VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY

UNIVERSITY OF ECONOMICS AND LAWS

FINAL PROJECT REPORT

MACHINE LEARNING IN BUSINESS ANALYTICS

BUILDING A SOFTWARE THAT INTEGRATES MACHINE LEARNING MODELS TO PREDICT CUSTOMER CHURN IN BANKS

LECTURER: Ph.D TRAN DUY THANH

COURSE ID: 232MI4304

MEMBERS OF GROUP 4

No.	Full Name	Student ID
1	Nguyễn Trần Thanh Huyền	K224111450
2	Phạm Tuyết Nhung	K224111460
3	Vũ Quỳnh Như	K224111461
4	Lê Nguyễn Minh Thảo	K224111462

HCM city, May 2024.

LECTURER'S ASSESSMENT

……

……

……

……

……

……

……

……

……

……

……

……

……

……

……

……

GROUP ASSESSMENT

No.	Full Name	Student ID	Contribution
1	Nguyền Trần Thanh Huyền	K224111450	100%
2	Phạm Tuyết Nhung	K224111460	100%
3	Vũ Quỳnh Như	K224111461	100%
4	Lê Nguyễn Minh Thảo	K224111462	100%

ACKNOWLEDGMENTS

This study was partially supported by Mr. Tran Duy Thanh. Words cannot express our gratitude to you. Without the two experts' assistance and dedicated involvement in every step throughout the process, this paper couldn't have been fully accomplished. Your insightful pieces of advice and feedback pushed us to sharpen our thinking and brought this work to a higher level. We want to express our deepest appreciation to you for supporting us during the past time and steering us in the right direction whenever we need it.

Most importantly, we had lots of pleasure working/collaborating with each other this semester.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

CHAP 1: INTRODUCTION

1.1 Reason to choose this topic

Nowadays, the phenomenon of customers leaving banks is becoming a significant issue for the financial industry. Increasing competition, along with the ease of switching between banking services and changes in customer needs and behaviors, have made customer retention a major challenge for the banks.

In this context, building a machine learning model to predict the likelihood of customers leaving the bank is essential. This model can help banks anticipate which customers are at high risk of leaving, thereby enabling timely interventions to retain them.

The purpose of developing this integrated machine learning model software is not only to predict but also to provide banks with an effective tool to optimize customer relationship management strategies. This software will help banks identify potential customers for enhanced care and tailor appropriate marketing strategies and services, thereby increasing business efficiency and minimizing the risk of customer loss.

For these reasons, the team has chosen the topic **"Building a software that integrates machine learning models to predict customer churn in banks."** This is not just a research project but also brings many benefits in improving management and business development in the banking industry.

1.2 Objectives

This research article aims to predict the likelihood of customers leaving the bank using machine learning methods through the following methods: XGBoost. This prediction

result can provide recommendations for bank managers in retaining customers using banking services.

Through that, we build and apply this machine learning model into practice in the bank's business operations, to help identify and take care of customers at risk of leaving, and improve the business performance of the bank.

1.3 Data Selection Criteria

To construct an efficient software that integrates machine learning models capable of predicting customer churn in banks, careful selection of data is crucial. The method of data collection, chosen selectively and accurately, is a determining factor in ensuring that the model is trained on reliable data and accurately represents a comprehensive view of customer behavior.

The data chosen by the team is secondary data. Searching for and selecting suitable datasets requires meticulous consideration. The criterion of reputable data sources is crucial, ensuring that the data collected from sources have been verified and recognized for their accuracy. Therefore, the team has selected reputable platforms such as Kaggle, Housing, etc., to reference datasets.

Another criterion is the completeness and diversity of the data. For a model predicting customer churn, having a rich and diverse dataset with variables such as transaction history, personal information, and service usage behavior will help the model better understand the factors influencing customer decisions.

The timeliness of the data is also an important factor not to be overlooked. With continuous market fluctuations and customer behavior, using the latest data helps the model

respond to the latest changes and trends, thereby increasing the accuracy and reliability of predictions.

CHAP 2: THEORETICAL FRAMEWORK

2.1 Introduction to Machine Learning

Machine Learning is the science of developing algorithms and statistical models that computer systems use to perform tasks based on patterns and inference without specific instructions. According to Dr.Chitra and B.Subahini (2013), machine learning has the main purpose of training computers to automatically "learn" without human intervention or assistance to perform and adjust actions. Computer systems use machine learning algorithms to process large volumes of historical data and identify data patterns. This allows them to predict results more accurately from the same set of input data.

In today's digital age, machine learning plays an undeniable role in analytics and predictions. Its ability to automate the extraction of information from data and create predictions based on that information cannot be ignored. One of the most important applications of machine learning is its ability to process big data. With large amounts of data being generated every day, machine learning helps us analyze and find patterns from these data sets effectively. Not only does it help predict future trends based on historical data, machine learning is also used to optimize workflows, detect fraud and unusual activity, provide personalized services, and assist in making smart business decisions. With such diverse and important roles, machine learning has become an indispensable tool in the modern technology world. Today, the development and application of machine learning is being widely applied in almost every area of life. Additionally, more and more industries are building on machine learning models that are capable of analyzing larger and more complex data and providing faster, more accurate results at scale.

In the world, many famous technology corporations have used the foundation of artificial intelligence and machine learning to launch many applications such as self-driving cars from Google and Tesla, Facebook's system for self-tagging faces in photos, Apple's Siri virtual assistant, Amazon's product suggestion system, Netflix's movie suggestion system, Gmail's spam email filtering system, credit card fraud detection in banks...

2.2 Analyze customer churn in banking

Customer churn is a situation in which a customer has started using a business's product or service, but for one reason or another, stops completely and switches to another competitor. In business, when customers are not satisfied with the services and products that a business provides, they will stop connecting and cooperating with the business.

Like other industries, the banking and finance industry also needs to predict and determine customer lifetime value. Therefore, assessing which customers will stay after a transaction and how they contribute to the company's future revenue are issues that businesses need to pay attention to. Thanks to data science, banks can screen and classify potential customer groups as well as practical future values through analysis and prediction, thereby helping to determine the right customers. customers as well as contribute to the growth and profitability of the bank (Jain, H., Khunteta, A., and Srivastava, S., 2020).

In short, customers' leaving or staying is the deciding factor for a bank's development and profits. Finding a new customer is much more expensive than retaining an existing one. Research by Roberts (2000) and a number of other studies has shown that the cost of finding new customers is much higher than the cost of retaining old customers. A 5% increase in customer retention rate can increase bank profits by 85% (Reichheld and Sasser, 1990).

