ሀዋሳ ዩኒቨርሲቲ

HAWASSA UNIVERSITY

CLASSIFYING EFFECT OF E-BANKING SERVICE ON DEPOSIT MOBILIZATION USING MACHINE-LEARNING TECHNIQUES

MSc. Thesis

By:

BALCHA BEKELE

Hawassa, Ethiopia

May 2024

CLASSIFYING EFFECT OF E-BANKING SERVICE ON DEPOSIT MOBILIZATION USING MACHINE-LEARNING TECHNIQUES

BY

BALCHA BEKELE

ID No. GPCoScW/004/13

MAJOR ADVISOR: DEGIF TEKA (PhD)

A THESIS SUBMITTED TO DEPARTMENT OF COMPUTER SCIENCE FACULITY OF INFORMATICS IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN COMPUTER SCIENCE

INSTITUTE OF TECHNOLOGY

HAWASSA UNIVERSITY

HAWASSA UNIVERSITY

HAWASSA, ETHIOPIA

May 2024

APPROVAL SHEET-I

This is to certify that the thesis entitled, "Classifying Effect Of E-Banking Service on Deposit Mobilization Using Machine-Learning Techniques" submitted in partial fulfilment of the requirements for the degree of Master's with specialization in Computer Science, the Graduate Program of the Faculty of Informatics, and has been carried out by Balcha Bekele. Therefore, we recommend that the student has fulfilled the requirements and hence hereby can submit the thesis to the department.

Degif T. (PhD)	_____	_____
Name of major advisor	Signature	Date

Efrem A.	_____	_____
Name of co-advisor	Signature	Date

EXAMINERS' APPROVAL SHEET

SCHOOL OF GRADUATE STUDIES

HAWASSA UNIVERSITY EXAMINERS' APPROVAL SHEET

(Submission Sheet-2)

We, the undersigned, members of the Board of Examiners of the final open defence by Balcha Bekele have read and evaluated his/her thesis entitled "Classifying Effect Of E-Banking Service on Deposit Mobilization Using Machine-Learning Techniques", and examined the candidate. This is, therefore, to certify that the thesis has been accepted in partial fulfilment of the requirements for the degree.

Dr. Degif Teka	_____	_____
Name of Major Advisor	Signature	Date
Tegegn Gobena (Asst. Prof)	_____	_____
Name of Internal Examiner-I	Signature	Date
Gizaw Melese (MSc)	_____	_____
Name of Internal Examiner-II	Signature	Date
Dr. Eng. Mesay Adinew		27/06/2024
Name of External examiner	Signature	Date
_____	_____	_____
SGS Approval	Signature	Date

Final approval and acceptance of the thesis is contingent upon the submission of the final copy of the thesis to the School of Graduate Studies (SGS) through the Department/School Graduate Committee (DGC/SGC) of the candidate's department.

Stamp of SGS Date: _____

Remark

- Use this form to submit the thesis with ***MINOR CORRECTION*** suggested by the examining board
- 3 copies

ABSTRACT

Identifying services that are more likely potential to E-banking product offering is an important issue. Cooperative Bank of Oromia S.C., being one of the former private banks in Ethiopia is offering E-Banking products. The main objective of this study is to apply machine learning algorithms for developing Deposit mobilization Performance prediction Model that forecast potential of E-banking channel service in Cooperative Bank of Oromia.

This research follows experimental research. For modelling purpose, data was gathered from the institution head office. Since irrelevant features result in bad model performance, data pre-processing was performed in order to determine the inputs to the model.

This thesis investigates the creation and assessment of six machine learning algorithms to forecast deposit behavior from customers: CART, SVM, KNN, Naïve Bayes, Logistic Regression and Random Forest. Cross tables were used to show the results of precision calculations and confusion matrices used to evaluate the performance of these models. With an emphasis on the relevance of various attributes in predicting customer deposits, the suitability of various classification algorithms, the relative effectiveness of ensemble versus base learning models, and forecasting based on influential attributes, the study tackled three main research questions.

Experimental results exhibit that, the ensemble learning model achieved 98.496% accuracy in categorizing deposits, outperforming individual algorithms like KNN (98.491%) and SVM (98.401%), emphasizing the superiority of ensemble methods for deposit mobilization prediction. Random Forest Classifier identified "other_debit," "gender," and "mobile banking" as the most significant predictors of deposit mobilization, with relevance scores of 20%, 18%, and 13% respectively. Moderately important features included "mobile_credit", "mobile_debit", "card_debit", and "marital_status", while "atm_card" and "other_credit" were negligible.

Finally, this thesis shows the effectiveness of machine learning in financial prediction by offering a thorough comparison of six popular categorization methods. The result offer valuable insights for enhancing customer deposit strategies at CBO and potentially other banking institutions.

Key words: Ensemble learning, Deposit Mobilization, E-Banking

TABLE OF CONTENTS

LIST OF FIGURE

LIST OF TABLES

ACRONYMS

ATM	Automatic Teller Machine
ANN	Artificial Neural Networks
CART	Classification and Regression Tree
CBO	Cooperative Bank of Oromia
CRISP-DM	Cross-Industry-Standard-Process for Data Mining
CRM	Customer Relationship Management
DSS	Decision Support System
GUI	Graphical User Interface
K-NN	K-Nearest Neighbor
MAR	Missing At Random
MCAR	Missing Completely At Random
MNAR	Missing Not At Random
ML	Machine Learning
SMOTE	Synthetic Minority Oversampling Technique
SVM	Support Vector Machine
USSD	Unstructured Supplementary Service Data
UTAUT2	Unified Theory of Acceptance and Use of Technology
WAP	Wireless Application Protocol
WEKA	Waikato Environment for Knowledge Analysis

ACKNOWLEDGEMENT

I would like to express my heartfelt appreciation to Dr. Degif for his guidance, expertise, and unwavering support throughout this research. His invaluable insights and constructive feedback have greatly contributed to the success of this thesis.

I would also like to extend my deepest gratitude to my wife, Tigist Mebrate, for her unwavering support and assistance throughout every step of this journey. Her constant encouragement and understanding have been invaluable to me.

Additionally, I want to express my gratitude to Doni Abera and Dawit Bekele, the Cooperative bank of Oromia, Database Administrators, for their invaluable assistance in collecting the data for this thesis. Their dedication and expertise have played a significant role in the successful completion of this research.

CHAPTER ONE

1 INTRODUCTION

1.1 Background of the Study

Economic growth is the common goal of all nations. Everybody lives with a more comfortable, better standard of living than before and holds better welfare because of the surge in economic growth. The government in each country aims to reduce poverty and increase the level of national income. Therefore, to achieve the main target of economic growth, governments may implement various kinds of policies such as encouraging saving and stimulating investment and production in their countries (Pinchawawee, 2011).

Mobilizing deposits is one of the essential issues in developing countries as domestic funds provide cheap and reliable sources of funds for development, which is of great value to these countries, especially when the economy has difficulty raising capital from international donors, investors, and markets. Yet, in many developing countries, there is a considerable amount of savings that are not intermediated through the formal sector particularly there exists significant savings potential in the rural (and/or semi-urban) sector of many developing countries.

Selvaraj & Kumar (2015) State that, the success of the banking industry greatly depends on the degree of deposit mobilization. Similarly, Performances of the bank depend on deposits, as the deposits are normally considered as a cost-effective source of working funds. In addition, enhancing the efforts of bank deposit mobilization can optimize rural savings through e-banking technologies is one of the key objectives of commercial Banks. As promotion of e-banking services to these areas will expand banking operations leading to increased deposits. This is linked to the successful functioning of commercial banks as it is attributed to the extent of funds mobilized.

Deposits constitute a vital source of funds required for providing loans and carrying out vital banking business functions. As banks maintain different types of deposits, with different maturity patterns carrying different rates of interest. Mobilization of deposits by banks can be considered crucial for banks' profitability and existence as oxygen is essential for sustaining human beings.

Nowadays, Machine Learning (ML) algorithms and techniques have been used to solve various financial problems in the banking domain following the existence of high computing power, and the availability of huge amounts of financial data gathered from e-banking transactions and services (Gafrej, 2023). Machine learning uses computational methods and learns new patterns from the existing data to have more experience in improving its performance or making accurate predictions. With the help of ML tools and algorithms business organizations like banks can predict future trends and behaviors, allowing businesses to make proactive and knowledge-driven decisions.

Currently, the banking industry is aggressively investing in implementing core banking solutions that deliver e-banking services to its customers in more efficient and effective ways. Using banking technologies customers can conduct financial transactions electronically without the need to physically appear in the bank premises (Abiodun, 2021). Besides, customers are able to deposit, transfer, make payments from their bank accounts more easily and conveniently using e-banking services from their personal computer or mobile phones. (Abiodun, 2021). Besides, banks like CBO, using e-banking technologies can deliver fast and reliable services such as online account opening, viewing accounts and their balance, allow online transfer from their account and manage customers there by leading to large deposit.

Deposit mobilization is the collection of cash or funds by a financial institution from the public through its current, savings, fixed, recurring accounts and other specialized schemes. Mobilization of deposit for a bank is as essential as oxygen for human being. Deposit mobilization is one of the main functions of banking business and so an important source of working fund for the bank. Since deposits are normally considered as a cost effective source of working fund, the bank's ability to lend more as well as its success greatly lies on its deposit mobilization.

Machine Learning methods can be considered of great assistance to banks for prediction, forecasting and decision-making. One particular method is the investigation of classification rules between products and services a bank offers. These rules used as additional tools aiming at the continuous improvement of bank services and products helping also in approaching new customers. In addition, the development and continuous training of machine learning is a technique which use to extract vital information from existing huge amount of data and enable better

decision-making for the banking industries. In order to merge heterogeneous data from databases into a format that is suitable for learning and testing machine algorithms, they use a variety of available data from numerous sources to classify according to a specific problem. The data is then analyzed and the information that is captured is used throughout the organization to support decision-making activities and other significant tasks, especially for bank organizations.

Machine learning can be used by deposit mobilization to find new customers, predict customer behavior and preferences, and tailor marketing efforts (Ozgur & Ozbugday, 2021). For instance, customer data can be examined using machine learning algorithms to identify patterns that indicate which customers are most likely to make deposits into their accounts (Ozgur & Ozbugday, 2021). Banks may then use this information to target these customers with customized marketing campaigns that offer them incentives like low interest rates. (Ozgur & Ozbugday, 2021). Similarly, machine learning can be used to detect fraudulent activities and prevent money laundering. Another method that machine learning might assist banks in more quickly and efficiently allocating deposits is by automating the process of opening new accounts through e-banking. (Orlova, 2020).

As the main business for banks is accepting deposits and granting loans. The more the loans the banks disburse the more profit they make. Also, banks do not have a lot of their own money to give as loans. They depend on customer deposits to generate funds for granting loans to potential borrowers.

Hence, the efforts of banks to mobilize more deposits can be improved through the expansion and delivery of e-banking services to their customers and by applying appropriate machine learning models to identify variables that would lead to better mobilization of profit. Therefore, classifying e-banking service variables that lead to better deposit mobilization, building models, and comparing their respective performance accuracy using machine learning algorithms for CBO will be the main objective of this study.

1.2 Motivation of the Study

Deposit mobilization is a method by which banks obtain deposits from their clients through the delivery of deposit services. It is a crucial service that is an essential component of the banking business. Around the world, banks and financial institutions are striving to mobilize more deposits to achieve their business objectives and survive in the financial market. Without enough deposits, banks and financial institutions might fail to achieve their business targets. Similarly, the issue of deposit mobilization is becoming a top priority agendum in Ethiopian banks and other financial institutions as the number of banks in Ethiopia is increasing drastically. This would have a significant negative impact on the respective bank's ability to mobilize deposits as customers might have many alternatives to get banking services.

Therefore, Banks in Ethiopia have paid sufficient attention and started investing in recent innovative banking solutions to provide e-banking services such as e-banking, mobile banking, mobile payment, e-wallet, and e-money services to attract more customers and enhance their deposit mobilization efforts. However, the acceptance of these services among customers is still sluggish, and needs to identify key variables that should lead to maintaining existing customers, attracting potential ones, and increasing their deposits.

To achieve these objectives, banks should identify the key e-banking variables that lead them to better mobilize deposits. One of the many strategies cooperative banks employ around the world to issue banking deposits is designing unique service delivery strategies. In this regard, Ethiopian banks including CBO face challenges in mobilizing efforts to increase their bank deposit. Hence, identifying e-banking and customer attributes that contribute more to deposit mobilization early, and applying Machine learning algorithms to train, test, evaluate, and compare the performances of each model leads to significant improvement in the mobilization of deposits.

Therefore, the main motivation of this study is to classify e-banking services and customer attributes that contribute more to bank deposit mobilization and apply machine learning algorithms to model, test, and compare their respective deposit mobilization performance prediction in the case of the Cooperative Bank of Oromia. The classification models developed from this study might be applied to other banks in providing some useful insight into the factors that influence

customers' preferences to adopt e-banking services that drive customers to make deposits at a particular bank to shed new light.

1.3 Statement of the Problem

The private banks in Ethiopia are in dire need of these resources (deposits) to satisfy their demands more than ever and they are engaged in stiff competition to acquire these resources. To beat the intense competition of the industry and to meet customers' need, the Bank launches different electronic banking services. However, those electronic services may have an adverse effect on deposit mobilization effort of the Bank and there are factors which have an impact on deposit mobilization and knowing these factors and their level of significance.

Current efforts to mobilize deposits are not successful. Usefulness of this research is to identify factors that can increase deposit mobilization performance of the bank. The research will address the following specific research questions.

- What are the most significant E-banking customer and service parameters or attributes that influence deposit mobilization performance?
- Which Machine-learning algorithms best learn from the existing features of the dataset and outperform the deposit mobilization performance?
- How can we compare and forecast the performances of the compared classification algorithms based on selected features that influence deposit mobilization performance?

1.4 Objectives of the Study

1.4.1 General Objective

The general objective of the study is to evaluate and compare the accuracy of machine learning algorithms for classifying deposit mobilization using data from E-banking services.

1.4.2 Specific Objective

To achieve the general objective of the study, the following specific objectives are stated:

- Identify the optimal set of features that can be used to accurately forecast deposit mobilization performance.
- Evaluate the performance of various machine learning algorithms in predicting deposit mobilization outcomes.
- Identify the optimal set of features that can be used to accurately forecast deposit mobilization performance.

1.5 Scope and Limitation of the Study

The scope of the study refers to the boundaries within which the research is conducted, while limitations are the factors that may have affected the study's findings or its ability to address certain aspects.

With respect to the quantitative method of investigation, the scope of the study is limited to one private bank of Ethiopia Cooperative Bank of Oromia S.C, due to unavailability of sufficient time to include more private bank incorporated with this data mining analysis and to develop the prototype of the model.

It is essential to consider these scope and limitations when interpreting the study's findings and when applying them to other contexts or organizations. Future research can build upon these limitations by expanding the scope, incorporating a larger sample size, and addressing other contextual factors to enhance the overall understanding of the relationship between e-banking services and deposit mobilization.

1.6 Significance of the Study

The private banking sector in Ethiopia must increase their deposits by overcoming the existing challenges; hence they need to know the factors that determine deposit which of those factors are influential and what the related costs of mobilizing deposits activity are and identify and analyze associated costs. The purpose of this research is to build a model by applying the machine learning technique that forecasts organizational success in deposit mobilization. The research exploits the opportunity of big data that can be processed and utilized to enhance business activities.

The private banking industry in Ethiopia needs deep insight into (knowledge extraction using machine learning) and has some demand of study on the current deposit problem faced by many banks in Ethiopia. In relation to deposit problem the effect of the currently added electronic services such as Mobile banking, Internet banking and Card transactions on the deposit score of the banks need to be studied. The selected private bank for this research work is Cooperative Bank of Oromia S.C as source of data and focus of study. The study's holds significance for organizations, managers and customers by informing strategic decision-making, enhancing operational efficiency and improving customer experience.

1.7 Ethical Considerations

The data collected from the bank used for this study only. The personal identity never been exposed to third party by any means, and would handle in a professional manner. Moreover, all major principles of ethical considerations and standards have noted during all phases of the study process that includes, but not limited to, respect for participants, respect to the organizations involved, keep confidentiality and avoiding any acts of misbehavior like intentional misinterpretations and using results for harming the whole or part of the society.

1.8 Organization of the Study

This thesis organized into five chapters described as follows.

Chapter Two exhaustively review about effects of e-banking service deposit mobilization with its related issues, and related works and their trends in Ethiopia. Brief explanation of machine learning techniques in the area presented and relevant machine-learning model chosen for the study. Chapter Three presents the detail of proposed research model along with a research method and research question that have mentioned above. The chapter specify the data preparation and experiment, analyze data with data collection procedure of the study. Moreover, the methodology, which would be employed for conducting the research, is organized in this chapter. Chapter Four describes all about the thesis findings and discussions based on the experiment on training set and test data sets on the applying python software with in different machine learning algorithms. This chapter also presents evaluation of model, which is the detail indicator variables accuracy and sensitivity was discussed. Therefore, the analysis and interpretation of the research findings

incorporated in this chapter. Chapter Five conclude all the key points of the study, the implication of the study, limitations of the study, presents some recommendations and future research directions. Finally, the study ends with appending Reference lists and Appendix.

CHAPTER TWO

2 LITERATURE REVIEW AND RELATED WORK

2.1 Introduction

In this chapter, we will look at how crucial a role e-banking services play in mobilizing deposits for banks and financial institutions. This chapter also examines recent literature conducted on e-banking services and their role in mobilizing deposits as well as the application of machine learning (ML) algorithms for determining e-service features and attributes that contribute most to building classification models applied to deposit mobilization tasks. For this, a very popular ensemble approach for classification will also be discussed. Classification is a machine learning technique used to predict group membership for data instances. The techniques used as part of an ensemble approach for classification namely as Logistic Regression (LR), Decision Trees (cart), Support Vector Machines (SVM), K- Nearest Neighbours (KNN), and Naive Bayes are also be discussed.

