to their profiles, empowering them to execute system configuration tasks such as user creation and removal. Within the mortgage officer's profile page, they can input customer features for predictive analyses. As depicted in Figure 5.17, the authentication process is illustrated through the login form.

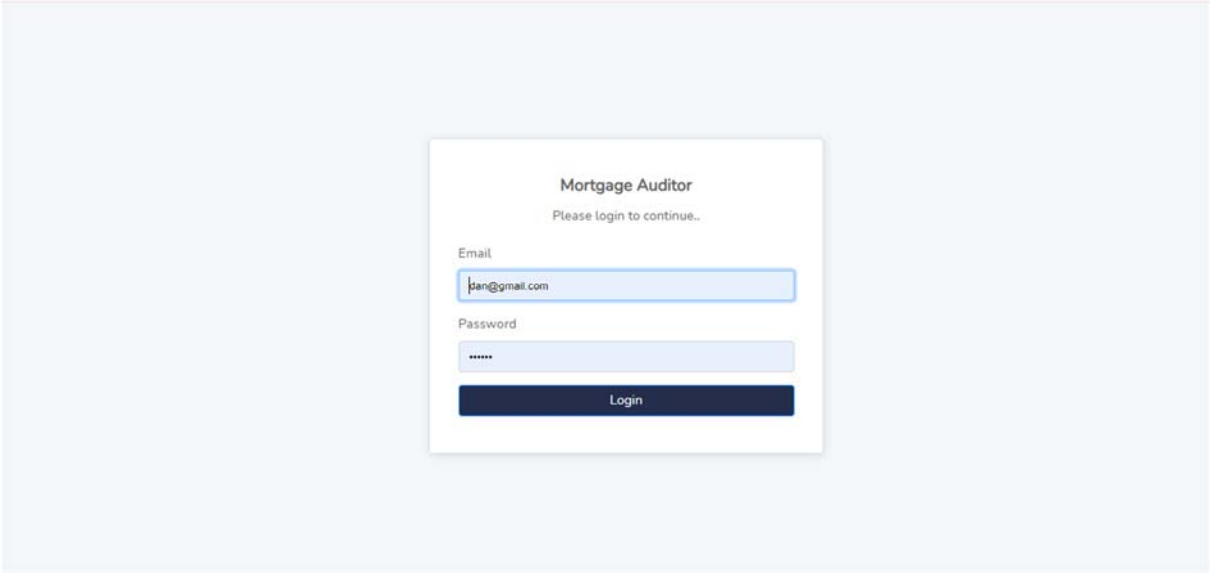


Figure 5.15: Login Form

Figure 5.6 shows the dashboard after successful authentication.

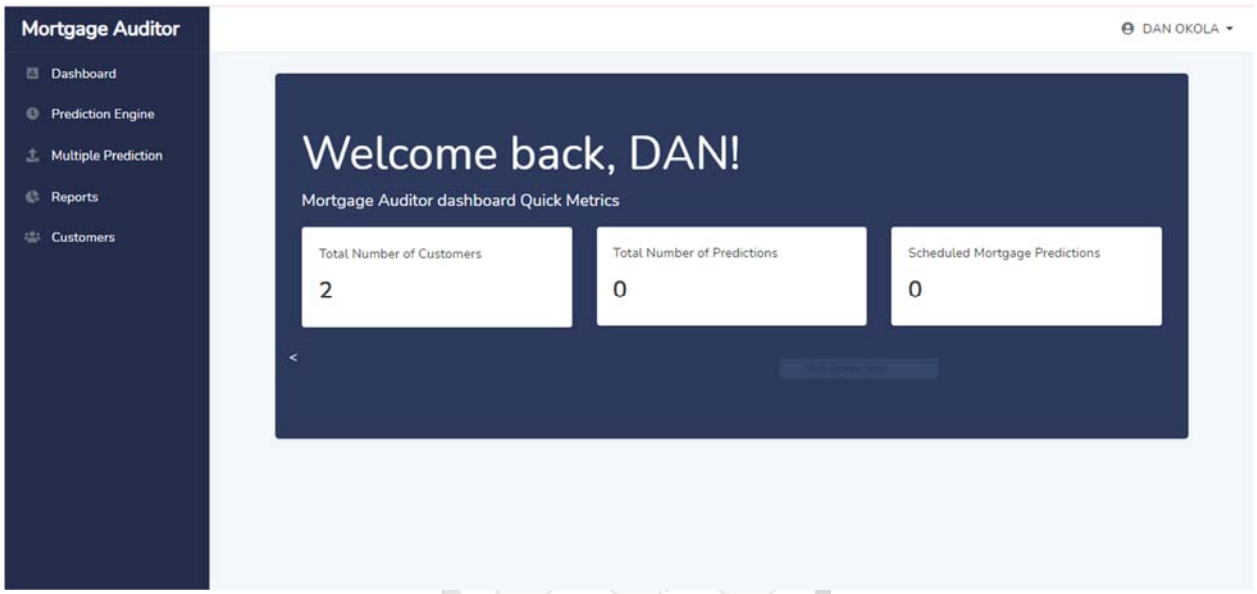


Figure 5.16: Dashboard

Figure 5.17 shows the interface from which users can input customer features and request predictions for mortgage default and get amount recommendation.