Therefore, the need to analyze customer churn rates is increasing. In particular, there is a need for predictive models built on methods in the field of data science. If banks can predict customer churn rates, marketing campaigns to retain customers will be improved, thereby significantly saving customer acquisition costs and bringing more efficiency to businesses.

2.3 Methods and Models for Prediction

Based on many reference documents, we decided to choose 4 algorithms including: Logistic Regression, Decision Tree, Random Forest, XGBoost combined with RandomizedSearchCV. To better understand each algorithm, we explores the definitions, advantages and disadvantages of each algorithm.

2.3.1 Logistic Regression

Logistic Regression is a major part of machine learning algorithms. It is a powerful and effective tool in understanding relationships and predicting probabilities, in this topic it is predicting the likelihood (probability) that customers will leave the bank.

Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, True or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

Equation:

$$log\left[\frac{y}{1-y}\right] = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n$$

2.3.2 Decision Tree

Decision Tree algorithm is a powerful tool for classification and regression tasks due to their intuitive nature and ability to model complex relationships. Furthermore, it is a simple, accessible and transparent tool.

Decision Tree algorithm works in simpler steps:

- Step 1: Starting at the Root: The algorithm begins at the top, called the “root node,” representing the entire dataset.
- Step 2: Asking the Best Questions: It looks for the most important feature or question that splits the data into the most distinct groups. This is like asking a question at a fork in the tree.
- Step 3: Branching Out: Based on the answer to that question, it divides the data into smaller subsets, creating new branches. Each branch represents a possible route through the tree.
- Step 4: Repeating the Process: The algorithm continues asking questions and splitting the data at each branch until it reaches the final “leaf nodes,” representing the predicted outcomes or classifications.

2.3.3 Random Forest

Random Forest Algorithm offer a powerful solution, particularly for complex classification and regression problems where accuracy is paramount. They address many of the limitations of Decision Tree, such as overfitting. It is also an advanced tool based on Decision Tree.

Steps involved in Random Forest Algorithm:

- Step 1: In the Random Forest model, a subset of data points and a subset of features is selected for constructing each decision tree. Simply put, n random records and m features are taken from the data set having k number of records.
- Step 2: Individual decision trees are constructed for each sample.
- Step 3: Each decision tree will generate an output.
- Step 4: Final output is considered based on Majority Voting or Averaging for Classification and regression, respectively.

2.3.4 XGBoost

Gradient Boost is a supervised learning method in which weak models (e.g., small decision trees) are trained sequentially, with each new model attempting to correct the mistakes of the previous model. XGBoost is an improved version of Gradient Boosting, designed to maximize computing speed and performance. Here we decided to use RandomizedSearchCV to find the optimal parameters for XGBoost.

Key features of XGBoost Algorithm include its ability to handle complex relationships in data, regularization techniques to prevent overfitting and incorporation of parallel processing for efficient computation. XGBoost is widely used in various domains due to its high predictive performance and versatility across different datasets.

In boosting, the trees are built sequentially such that each subsequent tree aims to reduce the errors of the previous tree. Each tree learns from its predecessors and updates the residual errors. Hence, the tree that grows next in the sequence will learn from an updated version of the residuals.

2.4 Evaluation Indicators

Evaluating machine learning models helps us find out which model is most effective and appropriate. Common model evaluation indicators commonly used are:

- Accuracy is defined as the percentage of correct predictions for test data. It can be easily calculated by dividing the number of correct predictions by the total number of predictions.

$$accuracy = \frac{correct\ predictions}{all\ predictions}$$

- Precision is defined as the fraction of relevant examples (true positives) among all examples predicted to belong to a given class.

$$precision = \frac{True\ Positives\ (TP)}{True\ Positives\ (TP) + False\ Positive\ (FP)}$$

- Recall is defined as the fraction of examples predicted to belong to a class relative to all examples that actually belong to that class.

$$recall = \frac{True\ Positives\ (TP)}{True\ Positives\ (TP) + False\ Negatives\ (NG)}$$

- F1-Score: combines Precision and Recall

$$F_{1-score} = \frac{2 * Precision * Recall}{Precision + Recall}$$

In there:

- True Positive (TP): the object is in the Positive class, the model classifies the object in the Positive class (correct prediction)
- True Negative (TN): the object is in the Negative class, the model classifies the object in the Negative class (correct prediction)

- False Positive (FP): the object is in the Negative class, the model classifies the object in the Positive class (wrong prediction) – Type I Error
- False Negative (FN): the object is in the Positive class, the model classifies the object in the Negative class (wrong prediction) – Type II Error

According to Kulkani et al (2020), there are the following 2 tables:

	Predicted to “churn”	Predicted to “not churn”
Actually “churn”	True Positive	False Negative
Actually “not churn”	False Positive	True Negative

Table 2.1 Unstandardized confusion matrix

This table presents the unnormalized confusion matrix, illustrating the number of correct and incorrect predictions made by the model based on actual data.

The matrix consists of four cells: True Positive (TP), False Negative (FN), False Positive (FP), and True Negative (TN).

- True Positive is the number of correctly predicted "churn" cases when the actual outcome is also "churn".
- False Negative is the number of incorrectly predicted "not churn" cases when the actual outcome is "churn".
- False Positive is the number of incorrectly predicted "churn" cases when the actual outcome is "not churn".
- True Negative is the number of correctly predicted "not churn" cases when the actual outcome is also "not churn".

This unnormalized confusion matrix provides a general overview of the model's performance in correctly and incorrectly classifying cases.

	Predicted to “churn”	Predicted to “not churn”
Actually “churn”	TPR = TP/(TP+FN)	FNR = FN/(TP+FN)

Actually “not churn”	FPR = FP/(FP+TN)	TNR = TN/(FP+TN)

Table 2.2 Normalized confusion matrix

This table presents the normalized confusion matrix, which shows the normalized values for the performance metrics.

In this matrix, True Positive Rate (TPR) and False Negative Rate (FNR) are used to evaluate the model's performance on "churn" cases.

- TPR, also known as recall or sensitivity, is calculated as TP/(TP+FN), while FNR is calculated as FN/(TP+FN).
- For "not churn" cases, False Positive Rate (FPR) and True Negative Rate (TNR) are used.
- FPR is calculated as FP/(FP+TN) and TNR, also known as specificity, is calculated as TN/(FP+TN).