2.2 Cooperative Bank of Oromia S.C.

Bank is a type of organization dedicated to receive deposits and provide loans to its customers. Banks can also engage in service provision relating to finance like management of wealth, exchange of currencies, and safe deposit boxes. To date, Bank types are classified into two groups namely Commercial and Investment. Usually, banks are governed by regulations and commands emanating from governments or Central Banks. Major responsibilities of banks include stabilizing currency, inflation and monetary policy control, and supervision of the overall supply of money (CBO, 2022)

The history of cooperative banks has been traced back to the financial exclusion faced by many communities in the 19th century. With the industrial revolution, the emerging financial services

sector was primarily focused on individuals and large enterprises in urban areas. The rural population, particularly farmers, small businesses, and the communities they supported, were excluded from financial services. Thus, cooperative banks were originally set up to correct this market failure and to overcome the associated problems of asymmetric information in favor of borrowers (CBO, 2022)

In respect of Ethiopia, the country has very low financial service coverage as mainstream financial institutions are heavily tilted towards the urban centers with good physical infrastructure, leaving the rural areas underserved. Traditionally, 'Equbs' and 'Idirs' are informal institutions that are deeply ingrained in the life of communities and have also been serving financial needs of the rural society to some extent. Reluctance and low capacity of the formal financial institutions in the country to serve rural community, a demand-supply gap prevailed in financial market especially in rural areas, coupled with farmers' awareness to be organized into cooperatives and the increasing need to finance cooperatives called for establishment of a cooperative bank. Furthermore, finance appeared to be the critical bottleneck to sustain the cooperative institutions and ultimately the farmers. It was all these glitches that initiated the inception and establishment of Cooperative Bank of Oromia (CBO, 2022).

Haile Gebre Lube, regarded as the founding father (proponent) of Ethiopia's cooperatives, brought the idea of founding the bank for he believed that the best way to fight poverty is through cooperation. Formally establishing a project office in 2002, the bank's formation was realized with majority of shareholders being the cooperative societies. The bank then is commercially licensed in October 2004 and commenced operations in March 2005. As there is no legal provisions that allow establishment of a cooperative bank in the country, the bank was registered in accordance with article 304 of the commercial code of Ethiopia (CBO, 2022).

The Bank has broad ownership base and diversified ownership structure. It comprises Cooperatives and Non-Cooperative members. Cooperative member includes Primary Cooperatives, Cooperative Unions and Cooperatives Federation whereas, non-Cooperative members includes Organizations, Associations and individuals (CBO, 2022). The Coopbank (Cooperative Bank of Oromia) now has a total asset value of more than ETB 121 billion. The bank has 610+ branch networks, 10 million account holders, and more than 11,500 employees in Ethiopia (CBO, 2022). With regard to Coopay-Ebirr's(Coop Electronic birr payment) digital

ecosystem performance during the fiscal year, 4,812 new agents, 24,619 new merchants, and 1.56 million new Coopay-Ebirr subscribers were recruited. This puts the total agents, and merchant subscribers of the bank at 7,351 and 46,827, respectively, while the bank's Coopay-Ebirr subscribers stood at a staggering 3.52 million at the end of June 2022. On the other hand, the total number of ATM card users has reached 401,772 at the end of June 2022 (CBO, 2022). Coop bank have made significant progress in incremental deposit mobilization with ETB 25.65 billion during fiscal year 2021/22. By the end of June 30, 2022, the bank's aggregate deposit was at ETB 96.77 billion signifying a 36 percent increase over the previous year's deposit (CBO, 2022).

2.3 Overview of Electronic Banking for Deposit Mobilization

In today's globalization and cutthroat competition, the banks are struggling to gain a competitive edge over each other. Apart from execution of business processes, the creation of knowledge base and its utilization for the benefit of the bank is becoming a strategy tool to compete. In recent years, the ability to generate, capture and store data has increased enormously. The information contained in this data can be very important. The wide availability of huge amounts of data and the need for transforming such data into knowledge encourage IT industry to use machine learning. The banking industry around the world has undergone a tremendous change in the way business is conducted. The banking industry has started realizing the need of the techniques like data mining and machine learning, which can help them to compete in the market. Leading banks are using machine-learning tools for customer segmentation and profitability, credit scoring and approval, predicting payment default, marketing, detecting fraudulent transactions, etc. This chapter provides an overview of the concept of machine learning and highlights the applications of machine learning to enhance the performance of some of the core business processes in banking industry which is deposit mobilization.

In the financial services industry throughout the world, electronic points of contact to reduce the time, cost of processing an application for various products, and ultimately improve the financial performance are replacing the traditional face-to-face customer contacts. The computerization of financial operations, use of internet and automated software's has completely changed the basic concept of business and the way the business operations are being carried out. The banking sector

is not an exception to it. It has also witnessed a tremendous change in the way the banking operations are carried out (Garo, 2015).

Electronic banking is a form of banking in which funds are transferred through an exchange of electronic signals rather than through an exchange of cash, checks, or other types of paper documents. Electronic banking relies on intricate computer systems that communicate using telephone lines. These computer systems record transfers and ownership of funds, and they control the methods customers and commercial institutions use to access funds. There are various electronic banking systems, and they range in size. An example of a small system is an ATM network, a set of interconnected automated teller machines that are linked to a centralized financial institution and its computer system. An example of a large electronic banking system is the Federal Reserve Wire Network, called Fedwire. This system allows participants to handle large, time-sensitive payments, such as those required to settle real estate transactions. Major opportunities offered by Technological developments are: Reducing cost per transactions; Broadened and easier access to target customers; More efficient system and techniques for dealing with information on customers (CRM); Possibility of diversifying into new business: increase controlling internal processes efficiency; and facilitate decision making (Kumar M. , 2010).

Determining e-banking service factors and attributes that affect the level of deposit mobilization in banks has gained tremendous importance for researchers and the financial industries around the globe. It helps the financial industries to mobilize or collect deposits among their customers by providing e-banking services. However, developing an accurate and effective deposit classification model is a challenging task due to class imbalance and the presence of some irrelevant features.

2.3.1 Types of Electronic Banking

There are many electronic banking delivery channels to provide banking services to customers. The most widely used channels are ATM, POS, Mobile banking, and Internet banking which are listed below.

Automated Teller Machine (ATM): An automated teller machine (ATM) is a machine where customers make cash withdrawals, deposits, funds transfers, or account information inquiries over the machine without going to the banking hall. It also sells recharge cards and transaction funds,

which can be accessed 24 hours/7 days with an account balance enquiry without direct approach to the banking officer (Fenuga, 2010).

There are two widely stated types of ATMs in the literature basic ATM and complex machine ATM. The basic ATM provides cash withdrawal and balance updating services to bank customers whereas the complex machine ATM besides the basic functions, accepts deposits and allows account-to-account transfers.

Mobile Banking: Mobile banking is the subset of e-banking in which customers access a range of banking products using cellular devices. It is used to monitor savings accounts and credit instruments, check account balances, transfer funds between accounts, bill payments, and spot an ATM through electronic channels. Mobile banking requires the customer to hold a deposit account to and from which payments to hold a deposit account to and from which payments or transfers may be made. Mobile banking reduces the transaction costs of payments because there is an electronically accessible state of value (Kumar M. , 2010).

Further mobile banking is a term used for performing balance checks, account transactions, payments credit applications, and other banking transactions through a mobile device such as a mobile phone or personal digital assistant (PDA). Mobile banking services were offered over SMS, a service known as SMS banking. Mobile banking is used in many parts of the world with Insufficient network infrastructure, especially in remote and rural areas. This aspect of mobile Communication is also popular in countries where most of their populations are banked. In most of these places, banks can only be found in big cities and customers have to travel hundreds of miles to the nearest bank. The scope of offered services may include facilities to conduct bank and stock market transactions, administrate accounts, and access customized information. (Kumbhar V. 2011)

Internet Banking: Internet banking allows customers of a financial institution to conduct financial transactions on a secure website managed by the institution, which can be a retail or virtual bank credit union or society. It may include any transactions related to online usage. Banks increasingly operate websites through which customers inquire about account balance, interest, and exchange rates but also access a range of nonfinancial service facilities such as send and follow-up check clearing process. (Arbar T.Timothy, 2012).

Point of Sale Banking (POS): It is also sometimes referred to as point of purchase (POP) or checkout the location where a transaction occurs. POS is a terminal or electronic device that is installed in the service provider's premises to facilitate non-cash on-sight payment of the client against the bank card or mobile account. In this case, the bank purchases the device, and required software, makes an interface with the bank network, and provides it to the client. The bank should hold deposits for a bit longer to serve for credit compared to other e-banking systems. The POS terminal manages the selling process by a salesperson-accessible interface. The same system allows the creation and printing of the receipt. It is one of the e-banking delivery channels in that customers use it to withdraw money from the bank when an ATM runs out of cash or becomes unfunctional. It can also be used to purchase services and products from hospitals, supermarkets, and hotels. (Shittu O,2010).

Agent Banking: In most developing countries, the banking sector has adopted several delivery channels to ensure that financial services reach both the banked and the unbanked, especially in remote areas. Agency banking will enhance access to financial services, and reduce expenses in rural regions, (David, 2012).

Branch Network: Organizations are expanded or get accessed by society at large using a branch network. The branch network establishes a physical presence to the public. Branches are linking their principal or other branches to perform common goal using different interfaces. One is manual and the other is technologically networked. A networked organization is expected to pass decisions faster than a non-networking organization. Moreover, networking organization reduce their labor cost and increase information quality compared to non-networked ones in addition to creating satisfaction for the customer. The main challenges for networking companies are reconciliation and follow-up issues. The number of branches a company has also impacted the company's performance.

Card banking: A bank card is typically a plastic card issued by a bank to its clients that performs one or more of several services that relate to giving the client access to a bank account. Physically, a bank card will usually have the client's name, the issuer's name, and a unique card number printed on it. Banking Cards (Debit / Credit / Cash / Travel / Others), banking cards offer consumers more security, convenience, and control than any other payment method. These cards provide 2 factor

authentications for secure payments e.g. secure PIN and OTP. RuPay, Visa, and MasterCard are some.

Electronic funds transfer system: Electronic funds transfer or EFT is the electronic exchange or transfer of money from one account to another, either within a single financial institution or across multiple institutions, through computer-based systems (Bahia, 2007). Bahia (2007) further stated that EFT is also a way of transferring money from one bank account directly to another without any paper money changing hands. One of the most widely-used EFT programs in our country is direct deposit, in which payroll is deposited straight into an employee's bank account, although EFT refers to any transfer of funds initiated through an electronic terminal, including credit card, ATM, and POS transactions.

2.3.2 The Role of Electronic Banking Services on Deposit Mobilization

Liébana-Cabanillas et al. (2013) recognized electronic banking portals as initial alternative channels to traditional bank branches. They mentioned many advantages of electronic banking; these include convenient and global access, availability, time- and cost-saving, wider choices of services, information transparency, customization, and financial innovation. Electronic banking (e-bank) services have a significant impact on deposit mobilization in the banking industry. Several variables associated with e-bank services can influence deposit behavior. The implementation of e-banking can bring about many competitive advantages for banks in today's highly competitive banking market. E-banking transactions are much cheaper than branches. Some of the major advantages of e-banking are:

Cost reduction: the major economic rationale of e-banking so far has been a reduction of overhead costs in providing banking services, that are expensive while using another channel. It also looks like the cost per transaction of e-banking often falls more rapidly than that of traditional banks once a critical mass of customers is achieved (Shah & Clarke, 2009).

Choice and convenience for customers: these are the most important benefits that outweigh any shortcomings of e-banking. Conducting transactions right from the comfort of home at the click of a button without taking too much time none would like to skip. Having a path of accounts through the internet is much faster and more convenient as compared to going to the bank for the

same. Even non-transaction facilities like ordering check books online, updating accounts, and enquiring about interest rates of various financial products become much simpler on the Internet (Sannes, 2001).

Load reduction on other channels: E-banking products are largely automatic and most of the customary activities such as account checking are carried out using these channels. This typically results in load reduction on other delivery channels, such as branches. This tendency is likely to continue as more high-level services such as mortgages or asset finance are offered using e-banking channels. In some countries, routine branch transactions such as cash deposit-related activities are also being automated, further reducing the workload of branch staff, and enabling the time to be used for providing better quality customer services (Shah & Clarke, 2009).

Easier expansion: Conventionally, when a bank wanted to expand geographically it had to expand new branches, thus incurring high startup and maintenance costs. E-channels, such as ATM, POS, mobile banking, and others have made this comparatively unnecessary in many circumstances. Now banks with a traditional customer base in one part of the country or world can attract customers from other parts, as most of the financial transactions do not require a physical presence near a customer's workplace (Shah & Clarke, 2009).

The amount of data collected by banks has grown rapidly in recent years. Existing statistical data analysis techniques find it difficult to manage the large volumes of data now available. This explosive growth has led to the need for new data analysis techniques and tools to find the information hidden in this data. Banking is an area where vast amounts of data are collected. This data can be generated from bank account transactions, loan applications, loan repayments, credit card repayments, etc. It is assumed that valuable information on the financial profile of customers is hidden within these massive operational databases and this information can be used to improve the performance of the bank (Steinbach, 2006).

Perceived ease of use: perceived ease of use is defined as the level of a person believes that using a mobile banking system would be free effort (Omwansa et al. 2012). By using this particular system it does not need a lot of effort as it is very easy and convenient. Since mobile banking services is easy to use it will increase the customer satisfaction while do the bank transaction instead of waiting in line at the bank counters (Abdoul Reza et al., 2011). Previous study done by Rahmath

et al. (2011) found that perceived ease of use has positive impact on the intention to adopt mobile banking. Sometimes customers will judge the best services that will give advantages and convenient while using it. Mobile phone can be used and bring at anywhere so the user can access the system anytime to do the bank transaction. Therefore, most the customer that use this service can adopt to their daily life especially the customer from government or private sector because they do not need go to the bank and also can save their time.

Security and Privacy: security is defined as a potential loss due to fraud or a hacker compromising the security of a mobile user. Fear of the lack of security is one of the factors that have been identified in most studies as affecting the growth and development of e-banking services. Therefore, it is very important to ensure the mobile banking system is secure while the user doing the financial transaction. Besides that, it will increase customer satisfaction with mobile banking and encourage the user to adopt this service. Privacy, on the other hand, refers to the protection of various types of data that are collected with or without the knowledge of the users during user's interactions with the Mobile banking system. Tashmia & Khumbula (2011) used the construct of perceived credibility which means the extent to which a person believes that using mobile banking will pose no security or privacy threats. In the same studies, a lack of credibility will be considered to be similar to security and privacy risks. Meanwhile, Luarn and Lin (2005) and Amin et al. (2008) supported security and privacy as two important dimensions under the construct of perceived credibility.

2.4 Overview of Machine Learning

A time deposit or term deposit is a deposit with a specified period of maturity and earns interest (Kagan, 2020). In other words, it is a money deposit at banking institution that cannot be withdrawn for a specific term or period unless a penalty is paid (Kagan, 2020). Although banks have various outreach plans to raise the capitals, one key plan is through engaging in direct marketing campaigns to sell their deposit products to clients. Therefore, identifying potential clients is of key importance for any banking institution. With the development of artificial intelligence, identifying potential clients becomes more feasible and manageable. Machine learning, as a powerful form of artificial intelligence, is one of the most exciting technological developments in history. Machine learning based predictive techniques can help banks determine which customers are more likely to subscribe a term deposit. Therefore, it is worthwhile for

researchers to analyze historic datasets by making a prediction model to forecast whether clients will subscribe a term deposit (Elsalamony, 2014).

Nowadays because of the availability and affordability of computer technology, organizations and businesses became able to record their day-to-day activities. With the advancement of this technology and wide application of database, corporations and organizations gather a huge amount of data record, which is stored in different form, but identifying valuable information or knowledge for taking a concrete decision becomes very difficult. According to (Kumar & Bhardwaj, 2011), the fact shows that data is growing at a very rapid rate, but most of data is simply being accumulated in storage device for no further use.

If the collected data from different sources processed properly, can provide hidden knowledge, which can be used for further development and that help to business decisions which bring business to the next level. But as mentioned (ZenTut., 2013), the unnecessary accumulated data itself is critical to a company‘s growth.

Because of such reasons, a new concept of Machine Learning need is important for business organizations for making better decision. Machine Learning refers to extracting knowledge from large set of data whatever may be the nature of data (ZenTut., 2013). The data may be web data, multimedia, text data etc. It is also the set of pattern used to find new or unexpected pattern in data using information contained in data warehouse. The discovered knowledge can be used by the bank managers to acquiring new customers, increasing revenue from existing customers, and retain good customers which is one of the important applications of Machine Learning techniques that can be used in banking organization.

Machine learning is an interdisciplinary field that requires knowledge from artificial intelligence and mathematical statistics to find and extract model from datasets that is beyond the capabilities of the SQL language (Structured Query Language) (Stamp, 2018) Moreover, Machine Learning requires following these steps:

Preprocessing->model->validation

- **Pre-processing:** Pre-processing is one of the important stages in Machine learning because real datasets are noisy, dirty, incomplete, and in different formats.

- **The model:** The model means that the techniques and the algorithms that Machine Learning uses are applied to the data to get results. There are wide range of Algorithms used in Machine Learning described in detail by (Chen, Hu, & Hsieh, 2014)some of the more common, ones being K-means clustering and A-priori algorithm.
- **Validation:** This is the final stage of Machine Learnings, which verifies the output Model from the Machine Learning algorithms. The entire output model that are found by Machine Learning algorithms are not necessarily valid. To overcome this challenge, the Machine Learning algorithms need to be tested on a test set of data. If the output Model meet the desired output of the Machine Learning algorithm, it will be applied to a larger dataset to discover knowledge. However, if the output model do not fit the desired results, the pre-processing and the Machine Learning algorithm steps need to be re-evaluated (Stamp, 2018).