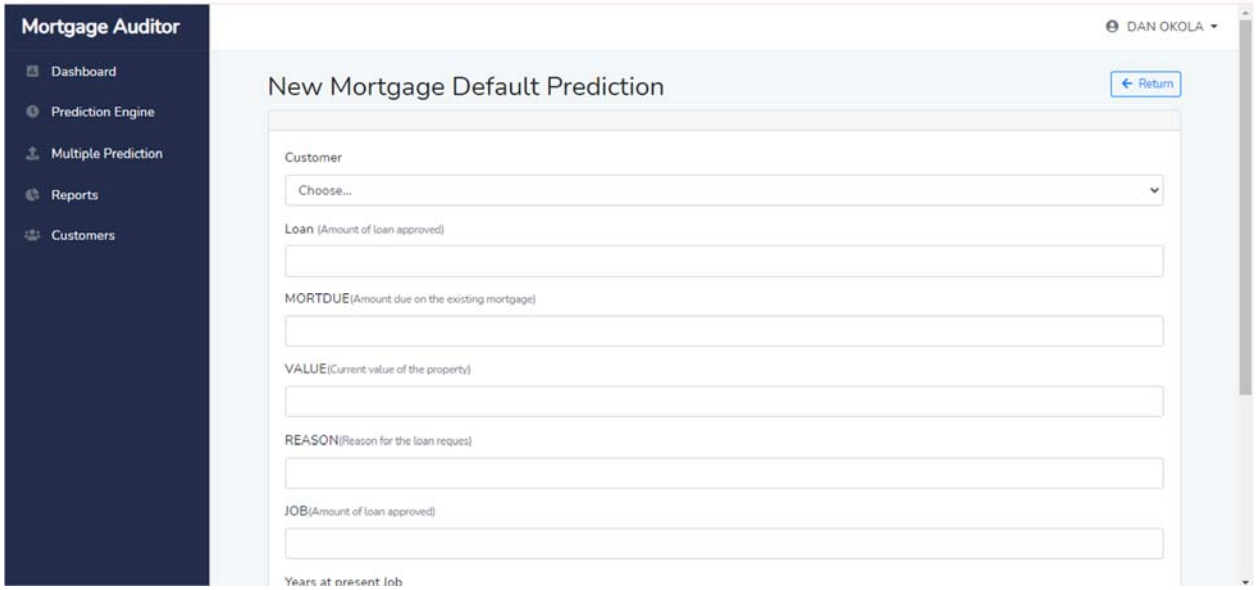


Figure 5.17: Prediction Form

Figure 5.18 shows default prediction history.

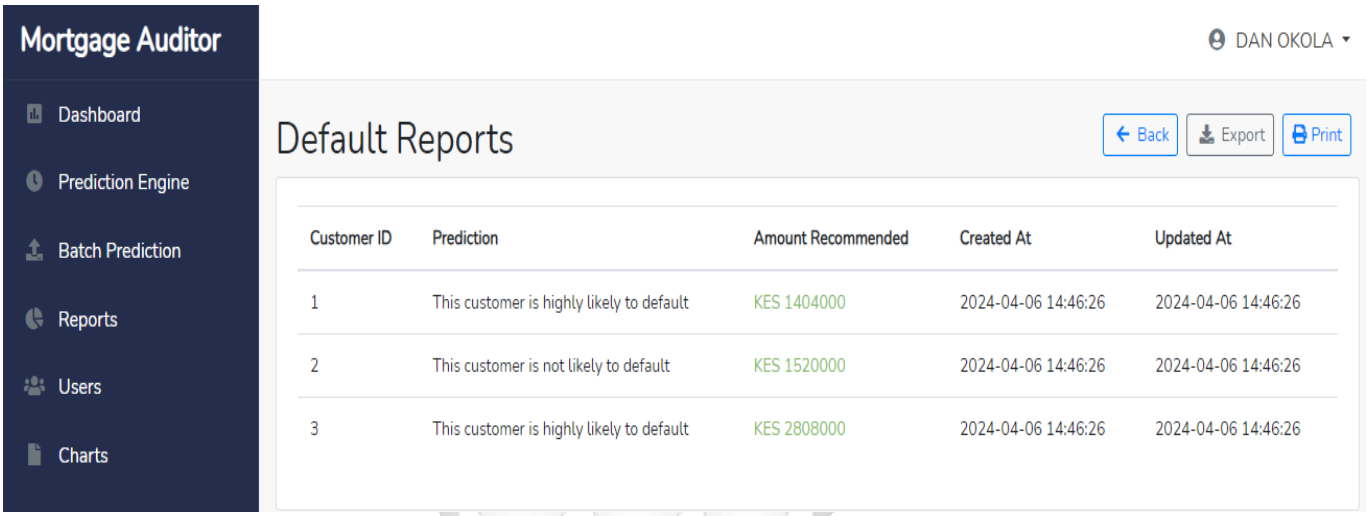


Figure 5.18: Prediction History

5.5 Testing and Validation

Testing and validation endeavours were undertaken with the primary objective of ascertaining the model's capability to forecast the likelihood of mortgage default among customers and propose appropriate amounts based on their individual attributes, thereby furnishing dependable outcomes applicable to risk management practices. The reliability of these outcomes was assessed through parameters such as model accuracy, precision, and recall. The mean accuracy of the mortgage default prediction model's forecasts was determined to be 97.14%. Table 5.1 below shows the test results for the mortgage default prediction tool.