This normalized confusion matrix helps to assess the model's ability to correctly classify cases through the normalized rates.

2.5 Proposal

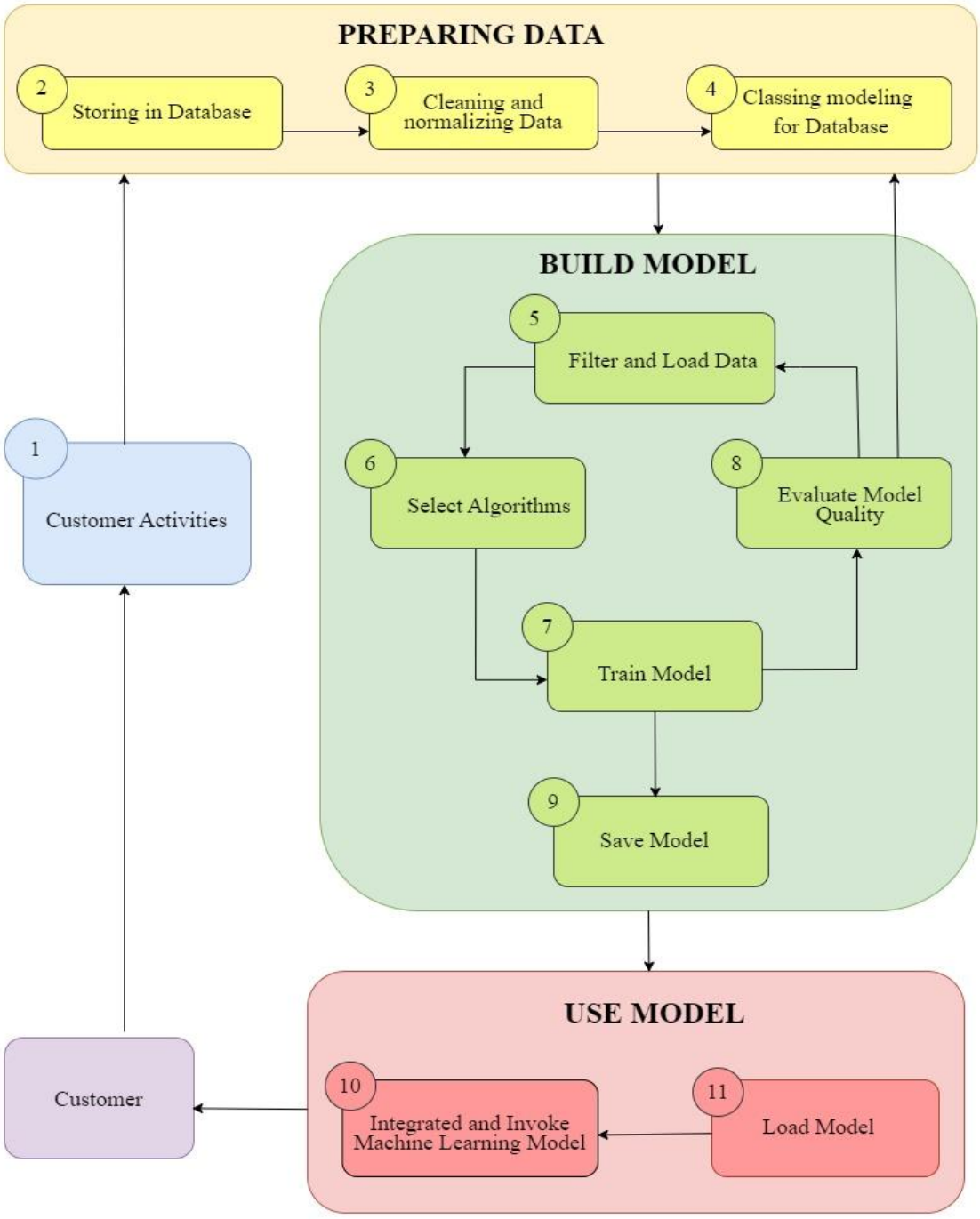

Figure 2.1 Proposal

PREPARING DATA

- **Step 1: Customer Activities**

Customers have activities on the system related to:

  - Transactions: Number of transactions, type of transactions, transaction value, transaction time, etc.
  - Accounts: Account number, account type, account balance, account opening date, etc.
  - Interactions: Number of logins, number of customer service contacts, customer service feedback, etc.
  - Personal Information: Age, gender, place of residence, income, etc.

- **Step 2: Storing in Database**

Collect data from reputable platforms, ensuring that the selected data is accurate and highly realistic. Proceed to store the data in a MySQL database by creating data tables to store customer information and account details.

- **Step 3: Cleaning and normalizing Data**

Data preprocessing and normalization is an important step in any machine learning project. This process involves multiple steps to clean, transform, and normalize data before using it for machine learning models.

  - Step 1: Remove unnecessary columns. In this data set, remove the "customer_id" column because it does not affect the prediction process. (Pandas library: pd.drop)
  - Step 2: Remove rows with null values (pd.dropna)

- Step 3: Convert data from categories to numbers (One - hot Encoding)
- Step 4: Standardize variables into a standard format (StandardScaler)
- Step 5: Data balancing (SMOTE)
- Step 6: Split the dataset into 2 files: training file and test file (sklearn library: train_test_split)

Use libraries: Pandas, SMOTE, StandardScaler, sklearn.

- **Step 4: Classing modeling for Database**

Describe the database structure:

Account Table: Systematized data includes columns for employee information.

- user_name: Employee's username
- pass_word: Employee's account password
- employee_id: Employee ID
- employee_name: Employee's name
- phone: Employee's phone number
- email: Employee's email

Data Table: Systematized data includes detailed information about customers.

- customer_id: Account Number
- credit_score: Credit Score
- country: Country of Residence
- gender: Sex
- age: Age of customer
- tenure: From how many years he/she is having bank account in ABC Bank

- balance: Account balance
- products_number: Number of Product from bank
- credit_card: Does this customer have a credit card? (1 - Yes, 0 - No)
- active_member: Is he/she an active member of the bank? (1 - Yes, 0 - No)
- estimated_salary: Salary of Account holder
- churn: 1 if the client has left the bank during some period or 0 if he/she has not

BUILD MODEL

- **Step 5: Filter and Load Data**

Attributes/variables affecting the model include:

- credit_score
- country
- gender
- age
- tenure
- balance
- products_number
- credit_card
- active_member
- estimated_salary
- churn

Use Python libraries to load data from the MySQL database.