2.5 Machine Learning Techniques

Computers can use machine learning, a branch of computer science, to learn from their prior experiences and provide results based on those experiences (Ayodele, 2010). An application of AI known as machine learning gives systems the capacity to learn and advance automatically from experience without being explicitly programmed (Ayodele, 2010). These characteristics are examined by utilizing a machine learning approach and methods as powerful tools for data mining and permit to take new insights into bank customer behavior.

Machine learning can be classified as supervised learning, unsupervised learning, reinforcement, and semi-supervised learning (Ayodele, 2010). Supervised learning involves training a model on labelled dataset, where the input data is paired with corresponding output labels. The goal of supervised learning is to learn a mapping between the input and output variables so that the model can make accurate predictions on new, unseen data. Unsupervised learning, on the other hand, involves training a model on unlabelled dataset, where the input data is not paired with any output labels. The goal of unsupervised learning is to discover patterns or structures in the data, such as clusters or latent variables. Reinforcement learning involves training a model to make decisions in an environment, based on feedback in the form of rewards or penalties. The goal of reinforcement learning is to learn a policy that maximizes the cumulative reward over time. In the context of

bank deposit mobilization, machine learning would allow to reveal the complex patterns driving the deposit process as these algorithms are able to identify complex and nonobvious patterns or knowledge hidden in a database with millions of data points (Ainan & Nur-E-Arefin, 2022). Number of methods used in demand studies to conclude that machine-learning techniques are both adequate and effective for this type of analyses as they reveal complex patterns. The advantages of following a machine learning approach in complex contexts, such as consumer behaviour, would explain why this machine learning is gaining ground.

Many Machine Learning algorithms techniques can be used to classify a problem given a set of features in both supervised and unsupervised learning. Instead of ex-ante selecting a default machine learning technique, we initially consider a number of machine learning techniques that have proved their value in this field (Fischer & Krauss, 2018). The following machine learning approach is employed to examine bank deposit mobilization. According to (Anastasiou, 2020) Bank policy makers, regulators, and managers must predict deposit levels. Different ML techniques have been employed to build bank deposit classification models. SVM separates the class samples by an optimal hyperplane and thereby significantly increases the performance of the models (Bauer & Kahavi). These work looks to supervised algorithms to see if any are particularly useful in analyzing the behaviors of CBO customer and give classification of customer deposit mobilization using e-banking service. Additionally, for those machine learning techniques that require selecting some hyperpara-meters (e.g. number of features for each tree in the random forest algorithm or C and gamma values in the SVM), they are not arbitrarily chosen but tuned to obtain the optimal parameter values for higher accuracy. The performance of all the machine learning methods and the logit models is computed after having optimized the hyper-parameters for each and every method. Machine Learning Algorithms investigated in this research are Random Forest, Decision Tree, K-Nearest Neighbor (K-NN), Support Vector Machine (SVM), Logistic Regression and Naive Bayes.

2.5.1 Decision Tree

Two common uses for decision tree approaches are creating classification systems based on several variables and creating prediction algorithms for target variables (Ayodele, 2010). In a decision tree, each inner node represents an attribute, while each leaf node represents an outcome. Top-

level nodes have the ability to split based on the value of the attribute. The method used to split the tree is called recursive splitting. This algorithm supports decision-making. The decision tree field values are used to separate subsets of records in the database. Appling this strategy Recursively to each subset will result in all instances of each node being assigned to a single class. Decision trees are easy to understand and explain and can handle both categorical and numerical data. Overfitting, however, overfitting can occur if the tree is too complex. The decision tree method can be used for both classification and regression applications. The data is recursively split based on feature values until the resulting subsets are as pure as is practical.

The decision tree algorithm is a popular and simple supervised machine learning method for classification and regression applications (Bali, Sarkar, & Lantz, 2017). This is a tree-based model that, to generate predictions, develops a hierarchy of if-else conditions based on the characteristics of the input data. Recursively dividing the input data into subsets based on feature values is how the decision tree algorithm work. The best data separation feature based on specific criteria, such as information gain or Gini impurity, is selected at each step. Until a stopping requirement is met, such as reaching a maximum depth or a minimum number of samples in a leaf node, the process continues. Divide and conquer tactics are used by decision tree learning to find the best split points within a tree by undertaking a greedy search. Recursive partitioning is another name for this method. Recursively and greedily growing the decision tree, the procedure begins by generating a single node (the root). All potential conditions are assessed and graded at each node. After that, the dividing procedure is repeated in a top-down, recursive fashion until most or all of the records have been assigned to particular class labels.

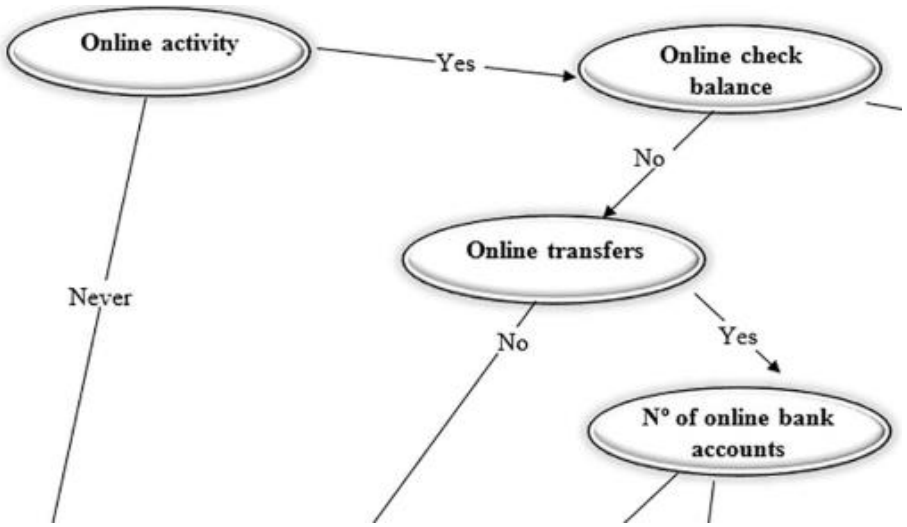

Figure 2.1 Classification using Decision Tree

2.5.2 Support Vector Machine (SVM)

Support Vector Machines are supervised learning models that can be used for classification, prediction and clustering problems. An SVM takes a set of input observations, associated binary outputs, and constructs a model that can classify new observations into one class or the other. Support vector machine is highly preferred by many as it produces significant accuracy with less computation power (Khalid, 2015). A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperactive plane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyper plane which categorizes new examples. In two-dimensional space this hyperactive plane is a line dividing a plane in two parts where in each class lay in either side (Jorma, 1996).

The key element is that support vector machine algorithm is to find a hyperplane in an N-dimensional space (N-the number of features) that distinctly classifies the data points.

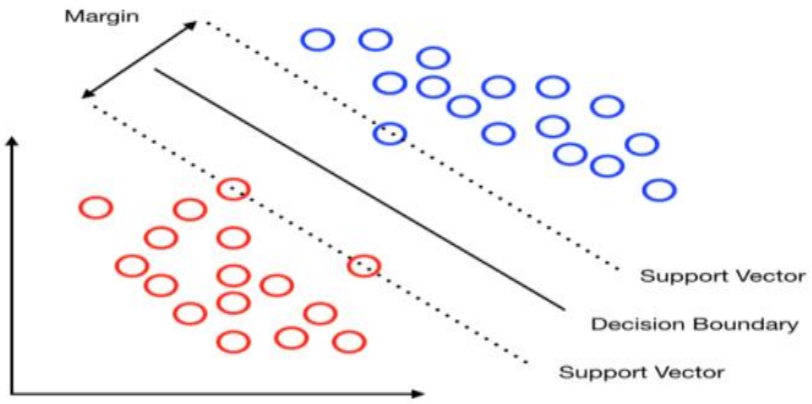

Figure 2.2 Linear Line Separating the Data Types

To separate the two classes of data points, there are many possible hyper planes that could be chosen. The main thing in SVM is to find a plane that has the maximum margin, which means the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

In real world application, there is trade off on finding perfect class when the two classes are not linearly separable like due to noise, the condition for the optimal hyperplane can be relaxed by including an extra term: millions of training dataset.

In the case of real-world application, it is not usually possible to get a line that perfectly separates the data within the space. Hence, we might have to use a curved decision boundary. It is possible to get a hyper-plane, which could separate the data, but this may not be desirable if the data has noise in it. In such cases, we need to use the soft margin method (Jorma, 1996). The soft margin method allows points to appear on the incorrect side of the margin. These points have a penalty associated with them. The penalty increases as the points are farther from the margin. The hyperplane separation looks to minimize the penalty of incorrectly labeled points, while maximizing the distance between the remaining examples and the margin.

The other approach is SVM kernel it employed to separate data that is not linearly separable, is to map the data into a higher dimensional space. By mapping x= (x, x2) the data will be mapped into two-dimensional space. When this two-dimensional mapping is graphed, an obvious linearly separable line appears (Jorma, 1996). The mapping used to increase the dimensionality of the problem is dependent on the data space being investigated. The above computations, which are

used to find the maximum-margin separator, can be expressed in terms of scalar products between pairs of data points in the high-dimensional feature space. These scalar products are the only part of the computation that depends on the dimensionality of the high-dimensional space.

2.5.3 Naive Bayes

Naive Bayes is a widely used classification method based on Bayes theory. Based on class conditional density estimation and class prior probability, the posterior class probability of a test data point can be derived and the test data will be assigned to the class with the maximum posterior class probability (Jiangtao Ren, 2009). Naive Bayes algorithm is one of the most effective methods and is a simple but surprisingly powerful algorithm for predictive modeling (Yuguang Huang, 2011).

The main reason behind its popularity is that it can be written into the code very easily delivering predictions model in very less time. Thus, it can be used in the real-time model predictions. In Naive Bayes probability theory, Bayes theorem relates the conditional and marginal probabilities of two random events (Jiangtao Ren, 2009). It is often used to compute posterior probabilities given observations.

Let x = (x 1, x2. . . xd) be a d-dimensional instance which has no class label, and our goal could be to build a classifier to predict its unknown class label based on Bayes theorem. Let C = {C1, C2… CK} be the set of the class labels. P (Ck) is the prior probability of Ck (k = 1, 2... K) That are inferred before new evidence, P (x|Ck) be the conditional probability of seeing the evidence x if the hypothesis Ck is true. A technique for constructing such classifiers to employ Bayes' theorem to obtain is given by the following formula:-

$$P(C_k|x) = \frac{P(x|C_k)P(C_k)}{\sum_{k'} P(x|C_{k'})P(C_{k'}}$$ Equation 1

2.5.4 K-Nearest Neighbor (KNN)

The nearest neighbor (NN) classifiers, especially the k-NN algorithm, are among the simplest and yet most efficient classification rules and are widely used in practice (Jorma, 1996). The purpose of KNN algorithm is to use a database in which the data points are separated into several separable classes to predict the classification of a new sample point. An object is classified by looking to its nearest examples (Saxena, 2016). In KNN algorithms, K states how many neighbors will be used in voting, it is important to decide the value of k because the accuracy of the classification is dependent on it. K=1 simply states that an object will be assigned the same class as its nearest example. As the number of K increases then we need to classify a given instant based on the resemblance of all the stated K instances. The measurement can be performed using any distance metric or similarity function, such as the Euclidean, Cosine or (Saxena, 2016). The classifier is developed from the training data documents in this case and to check for the efficiency and accuracy of the classifier the holdout method has been used. In the holdout method, the original training dataset is partitioned into two, a major portion to generate the classifier and the minor portion to check the precision of the classifier. It is an assumption that documents of the same class will always be the nearest neighbors of each other i.e. the distance between them will be less as they are in some way related to each other (Saxena, 2016).

The major problem of using the k-NN decision rule is the computational complexity caused by the large number of distance computations and the other most critical problem is selecting the value of K in K-nearest neighbor. A small value of K means that noise will have a higher influence on the result i.e., the probability of over fitting is very high. A large value of K makes it computationally expensive and defeats the basic idea behind KNN (that points that are near might have similar classes). A simple approach to select k is k = n^ (1/2) (Saxena, 2016).

In the general process of KNN algorithms, the first step is preprocessing the data in the database in such a way as to ensure that we can compare observations. Then, our observations become points in space and we can interpret the distance between them as their similarity (using some appropriate metric) One of the most widely used metrics is the Euclidean distance. The Euclidian distance between two instances (X1, X2, X3 … Xn) and (U1, U2, U3 … Un) is given by the following formula:-

$$\sqrt{(X_1 - U_1) + (X_2 - U_2) + \cdots + (X_n - U_n)}$$ Equation 2

2.5.5 Logistic regression

Logistic Regression is also called logistic model or logistic regression. It is a predictive analysis. It takes independent features and returns output as categorical output. The probability of occurrence of a categorical output can also be found by Logistic Regression model by fitting the features in the logistic curve (Vaidya, 2017).

Logistic Regression, falls under Supervised Machine Learning. It solves mainly the problems of Classification to make predictions or take decisions based on past data (Vaidya, 2017). It is used to predict binary outcomes for a given set of independent variables. The dependent variable's outcome is discrete.

The output of Logistic Regression is a sigmoid curve or more popularly known as S-curve. Where the value on the x-axis, independent variable would determine the dependent variable on the y-axis. In Logistic Regression, there are only two possible outcomes. 0 and 1. That something occurs, or it does not. We use a threshold value to make our prediction easier. If the x-axis' corresponding y-value probability is lesser than the threshold value, the outcome is taken as 0. If it is greater than the value, the outcome is taken as 1.

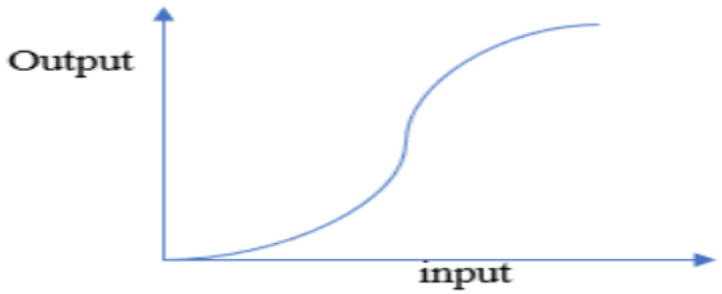

Figure 2.3 General Logistic Regression Curve

The simpler Linear Regression model can replace the Logistic Regression model when the output variable is taken to be continuous. When the output variable is not continuous or is dichotomous, another model has to be applied in order to consider this difference.

2.5.6 Ensemble Techniques

Rather than using only one model, ensemble techniques aim to increase the accuracy of outcomes in models (Ainan & Nur-E-Arefin, 2022). The integrated models have greatly improved the results' accuracy. The popularity of ensemble techniques in machine learning has increased as a result. Sequential aggregation techniques and parallel aggregation techniques are the two primary categories of aggregation procedures. The base of learners is created using sequential ensemble approaches, such as adaptive boosting (AdaBoost). The foundation of subsequent learning generations encourages interdependence between individuals. The performance of the model is then enhanced by assessing learners who were previously underrepresented.

There are various types of ensemble approaches, including stacking, bagging, and boosting [20] (Bali, Sarkar, & Lantz, 2017). Boosting is an ensemble technique that uses past prediction mistakes to improve future forecasts. By integrating several weak base learners into one strong learner, this technique dramatically improves the model's predictive capacity. In order to reduce training errors, the ensemble learning technique known as "boosting" turns a group of weak learners into a strong learner. A random sample of data is chosen, fitted with a model, and then trained successively in boosting, where each model tries to make up for the shortcomings of the one that came before it. Increasing efforts to create a powerful classifier by increasing the number of ineffective classifiers first, a model is created using the training set of data. The second model is then constructed in an effort to fix the flaws in the first model.

A random forest is an ensemble learning method where multiple decision trees are constructed and then they are merged to get a more accurate prediction. If there is one method in machine learning that has grown in popularity over the last few years, then it is the idea of random forests. The concept has been around for longer than that, with several different people inventing variations.

The idea is largely that if one tree is good, then many trees (a forest) should be better, provided that there is enough variety between them. The most interesting thing about a random forest is the ways that it creates randomness from a standard dataset. The first of the methods that it uses is the one that we have just seen: bagging. If we wish to create a forest then we can make the trees different by training them on slightly different data, so we take bootstrap samples from the dataset for each tree. However this isn’t enough randomness yet. The other obvious place where it is

possible to add randomness is to limit the choices that the decision tree can make. At each node, a random subset of the features is given to the tree, and it can only pick from that subset rather than from the whole set.

As well as increasing the randomness in the training of each tree, it also speeds up the training, since there are fewer features to search over at each stage. Of course, it does introduce a new parameter (how many features to consider), but the random forest does not seem to be very sensitive to this parameter; in practice, a subset size that is the square root of the number of features seems to be common. The effect of these two forms of randomness is to reduce the variance without effecting the bias. Another benefit of this is that there is no need to prune the trees. There is another parameter that we don't know how to choose yet, which is the number of trees to put into the forest. However, this is fairly easy to pick if we want optimal results: we can keep on building trees until the error stops decreasing. Once the set of trees are trained, the output of the forest is the majority vote for classification, as with the other committee methods that we have seen, or the mean response for regression. And those are pretty much the main features needed for creating a random forest. The algorithm is given next before we see some results of using the random forest.

The ensemble learning technique known as bagging, often referred to as bootstrap aggregation, is frequently used to lessen variance within a noisy dataset (Bali & Sarkar, 2016). In bagging, a training set's data is randomly sampled and replaced, allowing for multiple selections of the same data points. These weak models are then trained independently using many data samples, and depending on the task for instance, classification or regression the average or majority of those predictions results in a more accurate estimate. Using stacking, we can train numerous models to tackle related issues and then create a new, more effective model based on the combined results. One of the widely used methods for ensemble modeling in machine learning is stacking. In order to combine different weak learners with Meta learners and make better predictions for the future, they are ensemble in simultaneously. Layered generalization is an alternate name for the ensemble approach stacking (Bali, Sarkar, & Lantz, 2017).This method works by allowing a training algorithm to combine the predictions of several related learning algorithms.