Table 5.1 Test Results

Test	Results	Response Rate
Data upload to the tool in CSV form.	Data uploaded to the tool successfully	Fast
Mortgage Default Predicted by the tool.	Mortgage default predicted by the tool.	Fast
Mortgage Amount Recommendation.	The tool recommends mortgage amount.	Fast

Chapter 6: Discussion

6.1 Introduction

The study culminated in the creation of a predictive model tailored for forecasting mortgage default rates and suggesting optimal mortgage amounts. Leveraging Convolutional Neural Networks (CNN), the model was meticulously crafted. The dataset employed in the study encompassed both personal attributes of customers and transactional data. By using the model, it was possible to predict the probability of customers defaulting on their mortgage and also recommend mortgage amounts with an accuracy of 97.12%.

6.2 Discussion

The development of the mortgage default prediction tool marks a significant advancement in leveraging cutting-edge technology to address critical financial challenges. Utilizing CNN, a sophisticated deep learning algorithm, automated feature engineering was employed to create a robust predictive model. The data underwent pre-processing and cleaning, essential steps to ensure the model's accuracy and reliability. Initially, the model exhibited a commendable accuracy of 81.01%. However, through hyperparameter tuning, its performance soared to an impressive 97.14%, underscoring the effectiveness of fine-tuning in enhancing predictive capabilities. The culmination of these efforts resulted in a highly accurate and reliable mortgage default prediction model. Integral to the tool's functionality is its accessibility via an API, facilitated through the Flask framework. This accessibility ensures seamless integration into existing systems and workflows, enhancing its usability and applicability across diverse platforms. Moreover, the development of a consumer-facing tool using Laravel further underscores the commitment to user-centric design and accessibility. One notable aspect of this tool is its multifaceted functionality. Not only does it predict mortgage defaults with exceptional accuracy, but it also utilizes CNN to recommend optimal mortgage amounts, thereby mitigating default risks. This dual functionality sets it apart from conventional models, which typically focus solely on predicting default without considering proactive risk management strategies. Comparative analysis against existing literature highlights the superior performance of this model. Its ability to recommend mortgage amounts in addition to predicting defaults addresses a crucial gap in current research. By incorporating proactive risk management features, it empowers users to make informed decisions, thereby enhancing financial stability and resilience.

The consumer-facing interface further enhances the tool's utility, offering users a user-friendly platform to upload datasets or input customer characteristics for prediction. This intuitive interface streamlines the prediction process, ensuring efficiency and accuracy in decision-making. The development of the mortgage default prediction tool represents a significant milestone in leveraging advanced technology to address financial challenges. Its robust predictive capabilities, proactive risk management features, and user-friendly interface position it as a valuable asset in the financial sector, heralding a new era of data-driven decision-making.

6.3 Existing Algorithms and Models Used to Predict Mortgage Default

Drawing from the insights gleaned in chapter two, a multitude of machine learning algorithms have emerged as promising tools for predicting mortgage default. The realm of mortgage default prediction has witnessed extensive exploration of diverse algorithms. Among these, the KNN algorithm stands out; it categorizes new observations based on the nearest neighbors in the feature space, offering a valuable avenue for exploration in this domain. Additionally, researchers have delved into the utilization of Gradient Boosting algorithms. This method seeks to approximate a function of weights on weaker classifiers, such as Decision Trees, with the aim of minimizing the loss function. Despite these advancements, the integration of deep learning algorithms into mortgage datasets remains relatively underexplored. Notably, the CNN algorithm stands as a promising candidate due to its adeptness in analyzing vast and intricate datasets while excelling in feature extraction. It is noteworthy that prior studies have predominantly focused on either mortgage default prediction neglecting the recommendation of amounts.

6.4 CNN Model for Mortgage Default Prediction and Amount Recommendation

This research led to the construction of a CNN model, leveraging a dataset comprising various customer data attributes. These attributes encompass essential details such as the approved loan amount (LOAN), outstanding mortgage dues (MORTDUE), current property value (VALUE), and the reason behind loan requests, categorized as either Home Improvement (HomeImp) or Debt Consolidation (DebtCon). Additionally, the dataset includes information on the applicant's occupation (JOB), years in their current job (YOJ), and financial indicators like the number of derogatory reports (DEROG) and delinquent credit lines (DELINQ). Other factors such as the

age of the oldest credit line (CLAGE), recent credit inquiries (NINQ), existing credit lines (CLNO), and the debt-to-income ratio (DEBTINC) are also considered. Each of these attributes contributed to the comprehensive understanding of a customer's financial profile and creditworthiness. By training the CNN model on this diverse dataset, the study aims to predict outcomes related to loan approvals and assess customers' suitability for credit.

6.5 Performance of the Developed Model in Predicting Mortgage Default and Amount Recommendation

The model achieved top accuracy of 97.14% prediction accuracy, 94% Precision and 84% recall for mortgage default prediction.