- **Step 6: Select Algorithms**

The team uses machine learning algorithms suitable for predicting customer churn rates, including:

  - Logistic Regression
  - Decision Tree
  - Random Forest
  - XGBoost + RandomizedCV

- **Step 7: Train Model**

Split the data into training set and test set to ensure accurate model evaluation. Each method will be combined with a machine learning algorithm to train and predict based on the pre-split test set. Each iteration will provide training time, prediction time, scores, and accuracy for each specific model. Train/test model ratio is 80:20.

- **Step 8: Evaluate Model Quality**

The team uses common metrics to evaluate model quality such as:

  - Precision: The ratio of correctly predicted positive observations to the total predicted positives.
  - Recall: The ratio of correctly predicted positive observations to all observations in the actual class.
  - F1-score: The weighted average of precision and recall.
  - Accuracy score: The percentage of correct predictions for the test data.

Compare the prediction results on the test set with actual values to evaluate the model. Based on the evaluated metrics, the team will consider and select the model with the highest precision, recall, F1-score, and accuracy score.

- **Step 9: Save Model**

After executing the model, to help businesses easily monitor and make decisions, the experimental results are further visualized on intelligent reports.

Save the trained model to disk for future use. Tools like joblib or pickle are commonly used to store machine learning models.

Saving the model saves time and avoids the need to retrain the model from scratch during each deployment, facilitating easy backup and recovery, thus minimizing data loss.

USE MODEL

- **Step 10: Load Model**

Customers load the saved model from disk and integrate it into the system. This allows customers to reuse the model without retraining from scratch.

- **Step 11: Integrated and Invoke Machine Learning Model**

Use the loaded model to predict customer churn rates on new data. Provide new data to the model and receive prediction results, enabling necessary actions to retain customers.

CHAP 3: RESEARCH METHODOLOGY

3.1 Data Collection

Data link: dataset

Secondary data was collected by the team from the "Bank Customer Data for Predicting Customer Churn" data set of ABC Multistate Bank on the Kaggle platform. Owned by Google, Kaggle is a web platform that provides a variety of datasets for research.

This data set includes detailed information that can all influence a customer's decision to leave the bank. The data set is updated in 2022. The update time is close to the current year so the accuracy of the data can be guaranteed.

Data set details include columns:

- customer_id: Account Number
- credit_score: Credit Score
- country: Country of Residence
- gender: Sex
- age: Age of customer
- tenure: From how many years he/she is having bank account in ABC Bank
- balance: Account balance
- products_number: Number of Product from bank
- credit_card: Does this customer have a credit card? (1 - Yes, 0 - No)
- active_member: Is he/she an active member of the bank? (1 - Yes, 0 - No)
- estimated_salary: Salary of Account holder

- churn: 1 if the client has left the bank during some period or 0 if he/she has not

3.2 Preprocess/normalize data

Data preprocessing and normalization is an important step in any machine learning project. This process involves multiple steps to clean, transform, and normalize data before using it for machine learning models.

- Step 1: Remove unnecessary columns. In this data set, remove the "customer_id" column because it does not affect the prediction process. (Pandas library: pd.drop)
- Step 2: Remove rows with null values (pd.dropna)
- Step 3: Convert data from categories to numbers (One - hot Encoding)
- Step 4: Standardize variables into a standard format (StandardScaler)
- Step 5: Data balancing (SMOTE)
- Step 6: Split the dataset into 2 files: training file and test file (sklearn library: train_test_split)

3.3 Evaluation of Machine Learning Models

We evaluate 4 algorithms using the unstandardized confusion matrix table and the evaluation indicators mentioned above.

- **XGBoost**

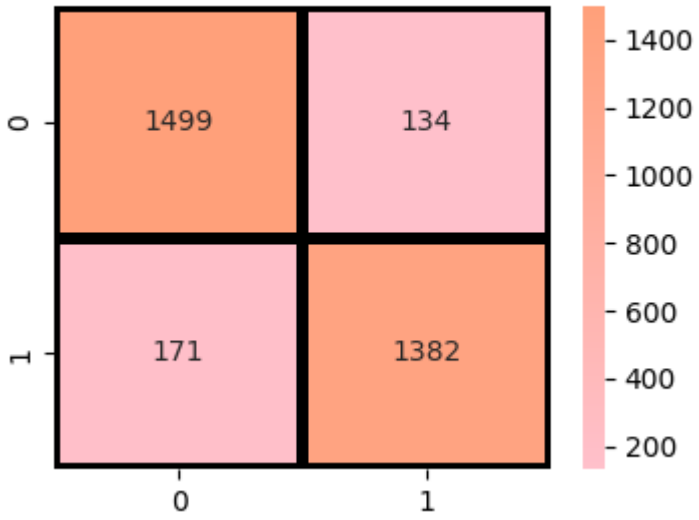

Figure 3.1 Unstandardized confusion matrix of XGBoost

From the matrix:

- There are 1499 customers who are actually staying customers and the prediction is correct.
- There are 134 customers who are actually staying customers but are predicted to churn.
- There are 171 customers who actually left but are predicted to stay.
- There are 1382 customers leaving banking services and the prediction is correct.

- **Logistic Regression**

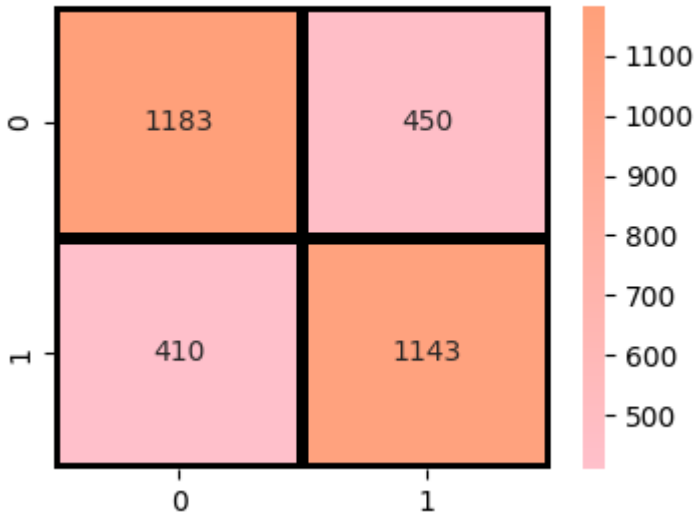

Figure 3.2 Unstandardized confusion matrix of Logistic Regression

From the matrix:

- There are 1183 customers who are actually staying customers and the prediction is correct.
- There are 450 customers who are actually staying customers but are predicted to churn.
- There are 410 customers who actually left but are predicted to stay.
- There are 1143 customers leaving banking services and the prediction is correct.