The best methods for minimizing the variance of models and boosting prediction accuracy are ensemble methods. When multiple models are integrated to create a single forecast that is selected

from among all other potential predictions from the combined models, the variance is eliminated. An ensemble of models integrates multiple models to guarantee that the final prediction is the best feasible one based on taking all predictions into account

2.6 Review of Related Works

In this section, research works related to the application of machine learning in areas of e-banking service and deposit mobilization will be discussed.

Previous works of (Palaniappan, Mustapha, Foozy, & Atan, 2017) on the bank, telemarketing data sets have been solely based on making class predictions using various classifying algorithms. Telemarketing is a type of direct marketing where a salesperson contacts the customers to sell products or services over the phone. The database of prospective customers comes from direct marketing database. The company needs to predict the set of customers with the highest probability of accepting the sales or offer based on their characteristics or behavior during shopping. Recently, companies have started to resort to machine-learning approaches for customer profiling. The study focused on helping banks to increase the accuracy of their customer profiling through classification as well as identifying a group of customers who have a high probability of subscribing to a long-term deposit. In the experiments, three classification algorithms are used, which are Naive Bayes, Random Forest, and Decision Tree. The experiments measured accuracy percentage, precision, and recall rates and showed that classification is useful for predicting customer profiles and increasing telemarketing sales.

The study conducted by (Muslim, Darsil, Alamsyah, & Mustaqim, 2020) aimed to evaluate the performance of the multiple linear regression (MLR) using a fixed-effects model (FE) and artificial neural network (ANN) models to predict the level of customer deposits in a sample of Tunisian commercial banks. The study as research methodology and design Training and testing datasets are developed to evaluate the level of customer deposits of 15 Tunisian commercial banks over the 2002–2021 period. This study used two predictive modelling techniques: the MLR using a FE model and ANN. In addition, it uses the mean absolute error (MAE), R-squared, and mean square error (MSE) as performance metrics. The main finding of the study proved that both methods have a high ability in predicting customer deposits of 15 Tunisian banks. However, the ANN method has

a slightly higher performance compared to the MLR method by considering the MAE, R-squared, and MSE.

According to the study conducted by (Mohammed et al,2021) the performance of ensemble learning algorithms which is a novel approach to predict whether a new customer will have a term deposit or not was evaluated. The study used Portuguese retail bank data containing 45,211 phone contacts with 16 input attributes and one decision attribute. The data are preprocessed by using the Discretization technique. 40,690 samples were used for training the classifiers, and 4,521 samples were used for testing. In this work, the performance of the three mostly used classification algorithms named Support Vector Machine (SVM), Neural Network (NN), and Naive Bayes (NB) were analyzed. Then the ability of ensemble methods to improve the efficiency of basic classification algorithms is investigated and experimentally demonstrated. Experimental results exhibit that the performance metrics of Neural Networks (Bagging) is higher than other ensemble methods. Its accuracy, sensitivity, and specificity are 96.62%, 97.14%, and 99.08%, respectively. Although all input attributes are considered in the classification method, in the end, a descriptive analysis has shown that some input attributes have more importance for this classification. Overall, the study indicated that ensemble methods outperformed the traditional algorithms in this domain.

Wisaeng et al.(2014) analyzed the value of the lifetime of customers and neural networks to improve the prediction of bank deposit subscriptions in telemarketing campaigns. Moro et al. [8] assessed and compared the classification performance of four different classification algorithms, i.e., Decision Tree (DT), Neural Network (NN), Logistic Regression (LR), and Support Vector Machine (SVM) on the bank direct marketing dataset to classify the bank deposit prediction. The area of the cumulative lift curve (ALIFT) and Receiver Operating Characteristic Curve (ROC) was used to compare these models. NN provides the best prediction performance, which produced the resulting ROC of 0.80 and ALIFT of 0.67.

(Carbo-Valverde S, et al, 2020) in their study a machine learning approach to examine the digitalization process of bank customers using a comprehensive consumer finance survey. By employing random forests, conditional inference trees, and causal forests algorithms, the study identifies the features predicting bank customers' digitalization process, illustrates the sequence of consumers' decision-making actions, and explores the existence of causal relationships in the digitalization process. The study results showed that the Random forests algorithm provides the

highest performance–they accurately predict 88.41% of bank customers' online banking adoption and usage decisions. The study finds that the adoption of digital banking services begins with information-based services (e.g., checking account balance), conditional on the awareness of the range of online services by customers, and then is followed by transactional services (e.g., online/mobile money transfer). The diversification of the use of online channels is explained by the consciousness about the range of services available and the safety perception. A certain degree of complementarity between bank and non-bank digital channels is also found. The treatment effect estimations of the causal forest algorithms confirm causality of the identified explanatory factors. These results suggest that banks should address the digital transformation of their customers by segmenting them according to their revealed preferences and offering them personalized digital services.

(Birhane G. 2022) in his study used a machine learning algorithms for customer churn prediction., To build the prediction model he used SVM, KNN, Naïve Bayes and Logistic Regression algorithms. To Confusion matrix was used to calculate the accuracy, recall and precision and evaluate the performance of the models. Based on the research findings, the KNN classifier produced an accuracy of 99.91%, the SVM classifier produced an accuracy of 92.4%, Logistic Regression model also produced an accuracy of 93.8%, and Naïve Bayes classifier produced an accuracy of 83.8 %. Therefore, the KNN classifier is proposed for constructing a bank customer churn prediction model.

Table 2.1 Summary of Literature Reviewed on ML application for E-service deposit Mobilization

No	Researcher/ Author & Year	Objective	Classification Technique & Model	Gap
1.	Wisaeng et al.(2014	analyzed the value of the lifetime of customers and neural networks to improve the prediction of bank deposit subscriptions in telemarketing campaigns	Four ML algorithms were used: Decision Tree (DT), Neural Network (NN), Logistic Regression (LR), and Support Vector Machine (SVM	. does not use ensemble model for boosting accuracy of the models.
2.	Carbo-Valverde S, Cuadros-Solas P, Rodríguez-Fernández F (2020)	This thesis employs a machine learning approach to examine the digitalization process of bank customers using a comprehensive consumer finance survey.	Random forest algorithm to predict the probability of a customer being a digital customer.	Favors for random forest algorithm by saying that "random forest outperforms other machine learning algorithms in predicting the probability of a customer being a digital customer".
3.	(Palaniappan, Mustapha, Foozy, & Atan, 2017)	This thesis focus on bank telemarketing data sets have been solely based on making class predictions using various classifying algorithms attempted to determining the likelihood of a consumer signing up for a term deposit.	Decision Tree Algorithm and Rough Set Theory	This thesis only uses two learning models this lacks comparison issues because one machine learning may outperform for some particular problem and given data set.

4.	Berhane Gebreegziabher Seyoum (2022)	This study an attempt is made to apply machine-learning algorithms for customer churn prediction.	Predictive model, machine learning algorithms such as SVM, KNN, Naïve Bayes and Logistic Regression	Its gap is only for prediction of customer churn, and does not use ensemble model for boosting accuracy of the models.
5.	Giragn Garo (2015)	Explores the theoretical as well as empirical analysis of those factors having an impact on deposit volume in banks and even assesses which ones are more significant or less significant.	The data is analysed through the econometric analysis using SPSS software.	Totally, this thesis does not use machine learning techniques and the finding shows e-banking service is not mentioned as deposit mobilization factor.
6.	Addisalem Tadesse Bogale, Tagesse Abebe Sugebo and Lemabo Jaboro Satore (2020)	An important indicator of the success and efficiency of any credit agency, which is also a banking institution is, the extent to which it is able to mobilize the savings of the community in the form of deposit.	Time serious secondary data to be collected from Commercial Bank of Ethiopia database.	They did not use machine-learning technique and does not focus on e-banking service. Only discuss macroeconomic factors that affect deposit mobilization.
7.	(Patwary, Akter, Bin Alam, & Rezaul Karim, 2021)	Based on classification models while we deal in this study with regressions problems deposit mobilization.	Used three classification techniques, which are the Naïve Bayes, ANN and support vector machine.	Does not focus on e-banking service

2.6.1 Gap Analysis

This study attempts to review different literature related to effects of e-banking service for mobilizing deposit. In addition, existing studies used different approach such as different attributes in terms of e-banking services and customers characteristics, different datasets and preprocessing activities, and different method for their studies, because of most commercial banks follow different way of handling customers and services provided, strategies and banks structure's and strategies throughout the world. In Related works above, suggest that the results will improve and more powerful model could be developed by including different attributes related to customer characteristics, e-banking services types and different datasets. In addition to researcher (Gafrej, 2023) The study suggests that improving results and developing more powerful models can be achieved by including customer characteristics attributes and datasets, clustering data based on demographics, and categorizing premium and ordinary customers.

Banks struggle with deposit mobilization due to lack of knowledge about deposit-affecting factors. By utilizing machine-learning approaches, the thesis creates a cutting-edge method for classifying effects of e-banking services for deposit mobilization. Most of the currently published articles do not specify the hyper-parameters that mean researchers randomly train a model. The hyper-parameters are, however, chosen and optimized by the thesis. Depending on how much the cost function and training procedure reduce the test error; grid search is used to evaluate the model using a validation set. Furthermore, an in-depth research is required. This study aims to fill this gap by using bank-specific variables. In addition, aims to identify these gaps for successful organization operation and understanding changing business environments.

With little data and few features, the existing research uses several machine learning techniques to classify factors that affect deposit, which results in a model with poor performance (Ainan & Nur-E-Arefin, 2022). A more comprehensive dataset with large number of parameters are used for this thesis. Most the earlier studies uses single machine learning algorithms but this thesis utilizes an ensemble machine learning methodology. Rather than using single model ensemble techniques can significantly increase the performance of classification by integrating the results from various models to generate better results, Support Vector Machines (SVM), Linear Regression (LR), Bayes, KNN and Decision Trees (DT) are the ensemble techniques used in this study.

CHAPTER THREE

3 METHODOLOGY AND EXPERIMENT

3.1 Overview

Data preparation is some of the primary tasks that highly determine the Machine Learning results. The model built mainly depends on how thoroughly and carefully the necessary data is obtained, analyzed and preprocessed. After description of the original data the next subsequent sections present the Business understanding has selected relevant attribute, Data understanding construct the task for relevant attribute used in deposit mobilization, and preprocessing tasks done to clean attribute in the dataset were performed. This chapter discusses a research methodology to prepare a dataset and techniques to achieve the objectives and answer the research question. This thesis develops classification model for e-banking service for deposit mobilization using a machine learning approach. However, all of the resources, procedures, and goals of this thesis, as well as the model used, must be described and explained.

3.2 Methodology

Research methodology is the general principle that guides the research. In order to conduct good research, a well-defined approach and principle has to be followed. This research is designed to apply machine learning technology for classifying effect of e-banking service on deposit mobilization in Cooperative Bank of Oromia S.C. Methodology is the process used to collect information and data for the purpose of making business decisions. The methodology may include publication research, interviews, surveys and other research techniques, and could include both present and historical information. The data source for experimentation is acquired from CBO database department.

3.2.1 Study Subject

The initial step involves task such as the problem definition and project goal determination, identification of key people and grasping the current solution to the problem through close

consultation with the domain experts. Then the project goals is transformed to machine learning preliminary selection of python tools is used in the study conducted.

In this step of the machine Learning model, the main task of the research is accomplished by utilizing different mechanisms. Review of documents including different manuals; policy documents, procedure documents, Different National Bank deposit directives , different deposit policy has an impact in understanding the problem domain and different reports of the Bank . In this case, the researcher understand in better to define the problem, determine the domain objectives, to assess the problem and to determine the machine learning goals, which are helpful for the next steps.

3.2.2 Study Design

This research follows experimental research. Experimental research is a study that strictly adheres to a scientific research design (Ghaneei & Mohammad, 2016). It includes a hypothesis, a variable that can be manipulated by the researcher, and variables that measured, calculated and compared. The researcher collects data and results, which are either support or reject the hypothesis. This method of research is referred to a hypothesis testing or a deductive research method (Babbie-Earl, 1998).

Most importantly, experimental research is completed in a controlled environment. To undertake the experiment in this research, this study followed machine-learning models. According to, (Kecerdasan, 2007), machine learning model is enhanced the knowledge discovery process by combining the academic and industrial models. Thus, these models are research-oriented, which present machine-learning step than the modeling step. Moreover, the knowledge that discovered in the final step for a deposit domain may be applied in bank.

3.2.3 Data Preparation

This phase is also known as data filtering and involves the process of organizing the data for ML. The following steps are completed in order to filter the data to be used in the ML process. These are Business Understanding (Customer Definition, Classification, Account Handling, and

Attribute Selection), Data Understanding (Data Representation), Data Collection (Data Quality Assurance), and Data Preprocessing (Data Cleaning and Data Set Splitting).

3.2.4 Data Management and Analysis

In this step, the data used is prepared to apply the machine learning algorithms. It consist of tasks such as sampling, testing the correlation and significance of the data, cleaning the data, checking the completeness of the tuples, handling noisy and missing values. Then, feature selection and extraction algorithms reduce the dimensionality of the data. This step also comprises, the derivation new attributes, summarization of the data. Finally, the datasets that meet the input requirements of machine learning tool in python stated in the first step is selected for modeling purpose. It is also include all activities that are needed to construct the final data set.

This step involves Machine-learning methods that are applied on the pre-processed data to discover knowledge. This step is application of the selected machine learning methods to the prepared depositor's data and testing the generating rules whether they achieve the required minimum threshold. Beside it is a step of finding hidden, non - trivial and previously unknown information from the data. Python is be used for mining the data. Python is one of the open source machine learning technique having several functionalities.

3.2.5 E-banking Services Deposit Mobilization Model Workflow

This thesis uses data from cooperative bank of Oromia to e-banking services for classify deposit mobilization. Machine learning techniques used as base model such as Logistic Regression (LR), Decision Trees (cart), Support Vector Machines (SVM), K- Nearest Neighbors (KNN), and Naive Bayes. After experiment is done, using weak learners he accuracy measure is become less for this reason an Ensemble techniques were used in the thesis by Random Forest algorithm, the initial dataset has to be trained and handled differently. The approach was then refined using data, and model was created. After creating the model and classifying he deposit mobilizing e-banking services, we evaluated it using test set. The thesis architecture, as shown in figure 3.1, examines and constructs the research first, then gathers the dataset and pre-processes the data to enhance the quality of the data set and effectively gather the precise results we need. Using several feature extraction algorithms, pre-processing converts text input into actual vectors. Separate the dataset

into training and test sets as well. Next, create a machine learning algorithms and classify the deposit mobilizing factors. The architecture for classifying deposit mobilizing factors using machine learning algorithms is shown in the image below.

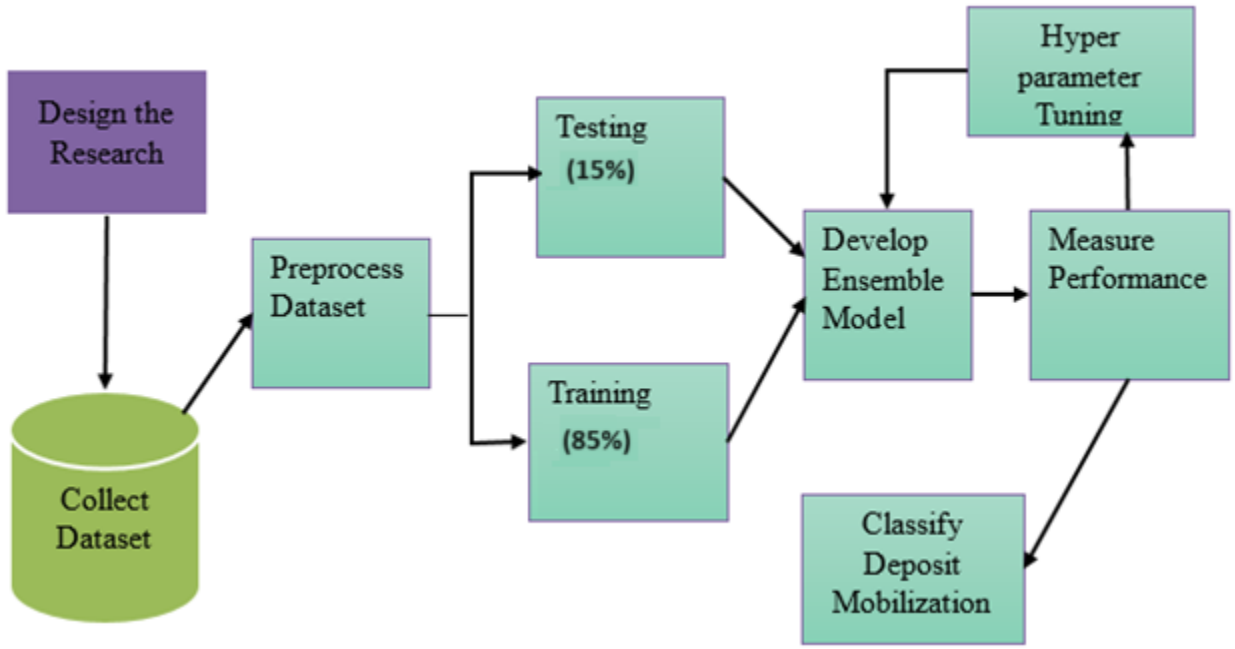

Figure 3.1 Classification model workflow

In this study, the researcher chose an experimental research strategy since it allows for the incorporation of several variations and facilitates comparison analysis. Our goal is to classify deposit mobilization by investigating a variety of e-banking service factors. It is essential to define a methodology through study design in order to construct a model. In order to do this, the issue at hand must be carefully examined, the investigation's scope must be determined, and the factors that affect the model's creation must be chosen. The study identifies the research area, focuses on the particular issue there, and chooses the best parameter for deposit mobilization. Furthermore, the classification of e-banking services for deposit mobilization in the context of cooperative bank of Oromia is thoroughly investigated and examined in this study.