Chapter 7: Conclusion and Recommendations

7.1 Conclusion

In conclusion, the development of the mortgage default prediction tool stands as a testament to the transformative potential of advanced technologies in addressing complex financial challenges. Through the utilization of Convolutional Neural Networks (CNN) and automated feature engineering, a highly accurate and reliable predictive model was created, with an impressive accuracy rate of 97.14% after meticulous hyperparameter tuning. The accessibility of the tool via an API and the user-friendly consumer-facing interface underscores its adaptability and usability across diverse platforms. Moreover, its unique dual functionality, encompassing both mortgage default prediction and proactive risk management through recommended mortgage amounts, sets it apart from conventional models and addresses a critical gap in existing research. By empowering users to make informed decisions and mitigate default risks, this tool heralds a new era of data-driven decision-making in the financial sector. Its superior performance compared to existing models, coupled with its intuitive interface, positions it as a valuable asset for financial institutions and stakeholders alike. Moving forward, the continued refinement and enhancement of such predictive tools hold the potential to revolutionize risk management practices, foster financial stability, and ultimately contribute to a more resilient and robust financial ecosystem. As technology continues to evolve, so too will the capabilities of such tools, ushering in a future where data-driven insights drive informed decision-making and shape the trajectory of the financial landscape. The focus of this study revolved around achieving five specific objectives, all aimed at the overarching goal of creating a tool capable of predicting mortgage default probabilities and providing recommendations for optimal mortgage amounts. These objectives are delineated as follows:

(i) **To identify the factors that leads to mortgage default.**

It can be concluded job and debt to income ration features greatly influence their tendency of customers to default. This was supported both by the literature that was reviewed and also by the findings of this study themselves.

(ii) **To review the techniques used for mortgage default predication.**

This research employed a CNN model to construct a predictive framework for mortgage default, achieving an impressive success rate of 97.14%. The findings suggest that this methodology holds promise for analogous scenarios, underscoring the efficacy of machine learning methodologies in leveraging data to facilitate informed decision-making within organizational contexts.

(iii) **To develop a tool to predict mortgage default and recommend mortgage amount using convolutional neural networks.**

The research endeavour culminated in the successful creation of a predictive model designed to forecast mortgage default probabilities and recommend optimal loan amounts. Additionally, a user-friendly interface was developed to operationalize the model, allowing users seamless access to its predictive capabilities.

(iv) **To test and evaluate the developed system.**

The performance of the model underwent rigorous testing, evaluating its accuracy, recall, and precision parameters. A comparative analysis was conducted against a similar model derived from a separate study, revealing superior performance. Notably, the model achieved an impressive accuracy rate of 97.14%.

7.2 Recommendations

The research has shed light on key attributes that significantly influence mortgage default occurrences, paving the way for actionable recommendations aimed at enhancing risk management strategies within the banking industry:

i). Real-Time Integration: It is recommended that the banking industry seamlessly integrate the developed model into their operational framework to enable real-time prediction of mortgage defaults. By leveraging this model, banks can swiftly identify potential default risks and make informed decisions regarding loan approvals. Moreover, the model can provide personalized recommendations for mortgage amounts tailored to each client's financial profile, fostering responsible lending practices and minimizing default probabilities.

ii). Foundational Risk Management: The banking sector should consider the model as a foundational element in implementing comprehensive risk management strategies. By incorporating the insights gleaned from the model into their risk assessment processes, banks can proactively identify and mitigate potential risks associated with mortgage lending. This proactive approach not only safeguards the financial stability of individual institutions but also contributes to the overall resilience of the banking sector in mitigating systemic risks.

iii). Continuous Model Refinement: To ensure the continued effectiveness of the predictive model, it is imperative for banks to prioritize ongoing refinement. This entails incorporating additional relevant features and fine-tuning algorithms to adapt to evolving market dynamics and enhance predictive accuracy. By continuously refining the model, banks can stay ahead of emerging trends and effectively navigate changing economic landscapes, thereby optimizing their risk management practices.

iv). Collaborative Data Sharing: Facilitate collaborative data sharing initiatives among financial institutions to enrich the predictive capabilities of the model. By pooling anonymized data from diverse sources, banks can access a broader spectrum of insights, leading to more robust risk assessment and mitigation strategies. This collaborative approach fosters collective resilience within the banking industry, enabling institutions to collectively address systemic risks and promote financial stability.

7.3 Future Work

In light of the study's findings, the following are part of future research:

i). Efforts should be directed towards the seamless integration of the developed models into existing banking systems. Such integration would pave the way for more effective and informed decision-making in banking industry.

ii). Exploring the application of other models for mortgage default prediction is a promising avenue for future research. By comparing the performance of these models with the CNN models used in this study, we can gain deeper insights into their respective strengths and weaknesses. Ultimately, this would allow organizations to make more informed decisions regarding which predictive model to utilize based on the specific circumstances at hand. Lastly,

iii). Risk management strategies would form an important part of the future research. Many mortgage default prediction algorithms are static and do not.

7.4 Limitations of the Study

This research addressed critical gaps in the credit scoring literature. Despite the commendable work, the research had the following limitations:

i). The research work focused on mortgage default prediction leaving out other types of loans.

ii). The research work only explored one algorithm and did not compare the performance of the algorithm with other models.