- **Random Forest**

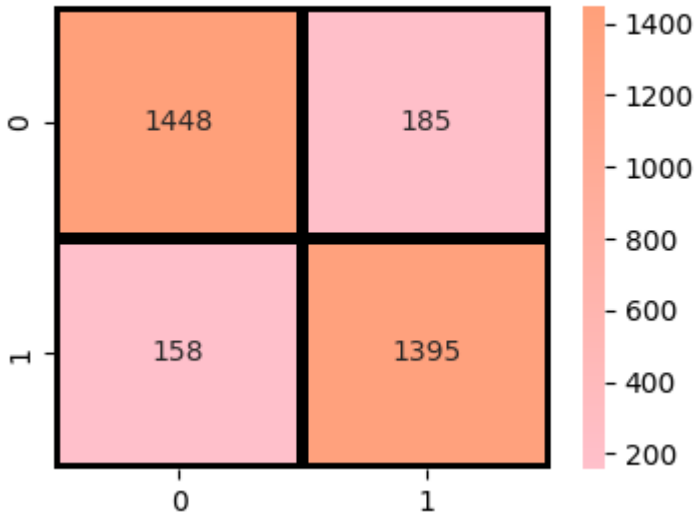

Figure 3.3 Unstandardized confusion matrix of Random Forest

From the matrix:

- There are 1448 customers who are actually staying customers and the prediction is correct.
- There are 185 customers who are actually staying customers but are predicted to churn.
- There are 158 customers who actually left but are predicted to stay.
- There are 1395 customers leaving banking services and the prediction is correct.

- **Decision Tree**

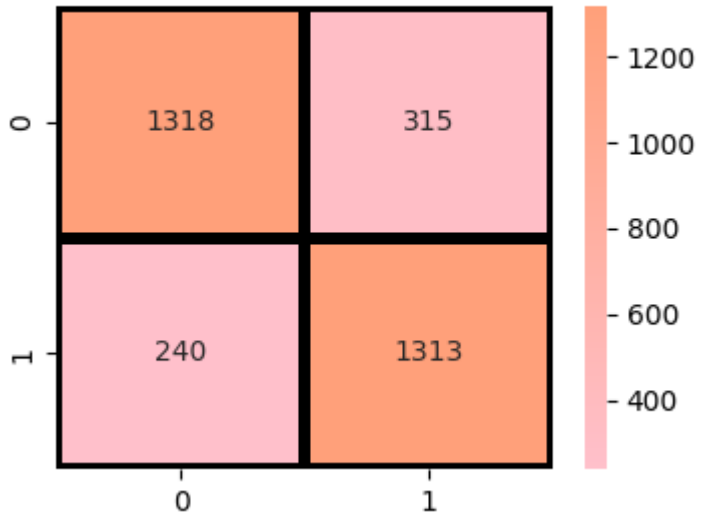

Figure 3.4 Unstandardized confusion matrix of Decision Tree

From the matrix:

- There are 1318 customers who are actually staying customers and the prediction is correct.
- There are 315 customers who are actually staying customers but are predicted to churn.
- There are 240 customers who actually left but are predicted to stay.
- There are 1313 customers leaving banking services and the prediction is correct.

Evaluation algorithms:

Indicators	XGBoost	Logistic Regression	Random Forest	Decision Tree
accuracy	**0.904**	0.730	0.892	0.826
precision	**0.912**	0.718	0.882	0.807
recall	**0.900**	0.736	0.898	0.846
f1-score	**0.901**	0.727	0.891	0.826

Table 3.1 Evaluation algorithms

The table provides a comparative analysis of the performance metrics for four different machine learning algorithms: XGBoost, Logistic Regression, Random Forest, and Decision Tree. The performance metrics considered are accuracy, precision, recall, and F1-score. These metrics are critical for evaluating the effectiveness of classification algorithms.

- Accuracy: Measures the proportion of correctly classified instances among all instances. XGBoost has the highest accuracy (0.904), indicating it correctly classifies a high percentage of instances.
- Precision: Measures the proportion of true positive instances among the instances classified as positive. XGBoost again leads with a precision of 0.912, meaning it has the highest rate of correct positive predictions.
- Recall: Measures the proportion of true positive instances among all actual positive instances. XGBoost has a recall of 0.900, showing it effectively identifies most of the positive instances.
- F1-score: The harmonic mean of precision and recall, providing a balance between the two. XGBoost achieves the highest F1-score (0.901), indicating a good balance between precision and recall.

It can be seen in the table above that all indexes of XGBoost combined with RandomizedSearchCV achieve the most optimal results compared to the other 3 algorithms. Therefore XGBoost combined with RandomizedSearchCV will be the algorithm used to perform predictions in our software.