3.2.6 Understanding of the Problem Domain

The initial step involves task such as the problem definition and project goal determination, identification of key people and grasping the current solution to the problem through close

consultation with the domain experts. Then the project goals are transformed to machine learning goal and the preliminary selection of machine learning tools to be used in the study is conducted.

In this step of machine learning, model the main task of the research, which is accomplished by using different mechanisms. Review of documents including different manuals; policy documents, procedure documents, Different National Bank directives, different deposit policy has an impact in understanding the problem domain and different reports; the reports which reviewed include annual financial report of the Bank. In this case, the researcher became in a good position in order to define the problem, determine the domain objectives, to assess the problem and to determine the data mining goals, which are helpful for the next steps.

3.2.7 Understanding of the Data

This step includes collecting the initial Transaction data from Cooperative Bank of Oromia Database Department. The data can be visualize using excel format. The visualization focuses on ensuring the data completeness since there are transactions records, which have incomplete data. Similarly, the visualization can be focused on cross checking the suitability of the data concerning the machine learning goals. Besides exploring the data and describe the data to identify the possible attributes is another task of understanding the data. Additionally, this step enables to determine the machine learning methods and algorithms. Moreover, from python matplotlib by importing pyplot is another data visualization tool, which uses in describing the attributes.

The data for this research has been compiled from cooperative Bank of Oromia head office. The original data of the customers' profile of the cooperative Bank of Oromia was taken and all the necessary operations of data selection and data preparation were carried out. Originally, information about the customer is recorded when the individual is registered and took the card as well as the transaction of the customer is also recorded whether he/she uses the ATM, mobile banking and other service. However, after discussions and survey of the data with the domain experts and database administrators, the data taken were constructive as well as sufficient and could be used for a realistic data analysis

3.2.8 Preparation of the Data

Once we have gathered the data for this research, our next step is to prepare data for model training and evaluation. A key focus of this stage is to recognize missing data and then appropriately handle them. This is important because they may substantially bias the results. Before jumping to the methods of addressing missing data, we first need to know the reasons why data will be missing. Normally, there are three typical mechanisms causing missing data: missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR). MCAR is quite straightforward and means what it says: the propensity for a data point to be missing is completely random (Little and Rubin 2002). In other words, whether a data point is missing has nothing to do with any missing or observed values in the dataset; the missing data are just a random subset of the data. MAR means that the propensity for a data point to be missing is not related to the missing data, but related to some of the observed data (Little & Rubin, 2002). This means the missing is not completely at random, but is conditional on another variable. MNAR means that propensity for a data point to be missing varies for reasons that are unknown to us (Bland, 2015). Two possible reasons for MNAR are that the missing value depends on the hypothetical value (e.g. individuals with high salaries usually do not want to reveal their incomes in surveys) or the missing value depends on the value of another variable (e.g., females generally don't want to reveal their ages and thus the missing value for age is related to the value of gender variable). After we obtain a better understanding of the missing data, we can then choose an appropriate approach to handle them. For the first two mechanisms, it is safe to remove the data with missing values depending on the number of their occurrences, whereas for the third mechanism, removing missing data can create a bias in the model (Little & Rubin, 2002).

In this step, the data used are prepared to apply the ML methods. It consists of tasks such as sampling, testing the correlation and significance of the data, cleaning the data, checking the completeness of the tuples, handling noisy and missing values. Then, feature selection and extraction algorithms reduce the dimensionality of the data. This step also comprises, the derivation new attributes, summarization of the data. Finally, the datasets that meet the input requirements of machine learning tools stated in the first step are selected for modeling purpose.

It is also include all activates that are needed to construct the final data set. The data, was taken from Cooperative Bank of Oromia Database Department, preprocessed and cleaned for the application of different data mining techniques.

The data contains information about e-banking services provided and customer details in .csv file format. It contains a total of 82593 observations and 13 variables. With the consultation of domain expert, all variables for the classification tasks were selected. These variables are total_deposit, other_credit, other_debt, mobile_debt, card_credit, card_debt, mobile_banking, gender, age, marital_status, marital status, job and ATM card.

3.2.8.1 Collect Initial Data

The initial data for this study were collected after getting permission from the administrations of cooperative Bank of Oromia. The data have been collected from the customers profile database. The table was saved to MS Excel file. The fields that were obtained from the collection are stated below in Table 3.1.

3.2.8.2 Description of the Collected Data

The decision on the data that was used for the analysis is based on several criteria, including its relevance to the machine learning goals. The attributes are selected based on understanding the problem domain with the help of domain expert and literature reviews. Table 3.1 presents the description of the collected attributes with their data type. The final selected data was prepared and preprocessed before developing the model.

Table 3.1 Description of the Selected Attributes from Cooperative Bank of Oromia

No.	Attribute Name	Data Type	Description
1.	Gender	Nominal	Sex of the customers
2.	Birth date	Numeric	The date birth of customers
3.	Mobile debit	Numeric	debit transaction using mobile
4.	Mobile credit	Numeric	Credit transaction using mobile
5.	Card debit	Numeric	Debit transaction using ATM card
6.	Card credit	Numeric	Credit transaction using ATM card
7.	Other debit	Numeric	Debit transaction

8.	Other credit	Numeric	Credit transaction
9.	ATM	Nominal	Does the customer have an ATM card?
10.	Mobile Banking	Nominal	Does the customer using mobile banking
11.	Marital status	Nominal	Marital status of the customer
12.	Job	Nominal	Job type of the customer
13.	Deposit balance	Numeric	Deposit balance of the customers

3.2.9 Apply the Machine Learning Method

When dealing with machine learning algorithms, we want to see how well our machine learning model picks up new information and adapts to it. Prediction errors, which are a measure of the machine learning model's performance and accuracy, are what we actually refer to when we discuss it. Overfitting and under fitting, which are majorly responsible for the poor performances of the machine learning algorithms. The choice of model depends on the type of data available and the specific problem being addressed. Train the machine-learning model using the pre-processed data. This involves feeding the model the input data and adjusting the model's parameters to minimize the error between the model's predictions and the true crop yield data. We develop the model after pre-processing the dataset, and splitting dataset in to training and testing the dataset. We evaluate different models to address optimization problems when working on a machine-learning problem with a given dataset. In machine learning, our goal is to create predictive models that forecast the result given an input of data. We adjust the trained model further to do this. Therefore, in order to determine which candidate model performs the best, we assess their performance.

The ensemble model employed to combines judgments from other models to enhance performance as a whole. In order to enhance model performance, six different algorithms Random Forest, CART, KNN, SVM, Naïve Bayes and Logistic Regression process were merged in this study. On average, many classifications are made for each piece of data. This method uses an average from all the models to arrive at the final forecasts. High-dimensional data handling, non-linear decision boundaries handled by the kernel and reliable generalization performance are all characteristics of SVM. The objective of SVM is to identify an ideal hyper plane that either predicts the target values with the greatest margin of error or maximally divides the samples from distinct classes. The hyper

plane is also known as a decision boundary that divides the classes by maximizing the distance between the nearest samples of various classes. The hyperplane with the biggest margin is the one that SVM seeks for because it is seen to be the best option. The entire dataset is represented by the decision tree algorithm's root node. The approach constructs a child node for each partitioned component in order to continue recursively. The process of selecting the optimal feature and threshold for splitting is repeated until a halting condition is met.

To increase overall prediction performance and robustness, ensemble techniques integrate the predictions of various independent models, such as Random Forest, CART, KNN, SVM, Naïve Bayes and Logistic Regression. Using these six models, here is a general explanation of how ensemble approaches work: First, the training data is used to individually train each model (Random Forest, CART, KNN, SVM, Naïve Bayes and Logistic Regression) using its corresponding algorithm. Then by employing several algorithms or by instructing each model to be trained on various subsets of the training data, ensemble strategies seek to produce diverse individual models. This diversity decreases the risk that all models will make the same mistakes by allowing them to better capture various parts of the data. The ensemble technique combines the predictions of the separate models after they have been trained to produce a final forecast.

3.2.10 Hyper-Parameter Tuning

Setting and managing the values of hyper-parameters that necessitate trial and error is a significant difficulty when dealing with machine learning algorithms. The performance of any model depends greatly on its parameter values, such as the kernel type, and the kernel coefficient for SVM and max_depth, min_samples_split, and min_samples_leaf for DT. By comparing several combinations of hyper-parameters and selecting the one that produces the greatest outcomes, the configuration parameters of a machine learning model are used to internalize the model and forecast the model by learning from the data. Their values can be predicted from the dataset. A machine learning model uses internal parameters to internalize the model, whose values may be guessed from the provided dataset. This allows the model to be predicted by learning from the data. A hyper-parameter is a configuration parameter used to externalize a machine learning model, and its values cannot be inferred from the dataset. The selection of the hyper-parameters affects performance of machine learning algorithm. To optimize the performance of the model,

tuning of hyper-parameters involves selecting the best combination of the parameters. There are three different types of hyper-parameter tuning techniques, but the two that are most frequently used are grid search tuning techniques and random search tuning techniques. We have chosen to use grid search for our models because random hyper-parameter setting makes it difficult to compare the performance of different algorithms; grid search works better by generating a grid of all possible hyper-parameter values. In order to analyze the range of hyper-parameters in a machine-learning model, grid search is one of the tuning approaches employed. Values of hyper-parameter, it is better approach which work by creating a grid of all possible hyper-parameter values. Grid search is the tuning techniques that used in machine learning model to evaluate the range of hyper-parameters.

3.2.11 Model Evaluation

This section presents the evaluation method that was employed to evaluate the performance of the deposit classification models. The first requirement for comparing performance between different learner models is to determine a method of evaluation. One of the popular performance evaluation methods in Machine Learning, Data mining, artificial intelligence and statistics is confusion matrix. In order to quantify the performance of a problem that contains two classes, confusion matrix is usually used (Glas, 2015). The research used Confusion matrix as evaluation methods because it comprises of evidence about actual and predicted classification performed by the proposed prediction and classification model.

The proposed work was evaluated by comparing the output against the manually observed phenomena using a confusion matrix and by conducting a comparison with other algorithms that are frequently used by previous works. The confusion matrix is a 2 by 2 table with 4 outcomes, which is true positive (TP), true negative (TN), false positive (FP), and false-negative (FN). These outcome measures are used to describe the performance of a classification model on a set of data.

This section will present a number of common statistics for measuring the performance of machine learners. These statistics will range from simple measures of model accuracy to measures of specific characteristics of the models as, for example, the proportion of relevant instances when dealing with deposit mobilization.

3.2.12 Confusion Matrix

The confusion matrix is used to measure the performance of two-class problem for the given dataset. The right diagonal elements TP (true positive) and TN (true negative) correctly classify instances as well as FP (false positive) and FN (false negative) incorrectly classify instances.

TN = the number of incorrect classifications that an instance is Negative.

FP = the number of incorrect classifications that an instance is positive.

FN = the number of correct classifications that an instance is Negative.

TP = the number of correct classifications that an instance is Positive.

		True Class	
		Positive	Negative
Hypothesized Class	Positive	TP	FP
	Negative	FN	TN

Figure 3.2 Confusion Matrix.

Total number of instances = correctly classified instance + incorrectly classified instance. Correctly classified instance = TP + TN. incorrectly classified instance = FP + FN. We can calculate the value of true positive rate, true negative rate, false positive rate and false negative rate by methods shown below (Glas, 2015).

- Accuracy (AC)

The accuracy is the total number of all correct predictions, TP and TN divided by the total number of the dataset. The best accuracy rate is 1.0, and 0.0 is the worst rate (Sunita & Chavan, 2013).

$$Accuracy = TP + TN \div TP + TN + FP + FN \qquad \textit{Equation 2.1}$$

- Precision

Precision positive prediction is the number of correct positive predictions divided by the total number of positive predictions, TP and FP. The best precision is 1.0, and the worst is 0 (Sunita & Chavan, 2013).

$$Precision = TP \div TP + FP \qquad \textit{Equation 2.2}$$

- False Positive Rate (FPR)

The false-positive rate is the number of incorrect negative predictions divided by the total number of negatives, TN and FP. The ideal false positive rate is 0.0, and the worst is 1.0 (Glas, 2015).

$$FPR = FP \div TN + FP \qquad \textit{Equation 2.3}$$

- False Negative Rate (FNR)

The false-negative rate is the number of incorrect positive predictions divided by the total number of negatives, FN and TP. The best false negative rate is 1.0, and the worst 20 is 0.0 (Glas, 2015).

$$FNR = FN \div FN + P \qquad \textit{Equation 2.4}$$

- Error Rate (ERR)

The error rate is the number of all incorrect predictions, FP and FN divided by the total number of the dataset. The best error rate is 0.0, and 1.0 is the worst rate (Sunita & Chavan, 2013).

$$ERR = FP + FN \div TN + TP + FN + FP \qquad \textit{Equation 2.5}$$

3.2.13 Implementation Tools

This research used experimentation of proposed method and application development environments that are required to produce the output of the dataset using Spider with Python programing Tools. Python programming tools is the language of Data science that integrated suite of software facilities for data manipulation, calculation and graphical facilities useful for experiment and analysis dataset. It is preferred because of the following reasons: Python has easy and friendly use; Interface-Studio has ability to present datasets in the form of figures with variety of presentations. In Python it is possible to link datasets in common simple format such as .CSV; it is simple and suitable for technical computing; it is also effective data handling and storage; Suite of operators for calculations on arrays and large, coherent, integrated collection of intermediate tools for data analysis (Harrington, 2010), (Verma, 2022)and (Han, 2006). In general, Python is selected for rules mining and MS Excel is employed for preprocessing the dataset.

This study uses multiple implementation tools and packages to implement a proposed prediction, model. A Python-based programming language for putting each suggested fix into practice, from data preparation through model construction, as well as for reviewing implementations and

potential classifier models. Because Python is the preferred programming language for researchers, developers, and data scientists who need to work with machine learning models, it is employed in this study. The implementation tools and Python packages used in this study are included in the table below, together with information on their versions and descriptions.

Table 3.2 Description of the Tools and Python Packages used During the Implementation

No	Tool	Explanation
1	Python 3.9.0	A compile environment for python code easy to learn, powerful programming language to develop a machine learning application.
2	Jupitor notebook	Easy cross-platform code editor well-known for its speed, comfort of use. It's an incredible editor right out of the box, but the real power comes from the ability to enhance its functionality using Package Control and creating custom settings.
3	Microsoft Excel 2013	Used data preparation tasks in cleaning, filtering, sorting the collected data, and remove duplicated data Also, used to manage the explanation task.
4	Scikit-learn 0.24.2	A set of python modules for machine learning and data mining. This study uses it for feature extraction and training and testing model. The name of the package is called sklearn.
5	Pandas 1.2.4	High-performance, easy-to-use data structures, and data analysis tools. This study uses it for data reading, manipulation, writing and handling the data frame.
6	NumPy 1.20	Array processing for number, strings, and objects.
7	Matplotlib	It is a comprehensive plotting library in Python that provides a wide range of customizable plots and visualizations
8	Seaborn	Seaborn is a high-level data visualization library built on top of matplotlib and It provides a simplified interface to create attractive and informative statistical graphics.

3.3 Experiment

The experiment was carried out to construct a model with their analysis based on the steps mentioned in the proposed framework architecture. This experimental evaluation methods used five base models and an ensembled model to compare the performances of each model based on accuracy measures to ensure the realization of the proposed framework architecture. The selected weak learn models used in this experiment were Logistic regression, K-Neighbors Classifier (KNN), Decision Tree Classifier (Cart), Support Vector Machine (SVM), and, Gaussian NB (bayes). In addition, to build ensemble model for converting the five weak learning models into an ensemble learning model a blending technique with logistic regression algorithm was implemented to compare the ensemble model generalization performance with that of the respective performance of the individual base models used in the experiment. Additionally, to determine the feature importance level of each Variable Random Forest Classifier model with various thresholds were derived.

3.3.1 Importing Important Libraries for Data Analysis

To design and implement the proposed models for the study various machine learning libraries and models were imported and modified to fit for the proposed tasks. The major libraries imported were used to perform data manipulation, oversampling techniques, feature selection, and classification tasks.

This comprehensive setup was used to address multiple aspects of classification tasks, including data preprocessing, handling imbalanced datasets, utilizing diverse base model classifiers, and potentially improving predictive performance through blending ensemble method, where used to determine the classification tasks to identify key attributes that lead to better deposit mobilization performance.

Furthermore, to better suit the dataset for the proposed experiments, various tools and techniques were used to handle imbalanced datasets, such as Synthetic Minority Over-sampling Technique (SMOTE), specifically SMOTE, and SVMSMOTE. Furthermore, Borderline SMOTE which

generates synthetic samples to balance class distributions was used and to handle categorical variables in the dataset preprocessing techniques such as Standard Scaler and encoders called LabelEncoder, OrdinalEncoder, and OneHotEncoder were used. Hence, the following libraries were imported from Python libraries.

```
# Blending Ensemble for Classification Problems
from numpy import hstack
from collections import Counter
import pandas as pd
import numpy as np
from numpy import sort
from imblearn.over_sampling import SMOTE
from imblearn.over_sampling import SVMSMOTE
from imblearn.over_sampling import BorderlineSMOTE
from sklearn.feature_selection import SelectFromModel
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.metrics import precision_score,recall_score,accuracy_score, con
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
from sklearn.naive_bayes import GaussianNB
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OrdinalEncoder
from sklearn.preprocessing import OneHotEncoder
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib import pyplot
```

Figure 3.3 Essential Libraries Imported for Data Analysis and Experimentation

3.3.2 Data Exploration and Visualization

This section describes the tasks used to explore the datasets used for the study. It starts with describing dataset used, identifying features important for the classification task, preprocessing the dataset to clean the data and transforming the datasets to relevant format. It ends with the machine learning models and what parameters were used.

3.3.2.1 Dataset Description

The dataset used for this study was collected from Cooperative bank of Oromia. The dataset contains information about e-banking services provided and customer details in .csv file format. With the consultation of domain expert, all variables for the classification tasks were selected. These variables are total_deposit, other_credit, other_debt, mobile_debt, card_credit, card_debt, mobile_banking, gender, age, marital_status, marital status, job and ATM card.