7.5 Research contributions

This study represents a noteworthy advancement in the domain of mortgage default prediction, significantly augmenting existing literature in several key aspects. Firstly, it ventures into the application of state-of-the-art algorithms for prediction tasks, leveraging automated feature engineering techniques to enhance the efficiency of feature selection processes. By minimizing the potential biases inherent in manual feature selection, this approach ensures more robust and accurate prediction models. Secondly, the study introduces novel recommendations regarding the optimal amount to extend to high-risk clients, presenting a strategic solution to mitigate risks for financial institutions when granting mortgages to such clientele. This innovative approach not only enhances risk management practices but also contributes to the overall stability of the lending environment. Moreover, this research represents a ground-breaking endeavour in the utilization of Convolutional Neural Networks (CNNs) for both mortgage predictions and recommendations, marking a significant departure from traditional methodologies. By harnessing the power of CNNs, the study opens new avenues for exploring complex data structures inherent in mortgage datasets, thereby enhancing prediction accuracy and decision-making capabilities. Lastly, the study demonstrates superior performance compared to conventional machine learning models, underscoring the efficacy of its methodologies and the potential for broader applicability in real-world scenarios. This achievement not only validates the efficacy of the proposed approach but also highlights its practical relevance and significance in the domain of mortgage lending. Overall, this study represents a seminal contribution to the

field, offering valuable insights and innovative methodologies that hold promise for advancing research and practice in mortgage default prediction and risk management.

References

Akindaini, B., & Juhola, M. (2017a). *MACHINE LEARNING APPLICATIONS IN MORTGAGE DEFAULT PREDICTION*. https://trepo.tuni.fi/bitstream/handle/10024/102533/1513083673.pdf?sequence=1&isAllowed=y

Akindaini, B., & Juhola, M. (2017b). *MACHINE LEARNING APPLICATIONS IN MORTGAGE DEFAULT PREDICTION*. https://trepo.tuni.fi/bitstream/handle/10024/102533/1513083673.pdf?sequence=1&isAllowed=y

Alushula, P. (2021, July 23). *Mortgage defaults hit Sh70bn, auctions jump*. Business Daily. https://www.businessdailyafrica.com/bd/economy/mortgage-defaults-hit-sh70bn-auctions-jump-3483188

Ameta, R., Solanki, M. S., Benjamin, S., & Ameta, S. C. (2018, January 1). *Chapter 6 - Photocatalysis* (S. C. Ameta & R. Ameta, Eds.). ScienceDirect; Academic Press. https://www.sciencedirect.com/science/article/pii/B9780128104996000061

Bhutta, N., Dokko, J., & Shan, H. (2010). *The Depth of Negative Equity and Mortgage Default Decisions*. https://www.federalreserve.gov/pubs/feds/2010/201035/201035pap.pdf

Ch, V. V. R. K., & Suman , M. (2014). *Download Limit Exceeded*. Citeseerx.ist.psu.edu. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.679.162&rep=rep1&type=pdf

Cheruiyot, T. (2015). *NON-PERFORMING LOANS AND FINANCIAL PERFORMANCE OF COMMERCIAL BANKS IN KENYA*. https://ir-library.ku.ac.ke/bitstream/handle/123456789/19312/Non-

performing%20loans%20and%20financial%20performance%20of%20commercial%E2%80%A6.pdf?sequence=1&isAllowed=y

Del Fiol, G., Hanseler, H., Crouch, B., Cummins, M., & Nelson, S. (2016). Software prototyping. *Applied Clinical Informatics*, *07*(01), 22–32. https://doi.org/10.4338/aci-2015-07-cr-0091

DiPietro, R., & Hager, G. D. (2020). *Recurrent Neural Network - an overview | ScienceDirect Topics*. Www.sciencedirect.com. https://www.sciencedirect.com/topics/engineering/recurrent-neural-network

Fabozzi, F. J., Bhattacharya, A. K., & Berliner, W. S. (2010). *Mortgage-Backed Securities*. John Wiley & Sons.

Foote, C. L., Gerardi, K., & Willen, P. S. (2008). Negative equity and foreclosure: Theory and evidence. *Journal of Urban Economics*, *64*(2), 234–245. https://doi.org/10.1016/j.jue.2008.07.006

Goodman, L. S., Ashworth, R., Landy, B., & Yin, K. (2010). Negative Equity Trumps Unemployment in Predicting Defaults. *The Journal of Fixed Income*, *19*(4), 67–72. https://doi.org/10.3905/jfi.2010.19.4.067

Gurucharan, M. (2020, December 7). *Basic CNN Architecture: Explaining 5 Layers of Convolutional Neural Network*. UpGrad Blog. https://www.upgrad.com/blog/basic-cnn-architecture/

Indolia, S., Goswami, A. K., Mishra, S. P., & Asopa, P. (2018). Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach. *Procedia Computer Science*, *132*, 679–688. https://doi.org/10.1016/j.procs.2018.05.069

Kvamme, H., Sellereite, N., Aas, K., & Sjursen, S. (2018). Predicting mortgage default using convolutional neural networks. *Expert Systems with Applications*, *102*, 207–217. https://doi.org/10.1016/j.eswa.2018.02.029

Lee, Ohanian, E., Gerardi, K., Herkenhoff, K., Ohanian, L., & Willen, P. (2012). *Unemployment, Negative Equity, and Strategic Default*. https://www.urban.org/sites/default/files/2015/02/16/gerardi-kerkenhoff-ohanian-willen-strategic-default.pdf