3.4 Flowchart

The diagram shows the information processing flow of the software

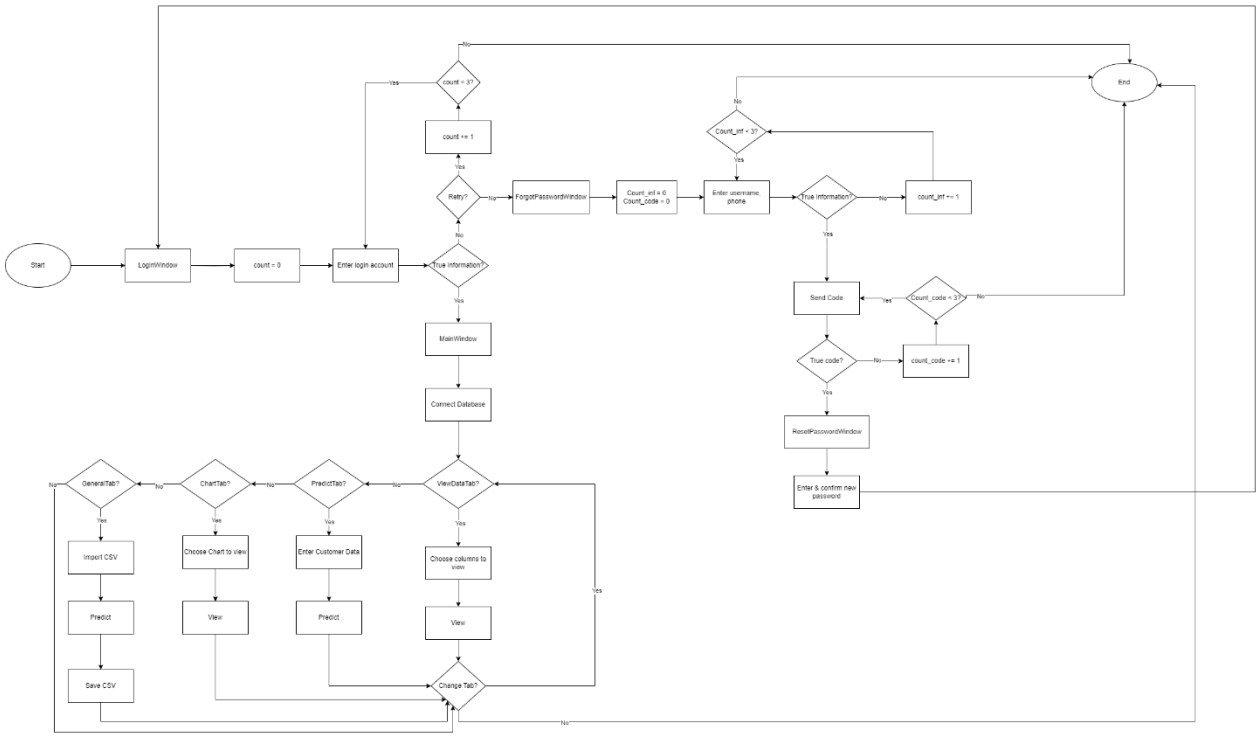

Figure 3.5 Flowchart of information process flow

First, when launching the software, the user will start with the LoginWindow screen, here 2 situations occur:

- If the login information is correct, the user goes to the next screen MainWindow.
- If you log in incorrectly:
    - Log in incorrectly 3 times, the interface automatically closes.
    - Forgot password: go to ForgotYourPassword screen. Enter username and phone number, a code will be sent to the user's phone number. If you confirm the correct code, you will go to the ResetPassword screen, enter a new password and return to the LoginWindow screen. If the wrong code is confirmed 3 times, the interface automatically closes.

On the MainWindow screen, after connecting to the database, the user can choose 1 of 4 tabs to operate. After that, if user wants to log out, click “log out” button, or if user wants to completely close the program, click exit.

CHAP 4: EXPERIMENTS, RESULTS AND EVALUATE

4.1 Software Development

Building ChurnCast software

The name "ChurnCast" was chosen for customer churn prediction software for several important reasons. First of all, this is a short, easy to remember and impressive name, helping users easily identify and relate to the software's functions. The combination of the words "Churn" (meaning to churn) and "Cast" (meaning to predict) makes it clear that the purpose of the software is to predict the likelihood of customer churn. Additionally, "ChurnCast" has a professional and modern sound, befitting a data analysis and prediction tool in the digital age.

4.1.1 Overview of software interface

No	Screen		Function
1	Login		Log in to the software
2	Forgot Password		Confirm user information when forgot password
3	Reset Password		Allows users to update their password
4	Connect Database		Access to the database
5	MainWindow	ViewData	View data details
		Predict	Predict customer churn
		Chart	Visualize data with charts

		General	Predict customer churn on file csv

Table 4.1 Overview of software interface

4.1.2 Description of interfaces

4.1.2.1 Login

- **Display**

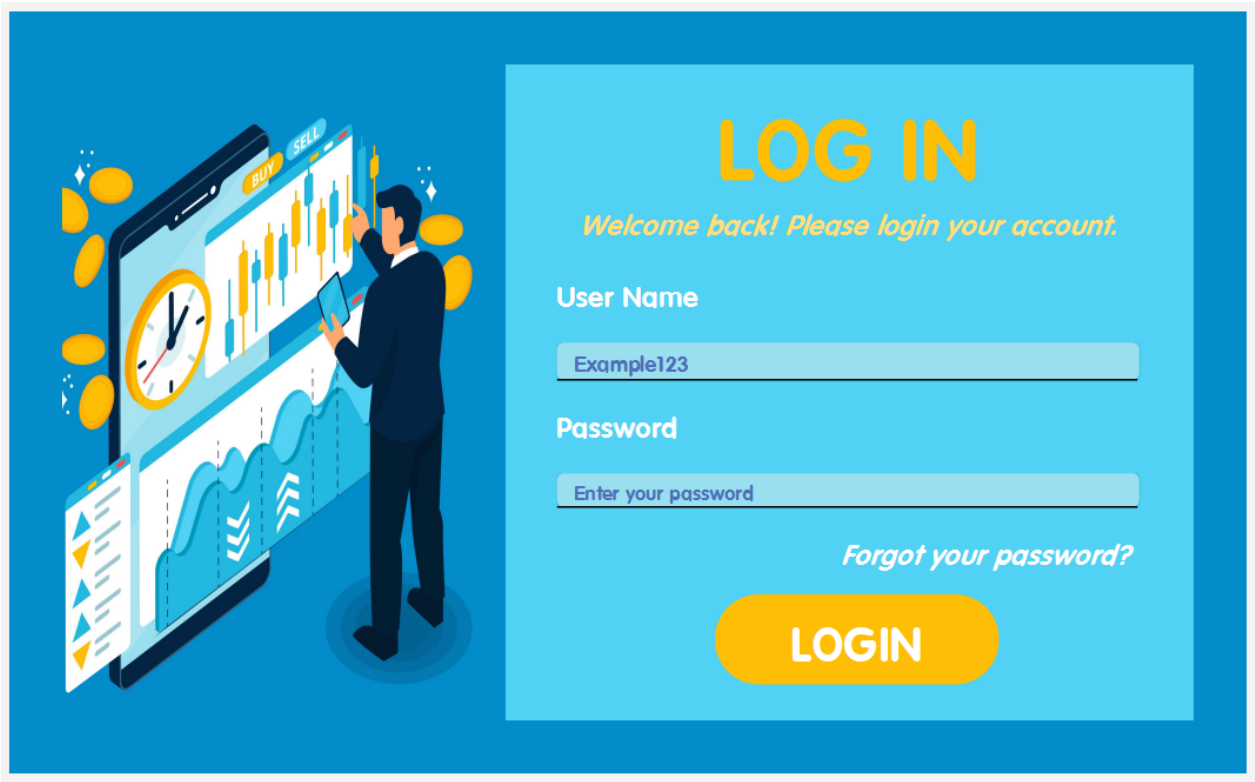

Figure 4.1 Login interface

- **List of operations and processing on the interface**

No	Operation	Event	Handle
1	Select *pushButtonForgot*		Close the current interface, displaying

			the Forgot Password interface
2	Select *pushButtonLogin*	Leave *lineEditUserName* or *lineEditPassWord* blank or both	Show warning
		Entering *lineEditUserName* or *lineEditPassWord* does not match the data in the account file	Show warning
		Enter *lineEditUserName* and *lineEditPassWord* to match the user data in the account file	Close the current interface, showing the MainWindow interface