In addition, one of the variables is used as a target variable known as a class variable that indicates the amount of deposit mobilized by each e-service variable categorized 'total_deposit' large deposits as "HIGH", Intermediate amounts of deposits as "MEDIUM" and small amounts of deposits as "LOW". In this study deposit, mobilization is stated as a tertiary classification problem.

3.3.2.2 Data Preprocessing

To make the data ready for experimentation, data exploration and manipulation tasks on the dataset were performed. Initially, missing values from the dataset were handled. Subsequently, the mode for the 'gender' attribute and the 'age' attribute was calculated, encoding 'Male' as 0 and female as 1, and the mode value for 'age' which is 38 was encoded as 0. Furthermore, the mean deposit amount for 'total_deposits' was computed which revealed an average deposit of $83,904.09. Additionally, the mode for the 'atm_card' attributes was determined and record with a missing value was identified and handled.

Moreover, data-cleaning operations were applied to make the dataset suitable for data analysis. These include filling missing values in the 'atm_card' column with 'No', rechecking for null values, and the cleaned dataset was exported in CSV format and saved for later use. The cleaned dataset Similarly, categorical features such as 'gender', 'marital_status', and 'job' were selected from the cleaned dataset for further analysis or processing. As part of exploratory data analysis steps, handling missing values, computing summary statistics, and cleaning operations aimed at preparing and refining the dataset, indicating an effort to ensure data quality and suitability for subsequent analysis or modeling tasks were carried out. For the stated asks, the following python codes were employed as shown in Figure 3.4.

```
df = pd.read_csv('eservices.csv')
# df.info()
```

```
# df. isnull ().sum()
df.isnull ().sum()
df ['gender']. mode ()
df ['age']. mode()
“Age encoded as 0 and its mode is 38
df ['total_deposit']. mean ()
The mean deposit amount is found to be 83904.09
df ['atm_card']. Mode ()
df_cleaned = pd.read_csv('eservices_cleaned.csv',low_memory = False)
```

Figure 3.4 Python code used for preprocessing tasks

3.3.2.3 Transforming the Dataset

Categorical variables available in the dataset were transformed into a one-hot encoded format, which is the most commonly used format in machine learning models to represent categorical data numerically. The transformed dataset was further parametrized to avoid multicollinearity by dropping the first column by setting ‘drop=True’. The categorical features are further transformed into a binary matrix format using the encoder. The resulting transformed features are stored as a new data frame with columns labeled as per the one-hot encoding, such as 'gender_m', 'gender_f', 'marital_status_m', etc . Each column represents the different categorical values of the original attributes after encoding.

Subsequently, the predictor variables (X) and the target variable (y) were extracted from the categorized Data Frame, with 'X' containing the features and 'y' representing the 'class' column. The shape of 'X' reveals it contains 82,593 samples with 13 features, while 'y' corresponds to 82,593 labels. The features and target variables extracted were made ready for further modeling.

This process primarily focused on preparing the dataset for predictive modeling by converting categorical attributes into a format suitable for machine learning algorithms. The utilization of one-hot encoding enables the conversion of categorical data into a binary matrix, allowing machine learning models to interpret and analyze categorical variables effectively. Hence, the successful transformation of the dataset into a format suitable for machine learning indicates a preparation stage aimed at training predictive models to perform classification or regression tasks on this prepared data.

To prepare the dataset for machine learning the input features (X) and the target variable (y) were identified from the previously cleaned dataset. All rows and all columns except the last column were assigned as input features and the last column was assigned as the target variable or the class labels.

Similarly, the following activities were performed, data shapes were used to display the shapes of the feature matrix 'X' and the target variable 'y'. In this case, 'X' has a shape of (82593, 13) and 'y' has a shape of (82593,). This means 'X' contains 82,593 samples/rows and 13 features, while 'y' contains 82,593 labels. After the segmentation of features and the target variable, the data was made ready for machine learning tasks, using the input features and the corresponding target labels for training the proposed experimental models. The following Python code was used as shown in Figure 3.5.

```
# # One hot encode input variables
# onehot_encoder = OneHotEncoder(drop='first', sparse=False)
# cat_features = onehot_encoder.fit_transform(cat_features)
# cat_features = pd.DataFrame(cat_features)
# cat_features
# cat_features.columns = ['gender_m','gender_f','marital_status_m','marital_
# cat_features
X = df_cleaned.iloc[:,:-1]
y = df_cleaned['class']
feature_list = list(X.columns)
X.shape,y.shape
```

Output

((82593, 13), (82593,))

Figure 3.5 Feature Encoding and Preparation

3.3.2.4 Class Distribution Analysis

In this section, Class distribution analysis was performed by counting the occurrence of each class label in the 'class' column using a counter, then calculating and displaying the percentage of each class in the dataset. Finally, a bar chart to visualize the distribution of the three class classes. The output of the class distribution analysis indicated that:

- Class 2 has 23,619 occurrences, making up about 28.597% of the dataset.
- Class 1 has 14,634 occurrences, accounting for approximately 17.718%.

- Class 0 has the highest occurrences, with 44,340 instances, constituting around 53.685% of the dataset.

The sample code shown in Figure 3.6 was used to perform exploratory data analysis, summary statistics, and basic visualization techniques to understand the dataset's characteristics and class distribution as well as

```
counter = Counter(y)
for k,v in counter.items():
per = v / len(y) * 100
print('Class=%d, n=%d (%.3f%%)' % (k, v, per))
# plot the distribution
pyplot.bar(counter.keys(), counter.values())
pyplot.show()
```

Figure 3.6 Sample code used for class type analysis

The output of data analysis results are summarized as shown in Table 3.3 under.

Table 3.3 Class Labels and Percentage of each Class in the Dataset

Class type	**No. of observation(n)**	**%Percentage**
Class=2	n=23,619	(28.597%)
Class=1	n=14,634	(17.718%)
Class=0	n=44,340	(53.685%)

Similarly, Bar chart is used to visualize the three class labels in the dataset. The x-axis represents the class labels, and the y-axis shows their respective counts. The output provided demonstrates the distribution of classes within the dataset: Accordingly, Class 2: 23,619 instances, making up about 28.597% of the dataset. Class 1: 14,634 instances, accounting for approximately 17.718%, and Class 0: 44,340 instances, constituting around 53.685% of the dataset. Figure 3.7. shows the distribution of classes in the dataset as depicted under.

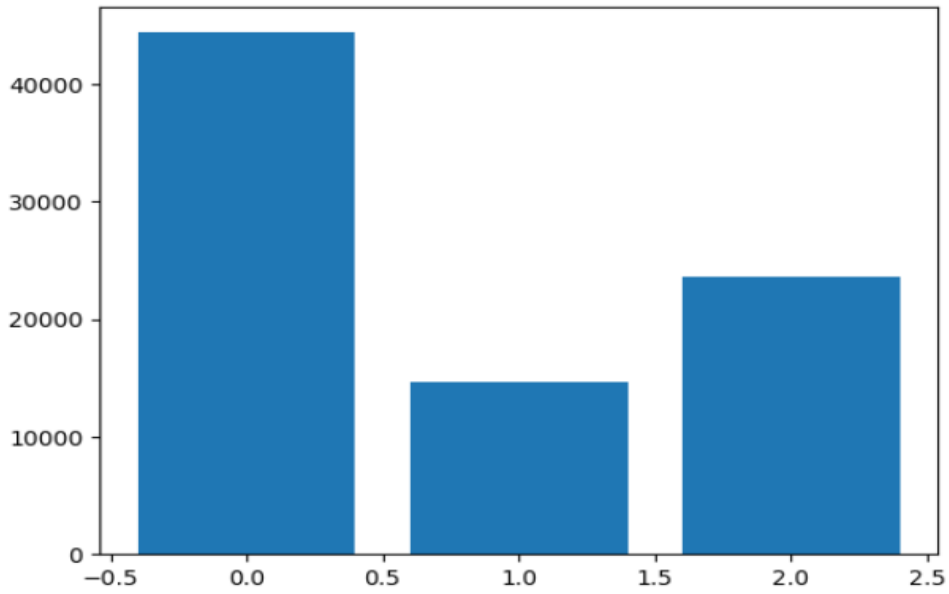

Figure 3.7 Distribution of classes in the dataset

3.3.2.5 Transforming the Imbalanced Dataset

In this section, to mitigate the impact of class imbalance on model building due to having an imbalanced dataset, SMOTH (Synthetic Minority Over-sampling Technique) technique is used. This technique synthesizes new instances of the minority classes by interpolating between existing instances. Furthermore, over-sampling was performed on the dataset by applying the SMOTE technique. This technique is often used to address class imbalance by generating synthetic samples for the minority classes. Specifically, it aims to oversample the minority classes (Classes 1 and 2) to balance the class distribution. After oversampling, the new shapes of (133,020, 13) (features) and (133,020,) (labels) were generated. After the application of oversampling on the dataset with SMOTE, the results indicated that the dataset now contains 133,020 samples and 13 features in the 'X' variable, and 133,020 labels in the 'y' variable. The minority classes have been oversampled to address the class imbalance issue.

Specifically, to generate synthetic samples for the minority classes in the dataset both BorderlineSMOTE, and SVMSMOTE variations of the SMOTE algorithms were applied. These algorithms are designed to focus on different aspects of the dataset, especially in the regions where the decision boundary is harder to distinguish between classes. After applying these oversampling techniques, the impact was assessed on the dataset's balance, distribution, and performance of machine learning models intended to train with this data. Similarly, proper evaluation, such as

cross-validation and validation on an unseen test set, was done to ensure the effectiveness of the oversampling methods in improving the model's generalization and handling class imbalance.

To address the class imbalance in the dataset, ADASYN (Adaptive Synthetic Sampling) is used. ADASYN generates synthetic samples for the minority classes, adapting the generation of new samples based on the density distribution of the classes. This results in an equal representation of all classes in the dataset after oversampling. The output indicates that after applying ADASYN, the dataset now contains an equal distribution of all classes as described below.

- Class 2: 44,340 instances, approximately 33.333% of the dataset.
- Class 1: 44,340 instances, also about 33.333%.
- Class 0: 44,340 instances, again approximately 33.333%.

The sample code used for this task is shown under.

```
# from imblearn.over_sampling import ADASYN
# # transform the dataset
# oversample = ADASYN()
# X, y = oversample.fit_resample(X, y)

# summarize distribution
counter = Counter(y)
for k,v in counter.items():
per = v / len(y) * 100
print('Class=%d, n=%d (%.3f%%)' % (k, v, per))
# plot the distribution
pyplot.bar(counter.keys(), counter.values())
pyplot.show()
```

Figure 3.8 the python code used for Distribution of classes in the dataset

A bar chart is used to visualize the distribution of the three class labels after oversampling with ADASYN which helps to create a more balanced dataset by emphasizing the area where that data is sparsely distributed. The resulting classes in the dataset are shown in Figure 3.9. below.

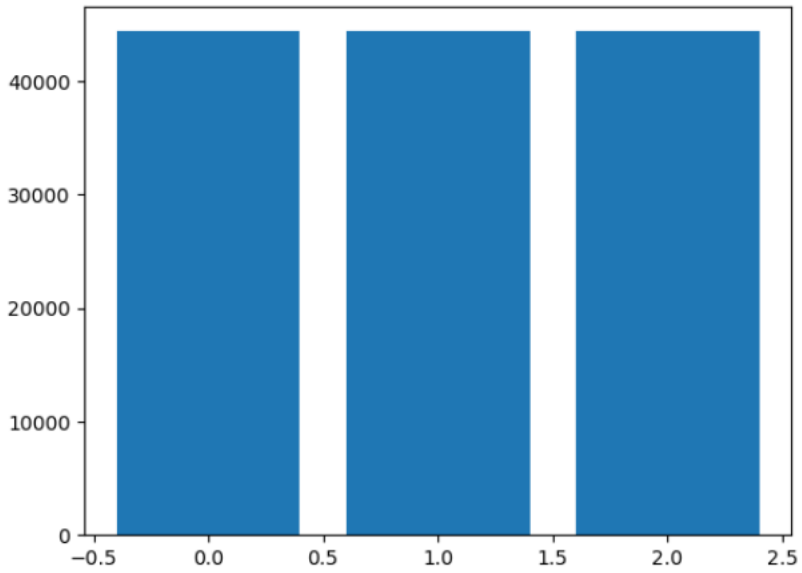

Figure.3.9 Oversampled distribution of the three classes in the dataset

3.3.2.6 Application of Feature Scaling

Feature scaling is a crucial preprocessing step in machine learning that standardizes or normalizes the range of features in a dataset. The StandardScaler object and the 'fit_transform' method are used to scale the feature matrix 'X'. This transformation ensures that each feature within 'X' has a mean of 0 and a standard deviation of 1, allowing machine learning models to handle features on a comparable scale. The aim is to partition the dataset into training and validation sets using 'train_test_split' with an 80:20 ratio.

3.3.3 Describing Standalone/Base Learners

Base learners are individual algorithms defined and each model performance was independently evaluated. These models will be also used within ensemble methods to collectively improve the classification performance of the model. The initialized weak learners provide a starting point for model experimentation, comparative analysis, or utilization within ensemble techniques, allowing for flexibility and diversity in model selection for various machine-learning tasks.

In this study, the most popular baseline classifiers were used as benchmarked and each classifier was assigned a label as ('lr', 'knn', 'cart', 'svm', 'bayes') along with an instance of the respective as shown in the Figure 3.10.

```
# Our weak learners
def get_models():
    models = list()
    models.append(('lr', LogisticRegression()))
    models.append(('knn', KNeighborsClassifier()))
    models.append(('cart', DecisionTreeClassifier()))
    models.append(('svm', SVC()))
    models.append(('bayes', GaussianNB()))
    return models
```

Figure 3.10 Weak learners' models used for classification tasks

The function constructs a list named 'models' and sequentially appends tuples containing a label and an instance of a specific classifier. The classifiers included are:

- **Logistic Regression ('LR'):** A linear model used for binary classification that estimates the probability of a binary outcome.
- **K-Nearest Neighbors ('KNN'):** A non-parametric algorithm used for classification and regression by identifying the class or value based on its neighbors in the feature space.
- **Decision Tree Classifier ('cart'):** A model employing a tree-like structure where internal nodes represent features, aiming to split data based on feature values to make predictions.
- **Support Vector Machine ('SVM'):** A versatile supervised learning algorithm for classification, regression, and outlier detection, effective in high-dimensional spaces through the creation of hyperplanes separating classes.
- **Gaussian Naive Bayes ('bayes'):** A probabilistic classifier based on applying Bayes' theorem with the assumption of independence among features, often used in text classification and simple learning tasks.

The five classification algorithms models are built and their individual performance is evaluated to determine their effectiveness in predicting deposit mobilization based on data

collected. The following Python code is used as shown below in figure 3.11.

```
# evaluate standalone model
for name, model in models:
 # fit the model on the training dataset
 model.fit(X_train_full, y_train_full)
 # make a prediction on the test dataset
 yhat = model.predict(X_test)
 # evaluate the predictions
 score = accuracy_score(y_test, yhat)
 # report the score
 print('>%s Accuracy: %.3f' % (name, score*100))
```

Figure 3.11 Weak learners' Python code is used classification tasks

3.3.4 Fitting the Blending Ensemble

Ensemble Learning, on the other hand, refers to a learning methodology that combines several baseline models to build a bigger single yet more powerful model than its constituents (Kumar et al., 2021). Also, ensemble learning can reduce the risk of overfitting thanks to the diversity of baseline models. In this study, we hypothesized that the ensemble would give a higher performance than the single baseline by integrating the five baseline models used in this study in the same framework to obtain a stronger model that outperforms them. The success of an ensemble method depends on several factors, including how the baseline models are trained and how they are combined.

The blending technique is used in ensemble models for converting the five weak learning models into an ensemble learning model. Subsequently, compare the ensemble model generalization performance with that of the Random Forest model performance being used as the base estimator.

A fit ensemble function is used to construct and train a blending ensemble by integrating multiple base models. This process involves fitting each model from the 'models' list on the provided training set. Within a loop that iterates through the models, each model undergoes training to learn from the features and target labels. Subsequently, these models predict outcomes on the validation set to capture their predictions regarding the validation data. These predictions are reshaped into a columnar format and stored in a list to assemble a collection of predictions from the base models. The blending ensemble was designed as shown below in the Figure. 3.12.

```
# Fitting our Blending Ensemble
def fit_ensemble(models, X_train, X_val, y_train, y_val):
    # Fitting all the models
    meta_X = list()
    for name, model in models:
        # Fit in training set
        model.fit(X_train, y_train)
        # Predict on hold out set
        yhat = model.predict(X_val)
        # Reshape predictions into a matrix with one column
        yhat = yhat.reshape(len(yhat), 1)
        # Store predictions as input for blending
        meta_X.append(yhat)
    # 2D array from predictions, each set is an input feature
    meta_X = hstack(meta_X)
    # Defining a blending model
    blender = LogisticRegression(solver='lbfgs', max_iter=1000 )
    # fit on predictions from base models
    blender.fit(meta_X, y_val)
    return blender
```

Figure 3.12 Fitting the Ensemble Model

Later accumulating the predictions, 'hstack()' is utilized to horizontally concatenate these predictions, creating a 2D array ('meta_X') where each set of predictions from the base models serves as an input feature for the blending process. A 'blender' model, specified here as a Logistic Regression model ('LogisticRegression') with specific solver and iteration settings, is then instantiated. This 'blender' model is trained on the aggregated predictions ('meta_X') and the corresponding labels from the validation set ('y_val'). The 'blender' model learns from the diverse outputs of the base models, aiming to combine these predictions effectively, potentially improving overall predictive performance and generalization on unseen data. Finally, the trained 'blender' model is returned, poised to make predictions by synthesizing outputs from the base models, constituting a crucial component of the blending ensemble for subsequent prediction tasks in the project.