Leow, M., & Mues, C. (2012). Predicting loss given default (LGD) for residential mortgage loans: A two-stage model and empirical evidence for UK bank data. *International Journal of Forecasting*, *28*(1), 183–195. https://doi.org/10.1016/j.ijforecast.2011.01.010

Li, B. (2022). Online Loan Default Prediction Model Based on Deep Learning Neural Network. *Computational Intelligence and Neuroscience*, *2022*, 1–9. https://doi.org/10.1155/2022/4276253

Martha, N., & Daniel, W. (2014). An Assessment of Mortgage Loan Uptake among Bank Staff: A Survey of Commercial Banks in Nakuru Town. In *International Journal of Science and Research*. IJSR. https://www.ijsr.net/archive/v3i10/T0NUMTQ3NTI=.pdf

Martin, M. (2019, October 24). *Prototyping Model in Software Engineering: Methodology, Process, Approach*. Guru99.com. https://www.guru99.com/software-engineering-prototyping-model.html

Nguyen, G., Dlugolinsky, S., Bobák, M., Tran, V., López García, Á., Heredia, I., Malík, P., & Hluchý, L. (2019). Machine Learning and Deep Learning frameworks and libraries for large-scale data mining: a survey. *Artificial Intelligence Review*, *52*(1), 77–124. https://doi.org/10.1007/s10462-018-09679-z

Nyang'iye, S. A., Iraya, C., & Ochieng, D. E. (2022). Residential Mortgage Portfolio, Product Innovation and Performance of Commercial Banks in Kenya. *European Journal of Business and Management Research*, *7*(3), 184–193. https://doi.org/10.24018/ejbmr.2022.7.3.1439

Object, object. (2017). FACTORS AFFECTING DEMAND FOR MORTGAGE LOANS IN KENYA: A CASE OF I&M BANK LIMITED, KENYA. *Core.ac.uk*. https://core.ac.uk/reader/224836780

Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Computer Science*, *2*(3), 1–21. Springer. https://doi.org/10.1007/s42979-021-00592-x

Siaw, A., Ntiamoah, B., Oteng, E., & Opoku, B. (2014). An Empirical Analysis of the Loan Default Rate of Microfinance Institutions. *Online)*, *6*(22). https://core.ac.uk/download/pdf/234625689.pdf

Wahab, M. H. A. (2018). *Figure 4: Prototype Methodology*. ResearchGate. https://www.researchgate.net/figure/Prototype-Methodology_fig5_268376151

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, *9*(4), 611–629. https://doi.org/10.1007/s13244-018-0639-9

Zaharieva, Prof. M., & Ignatov, B. (2019). *Bayesian Variable Selection and Model Averaging for modelling the Probability of Default of mortgage portfolios*. Erasmus University Rotterdam. https://thesis.eur.nl/pub/49581/Econometrics_Master_Thesis_Selm.pdf

Zhang, Q. (2015). *MODELING THE PROBABILITY OF MORTGAGE DEFAULT VIA MODELING THE PROBABILITY OF MORTGAGE DEFAULT VIA LOGISTIC*

REGRESSION AND SURVIVAL ANALYSIS LOGISTIC REGRESSION AND SURVIVAL ANALYSIS. https://digitalcommons.uri.edu/cgi/viewcontent.cgi?article=1543&context=theses

Zhou, X., Zhang, W., & Jiang, Y. (2020). Personal Credit Default Prediction Model Based on Convolution Neural Network. *Mathematical Problems in Engineering*, *2020*, 1–10. https://doi.org/10.1155/2020/5608392

Appendices

Appendix A: Ethical Clearance

Strathmore UNIVERSITY

18th September 2023

Mr Okola Dan Naftali,
naftali.okola@strathmore.edu

Dear Mr Okola,

RE: A Tool to Predict Mortgage Default and Recommend Mortgage Amount using Convolution Neural Networks

This is to inform you that SU-ISERC has reviewed and **approved** your above SU-masters research proposal. Your application reference number is **SU-ISERC1862/23.** The approval period is from **18th September 2023 to 17th September 2024.**

This approval is subject to compliance with the following requirements:

i. Only approved documents including (informed consents, study instruments, MTA) will be used.
ii. All changes including (amendments, deviations, and violations) are submitted for review and approval by SU-ISERC.
iii. Death and life-threatening problems and serious adverse events or unexpected adverse events whether related or unrelated to the study must be reported to SU-ISERC within 72 hours of notification.
iv. Any changes anticipated or otherwise that may increase the risks or affected safety or welfare of study participants and others or affect the integrity of the research must be reported to SU-ISERC within 72 hours.
v. Clearance for the export of biological specimens must be obtained from relevant institutions.
vi. Submission of a request for renewal of approval at least 60 days prior to the expiry of the approval period. Attach a comprehensive progress report to support the renewal.
vii. Submission of an executive summary report within 90 days of completion of the study to SU-ISERC.

Before commencing your study, you will be expected to obtain a research license from National Commission for Science, Technology, and Innovation (NACOSTI) https://research-portal.nacosti.go.ke/ and obtain other clearances needed.