Table 4.2 List of operations and processing on the interface

4.1.2.2 Forgot Password

- **Display**

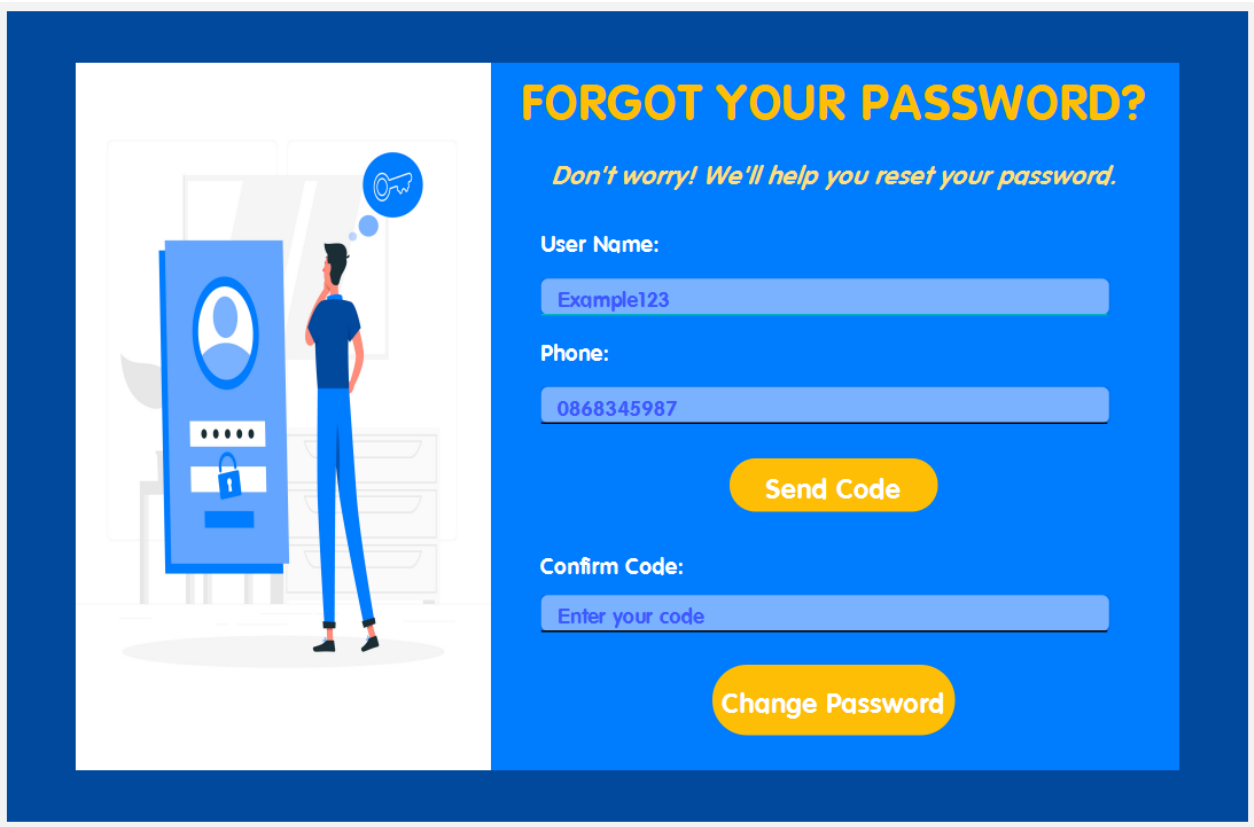

Figure 4.2 Forgot password interface

- **List of operations and processing on the interface**

No	Operation	Event	Handle
1	Select *pushButtonSendCode*	Leave *lineEditUserName* or *lineEditPhone* blank or both	Show warning
		Enter *lineEditUserName* or *lineEditPhone* does not match the data in the account file	Show warning
		Enter *lineEditPhone* and	Currently

		lineEditUserName to match the data in the account file	QMessageBox contains a code valid for 1 minute
2	Select *pushButtonChange*	Enter the correct code in *lineEditConfirmCode*	Close the current interface, the Reset Password interface will appear
		Enter the wrong code in *lineEditConfirmCode*	Show warning
		Leave *lineEditConfirmCode* blank	Show warning