3.3.5 Predictions with the Blending Ensemble

The 'fit_ensemble()' function is used to creation and train a blending ensemble by integrating multiple base models. This process involves fitting each individual model from the 'models' list on

the provided training set ('X_train' and 'y_train'). Within a loop iterating through the models, each model is trained on 'X_train' to learn from the features and target labels. Later, these models predict outcomes on the validation set ('X_val') to capture their predictions regarding the validation data. These predictions are reshaped into a columnar format and stored in a list ('meta_X') to assemble a collection of predictions from the base models.

After accumulating the predictions, 'hstack()' is utilized to horizontally concatenate these predictions, creating a 2D array ('meta_X') where each set of predictions from the base models serves as an input feature for the blending process. A 'blender' model, specified here as a Logistic Regression model ('LogisticRegression') with specific solver and iteration settings, is then instantiated. This 'blender' model is trained on the aggregated predictions ('meta_X') and the corresponding labels from the validation set ('y_val'). The 'blender' model learns from the diverse outputs of the base models, aiming to combine these predictions effectively, potentially improving overall predictive performance and generalization on unseen data. Finally, the trained 'blender' model is returned, poised to make predictions by synthesizing outputs from the base models, constituting a crucial component of the blending ensemble for subsequent prediction tasks in the project. The code used for this task is shown below in Figure 3.13

```
# Predictions with the blending ensemble
def predict_ensemble(models, blender, X_test):
    # Make predictions with base models
    meta_X = list()
    for name, model in models:
        # Predict with the base model
        yhat = model.predict(X_test)
        # Reshape the matrix
        yhat = yhat.reshape(len(yhat), 1)
        # store prediction
        meta_X.append(yhat)
    # Create 2D array from predictions, each set is an input feature
    meta_X = hstack(meta_X)
    # Predict
    return blender.predict(meta_X)
```

Figure 3.13 Making predictions with the blending ensembles

The 'predict_ensemble()' function aims to generate predictions using the blending ensemble previously trained and composed of multiple base models. It begins by initializing an empty list 'meta_X' to store predictions made by the individual base models on the provided test dataset

('X_test'). Through a loop iterating over the models, each base model predicts outcomes on the test data using 'model.predict(X_test)'. These predictions are reshaped to form a columnar structure ('yhat.reshape(len(yhat), 1)') and are subsequently appended to the 'meta_X' list, aggregating the individual predictions from the base models.

Once all predictions are accumulated, 'hstack()' is employed to horizontally concatenate these predictions, creating a 2D array ('meta_X') where each set of predictions from the base models constitutes an input feature for the blending process. Finally, the trained 'blender' model, representing the fusion of base model predictions, is utilized to make predictions on the aggregated test set predictions ('meta_X'). The 'blender.predict(meta_X)' function call synthesizes these diverse base model outputs using the learned blending techniques, providing predictions on the test dataset. This process of aggregating and synthesizing predictions from the base models through the blending ensemble serves to leverage the collective insights of the individual models, potentially enhancing predictive accuracy and robustness on unseen data.

3.3.6 Defining the Dataset for the Model

The fundamental steps in model building is preparing a dataset for machine learning analysis. Initially, the dataset 'X' containing the features and 'y' representing the target variable are split into training and testing subsets using the 'train_test_split' function. This division assigns 85% of the data to the training set ('X_train_full' and 'y_train_full') while allocating the remaining 15% to the test set ('X_test' and 'y_test'). The parameter 'random_state=1' ensures reproducibility by fixing the random seed for consistent data splitting across runs. The code used for this purpose is shown below in Figure. 3.14.

```
# Define dataset # Split dataset into train and test sets
X_train_full, X_test, y_train_full, y_test = train_test_split(X, y, test_size=0.15, random_state=1)
# Spliting training set into train and validation sets
X_train, X_val, y_train, y_val = train_test_split(X_train_full, y_train_full, test_size=0.33,
random_state=1)
# Summarize data split
print('Train: %s, Val: %s, Test: %s' % (X_train.shape, X_val.shape, X_test.shape))
```

Figure 3.14 Code used for splitting the dataset

The fundamental steps in model building is preparing a dataset for machine learning analysis. Initially, the dataset 'X' containing the features and 'y' representing the target variable are split into training and testing subsets using the 'train_test_split' function. This division assigns 85% of the data to the training set ('X_train_full' and 'y_train_full') while allocating the remaining 15% to the test set ('X_test' and 'y_test'). The parameter 'random_state=1' ensures reproducibility by fixing the random seed for consistent data splitting across runs.

Following the initial split, a further segmentation of the training data was made using 'train_test_split' once more, segregating 33% of the training data as a validation set ('X_val' and 'y_val'). This nested split designates 67% of the original data for actual model training ('X_train' and 'y_train') while setting aside a distinct portion for validation purposes, facilitating model evaluation and hyperparameter tuning without directly using the test data. The printed output provides a summary of the sizes of each segmented dataset, showing the shapes of the training set (75,754 samples and 13 features), validation set (37,313 samples and 13 features), and test set (19,953 samples and 13 features). These summary statistics offer insights into the distribution of data across the training, validation, and test subsets, ensuring proper division for subsequent model training, validation, and assessment stages in the machine learning process.

3.3.7 Building the Blending Ensemble Model

Based on the important features selected, the important features identified were used to initialize the base models, train the blending ensemble, make prediction on the test set, and evaluate the predictions. The python code used for this task is shown in Figure 3.15 below.

```
# Initializing the base models
models = get_models()
# Train the blending ensemble
blender = fit_ensemble(models, X_train, X_val, y_train, y_val)
# Make predictions on the test set
yhat = predict_ensemble(models, blender, X_test)
# Evaluate predictions
score = accuracy_score(y_test, yhat)
print('Blending Accuracy: %.3f' % (score * 100))
```

```
Blending Accuracy: 98.496
```

Figure 3.15 The python code used for building ensemble

To greatly improve the predictive ability of a KNN, we can use ensemble technique to combine their results. The blending technique indeed does this for us. A Logistic Regression model is employed to make predictions on the test dataset 'X_test'. The model generates predictions using 'model.predict(X_test)', storing these predictions in 'y_pred_test'. Subsequently, a list comprehension is utilized to round the predicted values ('y_pred_test') to the nearest integer, ensuring compatibility with the target variable format. These rounded predictions are stored in the 'predictions' list. Following this, the code computes the accuracy of the model's predictions using the 'accuracy_score()' function from scikit-learn. This function compares the predicted values ('predictions') against the actual target labels from the test set ('y_test'), calculating the proportion of correctly classified samples. The results are also very good, showing that the overall accuracy rate is 98.496%.

3.3.8 Importance Level of Each Feature

A RandomForestClassifier model is employed to make predictions on the test dataset 'X_test'. The model generates predictions using 'model.predict(X_test)', storing these predictions in 'y_pred_test'. Subsequently, a list comprehension is utilized to round the predicted values ('y_pred_test') to the nearest integer, ensuring compatibility with the target variable format. These rounded predictions are stored in the 'predictions' list. Following this, the code computes the accuracy of the model's predictions using the 'accuracy_score()' function from scikit-learn. This

function compares the predicted values ('predictions') against the actual target labels from the test set ('y_test'), calculating the proportion of correctly classified samples.

The generated output displays the RandomForestClassifier achieved an accuracy of 97.77%, signifying its ability to accurately classify approximately 97.77% of the samples within the test dataset. The second line presents the RandomForestClassifier achieved an accuracy of 97.77%, signifying its ability to accurately classify approximately 97.77% of the samples within the test dataset. The second line presents the accuracy score with a precision of three decimal places, providing the same accuracy value represented as 97.77. These accuracy metrics serve as crucial performance indicators, quantifying the model's success rate in correctly predicting the class labels on the unseen test data. An accuracy of 97.77% or 0.978 implies the model's effectiveness in making predictions on the provided test dataset, reflecting its overall predictive capability of predicting deposit mobilization variables. The code used for this task is shown below in Figure 3.16.

```
model = RandomForestClassifier()
model.fit(X_train_full, y_train_full)
# make predictions for test data and evaluate
```

```
RandomForestClassifier()
```

```
y_pred_test = model.predict(X_test)
predictions = [round(value) for value in y_pred_test]
accuracy = accuracy_score(y_test, predictions)
print("Accuracy: %.2f" % (accuracy * 100.0))
print("Accuracy: %.3f" % (accuracy))
```

```
Accuracy: 97.77
Accuracy: 0.978
```

Figure 3.16 Random Forest model code and result

3.3.9 Fit Model Using Feature Importance

To determine the feature importance of a RandomForestClassifier model various thresholds were derived. For each threshold value obtained features meeting the specific threshold were selected. Subsequently, the training set ('X_train_full' and 'y_train_full') is transformed to contain only the selected features ('select_X_train') based on the determined threshold. A new

RandomForestClassifier ('selection_model') is then trained using this modified training set. The Phyton code use for this task is shown below in Figure. 3.17.

```
# Fit model using each importance as a threshold
thresholds = sort(model.feature_importances_)
for thresh in thresholds:
    # select features using threshold
    selection = SelectFromModel(model, threshold=thresh, prefit=True)
    select_X_train = selection.transform(X_train_full)
    # train model
    selection_model = RandomForestClassifier()
    selection_model.fit(select_X_train, y_train_full)
    # eval model
    select_X_test = selection.transform(X_test)
    y_pred_test = selection_model.predict(select_X_test)
    predictions = [round(value) for value in y_pred_test]
    accuracy = accuracy_score(y_test, predictions)
    print("Thresh=%.3f, n=%d, Accuracy: %.4f" % (thresh, select_X_train.shape[1], accuracy))
```

Figure 3.17 Code used for fit mode

Upon training, the feature selection process is applied to the test set, selecting the same features as identified in the training set. Predictions are generated using the 'selection_model.predict()' function on this modified test set, and these predicted values are rounded and stored in 'predictions'. The accuracy metrics for each threshold level were evaluated against the true labels using an accuracy score. The accuracy achieved was obtained using the RandomForestClassifier model. The output generated a series of accuracy scores obtained by varying the feature selection threshold. It demonstrates how altering the threshold impacts the number of features ('n') selected for model training and the resultant accuracy. As the threshold increases, fewer features are considered relevant for training the model, leading to a reduction in the number of features ('n') utilized and, in some cases, a corresponding decline in accuracy. The reported accuracies illustrate the trade-off between feature selection and predictive performance, aiding in determining an optimal balance between feature subset size and model accuracy for this specific RandomForestClassifier. A better accuracy is achieved when the number of features(n) reaches 12. The result is shown in Figure 3.18

```
Thresh=0.026, n=13, Accuracy: 0.9771
Thresh=0.031, n=12, Accuracy: 0.9774
Thresh=0.041, n=11, Accuracy: 0.9745
Thresh=0.042, n=10, Accuracy: 0.9694
Thresh=0.046, n=9, Accuracy: 0.9626
Thresh=0.048, n=8, Accuracy: 0.9561
Thresh=0.059, n=7, Accuracy: 0.9547
Thresh=0.060, n=6, Accuracy: 0.9468
Thresh=0.061, n=5, Accuracy: 0.9285
Thresh=0.081, n=4, Accuracy: 0.9014
Thresh=0.126, n=3, Accuracy: 0.8753
Thresh=0.181, n=2, Accuracy: 0.7905
Thresh=0.198, n=1, Accuracy: 0.6365
```

Figure 3.18 Fit model feature importance results based on threshold value

CHAPTER FOUR

4 RESULTS AND DISCUSSION

4.1 Overview

This chapter discusses the experiment result with their analysis based on the steps mentioned in the proposed framework architecture. This experimental evaluation methods used five base models and an ensembled model to compare the performances of each model based on accuracy measures to ensure the realization of the proposed framework architecture. The selected weak learn models used in this experiment were Logistic regression, K-Neighbors Classifier (KNN), Decision Tree Classifier (Cart), Support Vector Machine (SVM), and, Gaussian NB (bayes). In addition, to build ensemble model for converting the five weak learning models into an ensemble learning model a blending technique with Logistic regression algorithm was implemented to compare the ensemble model generalization performance with that of the respective performance of the individual base models used in the experiment. To identify feature importance level of each attribute in determining deposit class Random Forest Classifier model is employed.

4.2 Results of Standalone Base Models

In this section the five classification algorithms namely Logistic Regression(lr), Neighbors classifier(knn), Decision Tree Classifier(cart), Support Victor Machine (SVM), and GaussianNB (bayes) models are built and their individual performance results are discussed. The following Python code is used as shown below in figure 4.1.

```
# evaluate standalone model
for name, model in models:
 # fit the model on the training dataset
 model.fit(X_train_full, y_train_full)
 # make a prediction on the test dataset
 yhat = model.predict(X_test)
 # evaluate the predictions
 score = accuracy_score(y_test, yhat)
 # report the score
 print('>%s Accuracy: %.3f' % (name, score*100))
```

```
>lr Accuracy: 92.733
>knn Accuracy: 98.491
>dt Accuracy: 94.116
>svc Accuracy: 98.401
>gnb Accuracy: 89.225
```

Figure 4.1 Python code used to evaluate standalone models

4.2.1 Results of Logistic Regression Model

The Logistic Regression model was developed to identify prospective not deposit and deposit customers in CBO like in other models. The experiment was conducted using the CBOs dataset which contains 82,593 observations and 13 attributes. The model performance was evaluated through the accuracy metric. It yielded with accuracy values of 92.73%. These accuracy values were very good as (Nicholas & Scott, 2021)estimates high accuracy to be in the range of 0.75 and 1 representing 75% and 100% respectively.

4.2.2 Results on K-Nearest Neighbors (KNN) Classifier

The Support K-Nearest Neighbors (KNN) Classifier yielded an accuracy of 98.491%. A data point is classified using KNN, a non-parametric technique, according to the majority class of its closest neighbors. It works remarkably well, nearly matching the accuracy of the ensemble model. This shows that KNN's similarity-based method does a good job of capturing the data's structure.

4.2.3 Results of Decision Tree Classifier

Decision Tree is a type of supervised learning algorithm in the sense that users can explain what the input is and what the corresponding output is in the training dataset. Decision Tree provides powerful techniques for both classification and prediction, and works for both categorical and continuous input and output variables (Belachew, 2013). During the Decision Tree analysis process, the data is continuously split based on certain parameters. The goal of using a Decision

Tree is to build up a training model that can predict the class or value of the target variable by learning simple decision rules inferred from the training data.

A Decision Tree can be seen as a flowchart with two entities, namely decision nodes and leaves. Decision nodes include non-leaf nodes and leaf nodes. Each non-leaf node represents a conditional test on one of the parameters and each leaf node represents a class label where predictions or classifications are made for the outcome. Each leaf represents an outcome of the conditional test. Paths from the root to all the leaf nodes give us all the decision rules. Decision Trees are not difficult to represent, construct, and understand, but they have one major drawback, that is, they are very prone to overfitting (Bali, Sarkar, & Lantz, 2017). Because of this reason, Decision Tree models usually do not generalize well. In the current research, we will follow the similar analytics modeling approaches in the literature to build a model based on Decision Trees (Belachew, 2013). We start with loading all the necessary libraries and test data into machine learning with Jupyter Notebook.

We then build a decision tree model with all the variables using training data and finally, we predict the classification value and evaluate the model with the test data. The results are also excellent. As shown in the following decision tree output, the accuracy rate is 94.116 %.

4.2.4 Results on Support Vector Machine (SVM) Classifier

The Support Vector Machine (SVM) Classifier yielded an accuracy of 98.401%. SVM is a supervised machine learning algorithm primarily utilized for classification tasks, although it can also serve for prediction purposes. The fundamental concept of SVM involves identifying the optimal hyperplane based on the training data, which subsequently aids in classifying new data points (Scholkopf & Smola, 2018). Non-linear kernel functions play a pivotal role in SVM, facilitating classification even when linear separation is unattainable in the original feature space. These kernels enable separation in a higher-dimensional transformed feature space, where a hyperplane can effectively delineate the classes (Bali & Sarkar, 2016).

4.2.5 Results on Naïve Bayes

The Naive Bayes classifier achieved an accuracy score of 89.225%, primarily attributed to the imbalance within the dataset. Naive Bayes is a straightforward probabilistic classifier grounded in Bayes' theorem, assuming independence between predictors (Ferrouhi, 2017). Its simplicity lies

in its design aimed at streamlining computational processes (Han, 2006), making it particularly suitable for large datasets (Kumar & Bhardwaj, 2011).

By leveraging class labels from the training dataset, Naive Bayes learns conditional attributes and computes the probability of class values for specific instances. Despite its simplicity, it often predicts the class with the highest probability, rendering it valuable for multi-label classification tasks like those addressed in this study. Furthermore, it is believed by many researchers to outperform more complex classification methods (Kumar & Bhardwaj, 2011).

The resulting accuracy of 89.225 % suggests that the variables used in the model may not be optimal for classification using the Naive Bayes approach in comparable with other used model. Further refinement and feature selection may enhance its performance.

4.2.6 Performance result summary of the standalone classification models

In summary, the baseline classifier models used for the deposit mobilization task were summarized in Table 4. 1 based on their respective performance accuracy achieved. As can be observed in the table KNN Classifier achieved a best result (98.491%), the second result (98.401%) SVM , the third result (94.116%) obtained by Decision Tree, the fourth result LR (92.733%) and the last result with the lowest accuracy of (89.225 %) in comparable with other used model was achieved by Bayes model.

Table 4.1. Performance summary of the standalone models

Model Name	Mode	Accuracy (%)
K-Nearest Neighbors	KNN	98.491
Support Vector Machine	SVM	98.401
Decision Tree Classifier ('cart'	CART	94.116
Logistic Regression	LR	92.733
Gaussian Naive Bayes ('bayes')	Bayes	89.225

In conclusion, the results obtained from the five standalone classification models are high as compared with results achieved in related research areas. This can be attributed due to the nature of the imbalance dataset used for model building and testing. Based on findings from literature by

using the ensemble learning the results obtained from the baseline models above the performance accuracies of the five models might be improved by using the ensemble learning classifier model.