Yours sincerely,

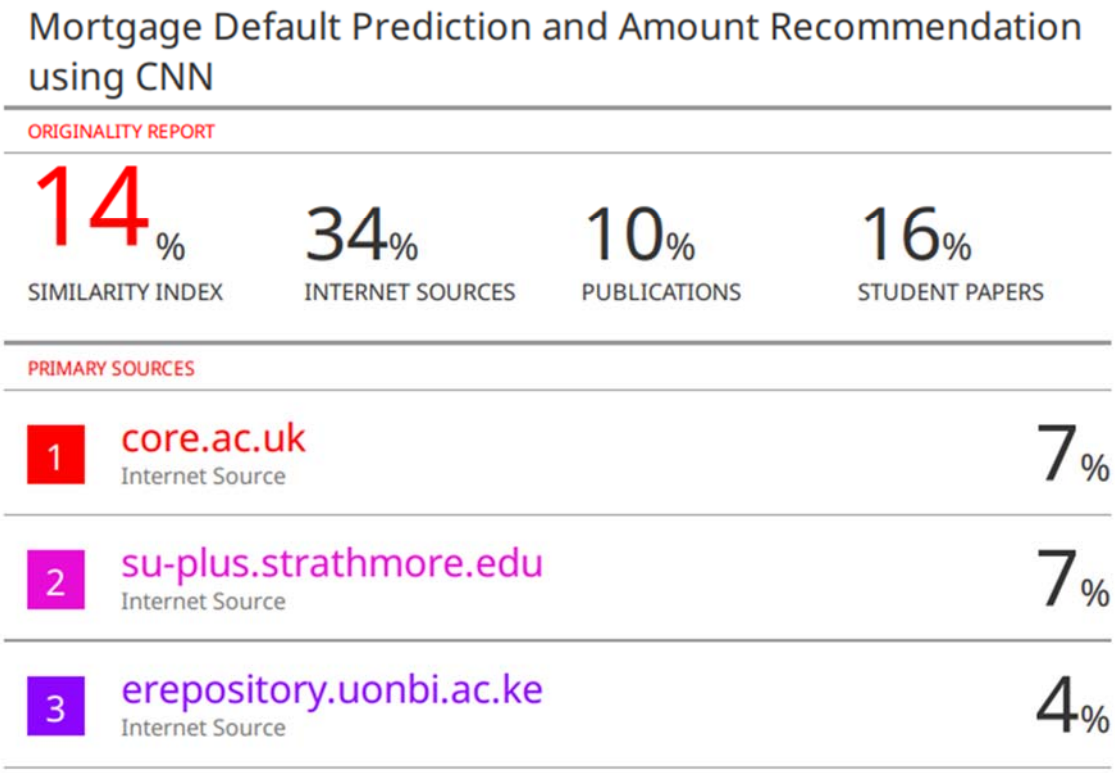
Mr Ambrose Rachier,
Chairperson; SU-ISERC

Ole Sangale Rd, Madaraka Estate. PO Box 59857-00200, Nairobi, Kenya. Tel +254 (0)703 034000
Email admissions@strathmore.edu www.strathmore.edu

Ethical clearance was secured to obtain approval for the continuation of the research endeavour.

Appendix B: Gantt chart

Appendix C: Turnitin Report



Mortgage Default Prediction and Amount Recommendation using CNN

ORIGINALITY REPORT

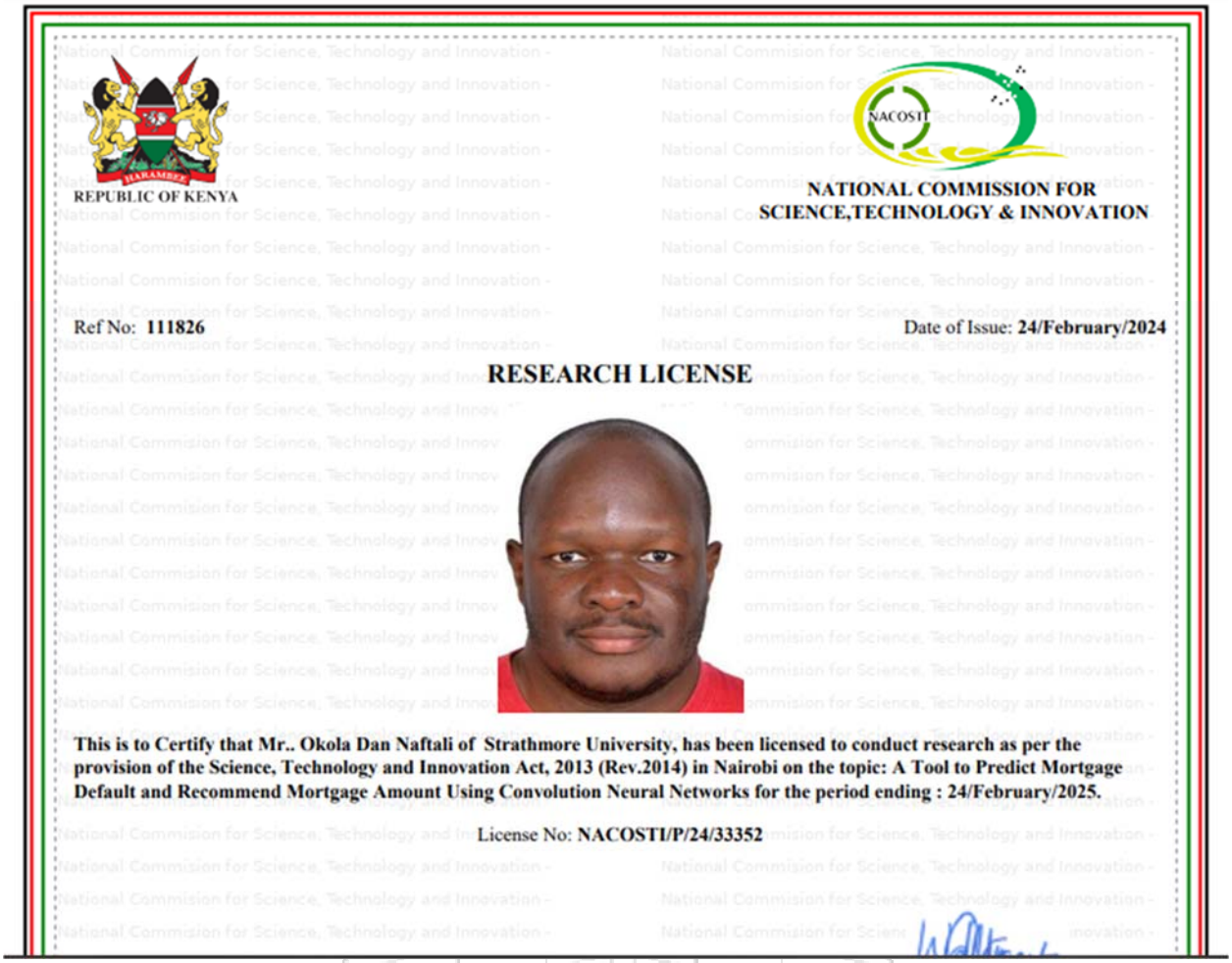
14%	34%	10%	16%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	core.ac.uk Internet Source	7%
2	su-plus.strathmore.edu Internet Source	7%
3	erepository.uonbi.ac.ke Internet Source	4%