Hence, this study also attempts to use the ensemble learning model in the next section and compares the results obtained with the baseline models.

4.3 Blending Ensemble Model Result

To greatly improve the predictive ability of a Decision Tree, we can use ensemble technique to combine their results. The blending technique indeed does this for us. A RandomForestClassifier model is employed to make predictions on the test dataset 'X_test'. The model generates predictions using 'model.predict(X_test)', storing these predictions in 'y_pred_test'. Subsequently, a list comprehension is utilized to round the predicted values ('y_pred_test') to the nearest integer, ensuring compatibility with the target variable format. These rounded predictions are stored in the 'predictions' list. Following this, the code computes the accuracy of the model's predictions using the 'accuracy_score()' function from scikit-learn. This function compares the predicted values ('predictions') against the actual target labels from the test set ('y_test'), calculating the proportion of correctly classified samples. The results are also very good, showing that the overall accuracy rate is 98.496%.

```
# Initializing the base models
models = get_models()
# Train the blending ensemble
blender = fit_ensemble(models, X_train, X_val, y_train, y_val)
# Make predictions on the test set
yhat = predict_ensemble(models, blender, X_test)
# Evaluate predictions
score = accuracy_score(y_test, yhat)
print('Blending Accuracy: %.3f' % (score * 100))
```

```
Blending Accuracy: 98.496
```

Figure 4.2 Blending Ensemble Model Result

4.4 Random Forest Result on Feature Importance Level

We start our analytical process by loading the necessary libraries into the Jupyter Notebook and then build the random forest training model with all the 13 variables. Using the importance feature function, we can find the importance value for each variable in the Random Forest algorithm as shown below in Figure 4.3. Next, we perform the predictions on the test data and obtain the results as follows by other algorithms.

Using Random Forest feature selection model, the feature importances are displayed in the Figure.4.4. below. This visualization helps in understanding which features are most influential in the model's predictions.

```
# Get numerical feature importances
importances = list(model.feature_importances_)
# List of tuples with variable and importance
feature_importances = [(feature, round(importance, 2)) for feature, importa
# Sort the feature importances by most important first
feature_importances = sorted(feature_importances, key = lambda x: x[1], rev
# Print out the feature and importances
[print('Variable: {:20} Importance: {}'.format(*pair)) for pair in feature_
```

```
Variable: other_debit          Importance: 0.2
Variable: gender               Importance: 0.18
Variable: mobile banking       Importance: 0.13
Variable: mobile_credit        Importance: 0.08
Variable: mobile_debit         Importance: 0.06
Variable: card_debit           Importance: 0.06
Variable: marital_status       Importance: 0.06
Variable: age                  Importance: 0.05
Variable: total_deposit        Importance: 0.05
Variable: card_credit          Importance: 0.04
Variable: job                  Importance: 0.04
Variable: other_credit         Importance: 0.03
Variable: atm_card             Importance: 0.03
```

Figure 4.3 Feature Importance Level result

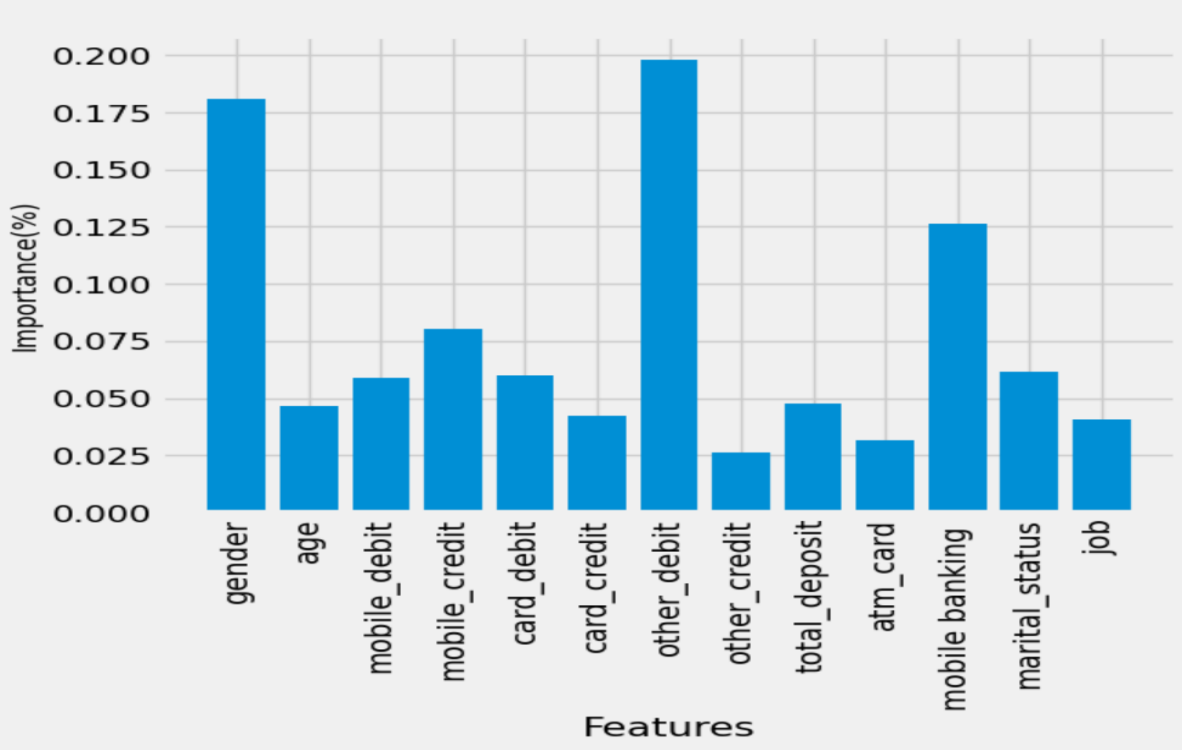

Figure 4.4. Feature importance for deposit mobilization

To summarize the result, the processes involved in building the six-prediction model Random Forest, Cart, SVM, KNN, Naïve Bayes and Logistic Regression Machine Learning Algorithms Techniques; as well as the performance evaluation procedures were discussed mean while the cross table was presented to visualize the model result and the confusion matrix was presented for the accuracy, also precision was calculated.

Responding to the research, question 1. "Which attributes are more significant to predict customers deposit at CBO?"

Using the results obtained in Figure 4.4, the feature importance derived from Random Forest Classifier model were used. The output showed that the 'other_debit', 'Gender' and 'mobile banking' variables were identified as the most important features for the model in predicting the target variable "deposit Mobilization" with importance score of 20%, 18% and 13%, respectively. Similarly, 'mobile_credit', 'mobile_debit', 'card_debit', and 'marital_status' features also demonstrate considerable importance, each contributing around 8% and 6% to the model's predictive capability. features like 'job', 'age', 'total_deposit', 'card_credit', 'other_credit', and

'atm_card' exhibit varying degrees of importance, even if with relatively lower contributions in predicting the target variable.

The importance features selection model showed that the feature 'atm_card' and ‘other_credit’ indicated no importance almost in the model's predictive process, with an importance score of 3%, suggesting it contributes none to predicting the target variable compared to other attributes in the dataset.

In addition, in relation to research question two, “Which classification algorithm is more suitable for constructing CBO customer deposit classification model?” ensemble model was emerged the most suitable for customer deposit classification model Accuracy 98.496 percentage. Concerning question three Is the ensemble learning model performance accuracy better than base learning models in determining the level of accuracy of deposit mobilization efforts? Yes. In relation to research, question four, “How can we forecast the performance based on the given attribute that influence deposit mobilization performance?” by referring Table 4.1 above, Ensemble model by 98.496%, CART by 94.116%, KNN by 98.491%, SVM by 98.401%, LR by 92.733%and Naïve Bayes by 89.225% were modeled to find out how machine learning can be utilized in making decision whether to e-banking service was contributing for deposit mobilization.

CHAPTER FIVE

5 CONCLUSIONS AND RECOMMENDETIONS

5.1 Overview

This chapter summarizes the entire findings of the study. It is divided into two main sections conclusion, recommendation it provides the conclusion and recommendations the future works.

5.2 Conclusions

The necessary data for the experiment was obtained from the Cooperative Bank of Oromia on a scattered excel sheets, which totally amounted to a dataset of 82593. The necessary preprocessing activities were applied on the dataset after which 82593 data was prepared for the experimentation. Classification model building was experimented with Random Forest, CART, KNN, SVM, Naïve Bayes and Logistic Regression algorithm. In addition, the tools were used to simulate all the experiment is Python programing. The confusion matrix was used and then accuracy was calculated. The ensemble model scores remarkable result accuracy of 98.496% when it comes to correctly classifying instances. The three models (KNN, and SVM) very related results when it comes to classifying instances, SVM with accuracy of 98.401%, and KNN with accuracy of 98.491% on balanced and imbalanced dataset respectively.

After understanding our data and exploring the several important models to predict if the client will subscribe a term deposit, we have arrived at the conclusion that all these models have done a good job and there is one model that is all the best among them which is Random Forest. One limitation of this research is that we do not improve the models by using some techniques such as attribute selection. This may suggest that more advanced studies on these six machine-learning algorithms are necessary in the future.

However, I would say this thesis as a brief study about six important and mostly used classification algorithms: K-Nearest Neighbors, Classification and Regression Tree and Naïve Bayes, Random Forest and Logistic Regression. Each example scenarios have been implemented with Python code using an open source data science tool called Anaconda powered by Python 3.0 Jupyter Notebook

with inbuilt scientific modules such as NumPy and Matplotlib and scikit-learn library. The results have been clearly given after each implementation along with the detailed explanation of the python code.

5.3 Recommendations

The performance of the ensemble techniques by combining different model (KNN, SVM, CART, Naïve Bayes and Logistic Regression) is dependent on the quality and representativeness of the input data. Furthermore, additional machine learning techniques that could potentially increase the accuracy of deposit mobilization estimates were not included in the study. Future research could compare the performance of alternative approaches to the ensemble techniques model. In addition, use another model and large data for the development of classification model. By using the dataset of this study and some other additional datasets, investigate the factor that is used to affect the deposit classification that brings higher deposit. Based on the findings and conclusion of the study, here are recommendations for commercial banks:

- Use Ensemble Learning: Make use of the higher accuracy of ensemble learning when predicting deposit mobilization as opposed to individual models.
- Emphasis on Crucial Attributes: Give top priority to improving attributes that have been found to have a major impact on deposit mobilization, such as "other_debit," "gender," and "mobile banking." Monitor and Adapt:
- To ensure sustained forecasting accuracy, continuously monitor and update the ensemble learning model in response to shifting consumer behaviours and market conditions.
- Improve Customer Segmentation: Apply model data to improve customer segmentation tactics, allowing for targeted advertising and service improvements catered to various segments according to their patterns of deposit activity.

References

Abiodun, A. O.-A. (2021). Deposit determinants and domestic currency deposits in Nigeria (2000–2018).

Ainan, U., & Nur-E-Arefin, H. (2022). Prediction of bank performance using machine learning and genetic algorithm hybrid models. *International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE)*.

Anastasiou, D. a. (2020). Bank deposits flows and textual sentiment: when an ECB President's speech is not just a speech. *MPRA Working Paper No. 99729, available at: https:// mpra.ub.uni-muenchen.de/99729/*.

Ayodele, T. (2010). Types of machine learning algorithms. *New Adv. Mach. Learn., vol. 3*, 19-48.

Babbie-Earl, R. (1998). The-Basics-of-Social-Research. 1998.

Bali, R., & Sarkar, D. (2016). R machine learning by example. *Packt Publishing.*

Bali, R., Sarkar, D., & Lantz, B. (2017). R: Unleash machine learning techniques: . *Smarter data analytics. Packt Publishing.*

Belachew, R. (2013). Application of Data Mining Techniques for CustomersSegmentation and Prediction: The Case of Buusaa Gonofa Microfinance Institution.

Bland, M. (2015). An Introduction to Medical Statistics. *Oxford University Press.*

CBO. (2022). *Https:///about/#history, "About Us – Cooperative Bank of Oromia", [Online].* Retrieved from coopbankoromia.com.et: https://coopbankoromia.com.et/about/#history

Chen, K., Hu, Y.-H., & Hsieh, Y.-C. (2014). Predicting customer churn from valuable B2B customers in the logistics industry: a case study. *IseB 13:475–494. Doi:10.1007/s10257-014-0264-1*.

Coopbank. (2021). *A. Report, "Coopbank Annual Report," 2021.* Coopbank Annual Report.

David, A. (2012). Introduction to Predictive Modeling with Examples,. *SAS Global Forum*.

Elsalamony, H. A. (2014). Bank direct marketing analysis of data mining techniques. *International Journal of Computers and Applications, 85,*, 12-22.

Ferrouhi, E. (2017). Determinants of Bank Deposits in Morocc. *Munich Personal RePEc Archive,*, 23-26.

Fischer, T., & Krauss, C. (2018). *European Journal of Operational Research, 270(2),*, 654–669.

Gafrej, O. (2023). Predicting customer deposits with machine learning algorithms: evidence from Tunisia. *https://www.emerald.com/insight/0307-4358.htm*.

Garo, G. (2015). P. Fulfillment and E. Masters, “Determinants of Deposit Mobilization and Related Costs of Commercial Banks in Ethiopia . *By A Research Project Paper Submitted to the Department of Management College of Business and Economics*.

Ghaneei, H., & Mohammad, S. (2016). Developing a prediction model for customer churn from electronic banking services using data mining.

Glas, F. (2015). Machine-Learning Techniques for Customer Recommendations. *Department of Computer Science Faculty of Engineering LTH*.

Han, J. W. (2006). Data mining concepts and techniques . *Morgan Kaufmann Publishers.(2nd ed.).*

Harrington, P. (2010). Machine Learning in Action. . *Manning Publications Co. ISBN 9781617290183*.

Jiangtao Ren, S. D. (2009). Naive Bayes Classification of Uncertain Data. *Ninth IEEE International Conference on Data Mining.*

Jorma, L. E. (1996). Classification with Learning k-Nearest Neighbors”,. *Helsinki University of Technology Laboratory of Computer and Information Science IEEE.*

Kagan, J. (2020). Time Deposit. *Investopedia*.

Kecerdasan. (2007). A Knowledge Discovery Approach. *New York*.

Khalid, M. F. (2015). Comparative Analysis of Support Vector Machine: Employing Various Optimization Algorithms”. *14th International Conference on Information Technology,IEEE*.

Kumar, A., Sharma, M., & Mahavi, M. (2021). Machine learning (ML) technologies for digital credit scoring in rural finance: a literature review. *Risks*.

Kumar, D., & Bhardwaj, D. (2011). Rise of Data Mining: Current and Future Application Areas. *International Journal of Computer Science Issues*, 256-260.

Kumar, M. (2010). Comparative evaluation of critical factors in delivering service quality of banks. *International Journal of Quality &Reliability Management,* , 351-377.

Little, R. J., & Rubin, D. B. (2002). Statistical analysis with missing data. *Wiley.*

Muslim, M., Darsil, Y., Alamsyah, A., & Mustaqim, T. (2020). Bank prediction for prospective long-term deposit investors using machine learning light GBM and SMOTE. *Journal of physics: Conference Series, Vol. 1918 No. 1*, 1-6.

Nicholas, G., & Scott, F. (2021). Logistic Regression through the Veil of Imprecise Data.

Orlova, E. (2020). Decision-making techniques for credit resource management using machine learning and optimization. *Information 11(3):144 Google school*.

Ozgur, O. K., & Ozbugday, F. (2021). Machine learning approach to drivers of bank lending: evidence from an emerging economy. *inanc Innovation https://doi.org/10.1186/s40854-021-00237-1*.

Palaniappan, S., Mustapha, Foozy, C. M., & Atan, R. (2017). Profiling using Classification Approach for Bank Telemarketing. *International Journal on Informatics Visualization*.

Patwary, M., Akter, S., Bin Alam, M., & Rezaul Karim, A. (2021). "Bank deposit prediction using ensemble learning. *Artificial Intelligence Evolution,*, 42-51.

Saxena, R. (2016). Introduction to k Nearest Neighbor Classification and Condensed Nearest Neighbour Data Reduction" Data Science, Machine. *Machine Learning journal monthly blog*.

Scholkopf, B., & Smola, A. J. (2018). Learning with kernels: Support vector machines, regularization, optimization, and beyond. *The MIT Press*.

Stamp, M. (2018). A Survey of Machine Learning Algorithms and Their Application in Information Security: An Artificial Intelligence Approach," . *San Jose State University, San Jose, California*.

Steinbach, P. T. (2006). Introduction to Data Mining Instructor ' s Solution Manual,".

Sunita, Y., & Chavan, S. (2013). predictive model Insurance Industry using R tool. *Journal of Research in Commerce & Management*.

Vaidya, A. (2017). Predictive and Probabilistic Approach Using Logistic Regression. *Application To Prediction of Loan Approval" 8th ICCCNT 2017* .

Verma, J. (2022). Application of machine learning for fraud detection A decision support system in the insurance sector. In Big Data analitics. *Emerald Publishing Limited: Bingley, UK, 2022; pp. 251–262*.

Yuguang Huang, L. L. (2011). Naïve Bayes Classification Algorithm Based on Small Sample Set. *Beijing University of Posts and Telecommunications, Proceedings of IEEE*.

ZenTut. (2013). Data Mining Techniques. . *Data mining. Retrieved April 09*.

Appendix

Blending Ensemble for Classification Problems

```
# Blending Ensemble for Classification Problems
from numpy import hstack
from collections import Counter
import pandas as pd
import numpy as np
from numpy import sort
from imblearn.over_sampling import SMOTE
from imblearn.over_sampling import SVMSMOTE
from imblearn.over_sampling import BorderlineSMOTE
from sklearn.feature_selection import SelectFromModel
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.metrics import precision_score,recall_score,accuracy_score, confusion_matrix
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
from sklearn.naive_bayes import GaussianNB
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OrdinalEncoder
from sklearn.preprocessing import OneHotEncoder
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib import pyplot
```