Appendix C shows the plagiarism report.

Appendix D: NACOSTI License



REPUBLIC OF KENYA

NATIONAL COMMISSION FOR SCIENCE,TECHNOLOGY & INNOVATION

Ref No: **111826**

Date of Issue: **24/February/2024**

RESEARCH LICENSE

This is to Certify that Mr.. Okola Dan Naftali of Strathmore University, has been licensed to conduct research as per the provision of the Science, Technology and Innovation Act, 2013 (Rev.2014) in Nairobi on the topic: A Tool to Predict Mortgage Default and Recommend Mortgage Amount Using Convolution Neural Networks for the period ending : 24/February/2025.

License No: **NACOSTI/P/24/33352**

The NACOSTI (National Commission for Science, Technology and Innovation) license is required for research activities conducted in Kenya. NACOSTI oversees and regulates research initiatives to ensure that they adhere to ethical standards, promote scientific integrity, and contribute positively to national development goals. Obtaining a NACOSTI license is necessary for researchers and institutions planning to conduct scientific research, experiments, or studies involving human participants, animals, or genetic materials within Kenya. The license serves as a formal authorization from the government, indicating compliance with regulatory requirements and ensuring that research activities are conducted responsibly and ethically.

Appendix E: Login Code

```php
protected function authenticated(Request $request)
{
    $type = \Auth::user()->acc_type;

    if($type == 2)
    {
        return redirect('admin/dashboard');
    }

    if($type == null || $type == 0)
    {
        return redirect('login');
    }
}
```

The code above authenticates user requests and redirects them to the dashboard after the authentication is successful.

Appendix F: Model Training Code (Default Prediction)

```
# Build a CNN model
model = Sequential()
model.add(Conv1D(32, 3, activation='relu', input_shape=(x_train.shape[1], 1)))
model.add(MaxPooling1D(2))
model.add(Conv1D(64, 3, activation='relu'))
model.add(MaxPooling1D(2))
model.add(GlobalAveragePooling1D())
model.add(Dense(1, activation='sigmoid'))

# Add fully connected layers
model.add(Dense(128, activation='relu'))
model.add(Dense(1, activation='sigmoid'))  # Binary classification, so using sigmoid activation

# Compile the model
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
model.fit(x_train, y_train, epochs=10, batch_size=32, validation_split=0.2)

# # Reshape the input data if necessary (assuming x_train is a NumPy array)
# x_train_reshaped = x_train.reshape(x_train.shape[0], x_train.shape[1], 1)
```

The code constructs a CNN model architecture for binary classification task which is predicting mortgage default or not, compiles it, and trains it on the provided training data.

Appendix G: Model Training Code (Amount Recommendation)

```
# Build the CNN model to recommend amount
model = Sequential()
model.add(Conv1D(32, 3, activation='relu', input_shape=(x_train.shape[1], 1)))
model.add(MaxPooling1D(2))
model.add(Conv1D(64, 3, activation='relu'))
model.add(MaxPooling1D(2))
model.add(GlobalAveragePooling1D())
model.add(Dense(1, activation='sigmoid'))

# Add fully connected layers
model.add(Dense(128, activation='relu'))
model.add(Dense(1, activation='sigmoid'))  # Binary classification, so using sigmoid activation

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

The code snippet above constructs a CNN model for recommending mortgage amounts.