Table 4.3 List of operations and processing on the interface

4.1.2.3 Reset Password

- **Display**

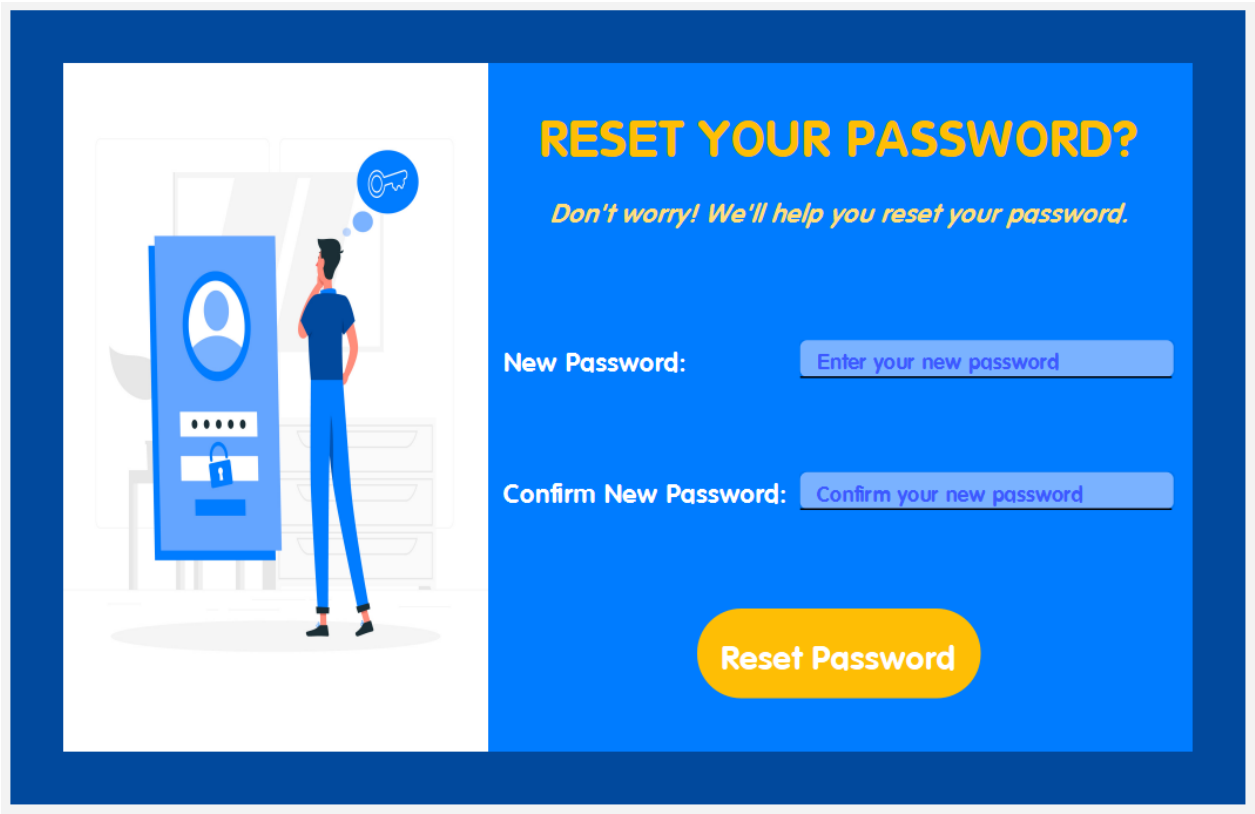

Figure 4.3 Password reset interface

- **List of operations and processing on the interface**

No	Operation	Event	Handle
1	Select *pushButtonResetPassword*	Leave *lineEditNewPassword* or *lineEditConfirmPassword* blank or both	Show warning
		Enter *lineEditNewPassword* and *lineEditConfirmPassword* not as the same	Show warning

		Enter *lineEditNewPassword* and *lineEditConfirmPassword* as the same	Close the current interface, display the Login interface and update user data in the account file

Table 4.4 List of operations and processing on the interface

4.1.2.4 Connect Database

- **Display**

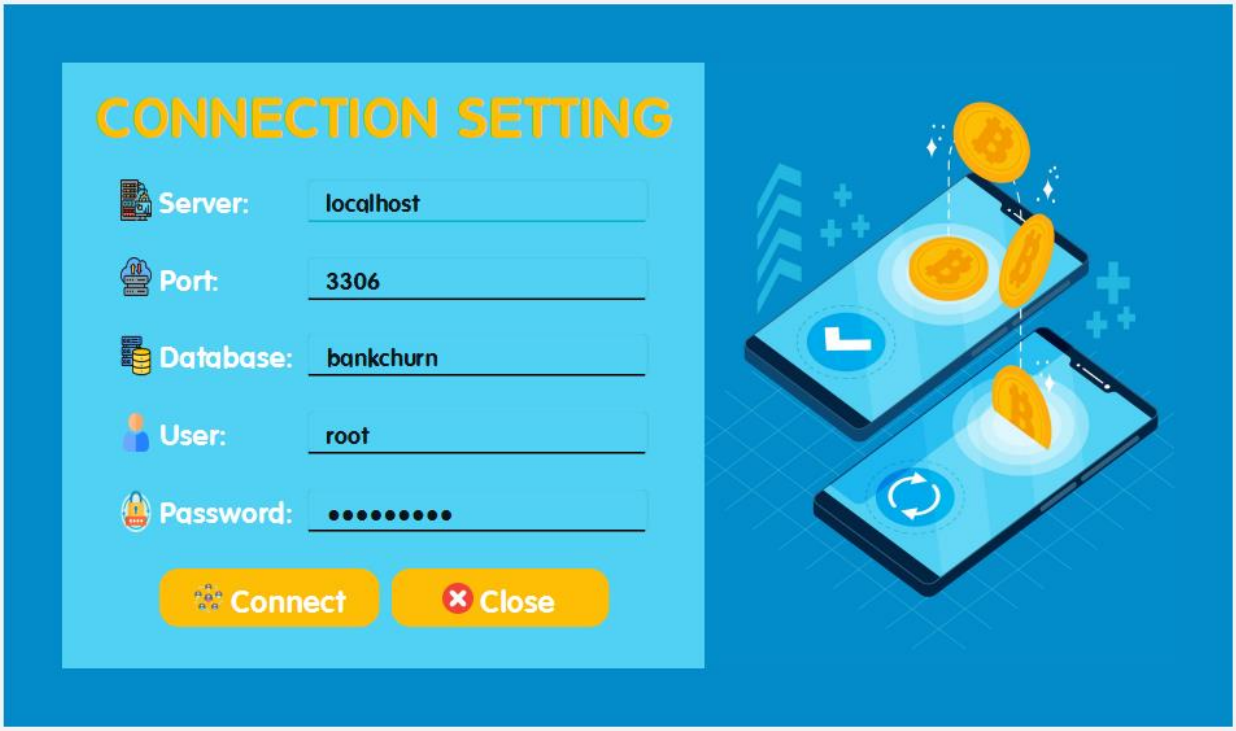

Figure 4.4 Database connection interface

- **List of operations and processing on the interface**

No	Operation	Event	Handle

1	Select *pushButtonClose*		Close the current interface, showing the MainWindow interface and not connecting to the database
2	Select *pushButtonConnect*	Leave any lines blank in *lineEditServer*, *lineEditPort*, *lineEditDatabase*, *lineEditUser*, *lineEditPassword*	Show warning
		Enter all 5 lines *lineEditServer*, *lineEditPort*, *lineEditDatabase*, *lineEditUser*, *lineEditPassword* that match MySQL Workbench account information	Close the current interface, show the MainWindow interface and connect to the database successfully

Table 4.5 List of operations and processing on the interface

4.1.2.5 MainWindow

- **Display**

Figure 4.5 Main interface

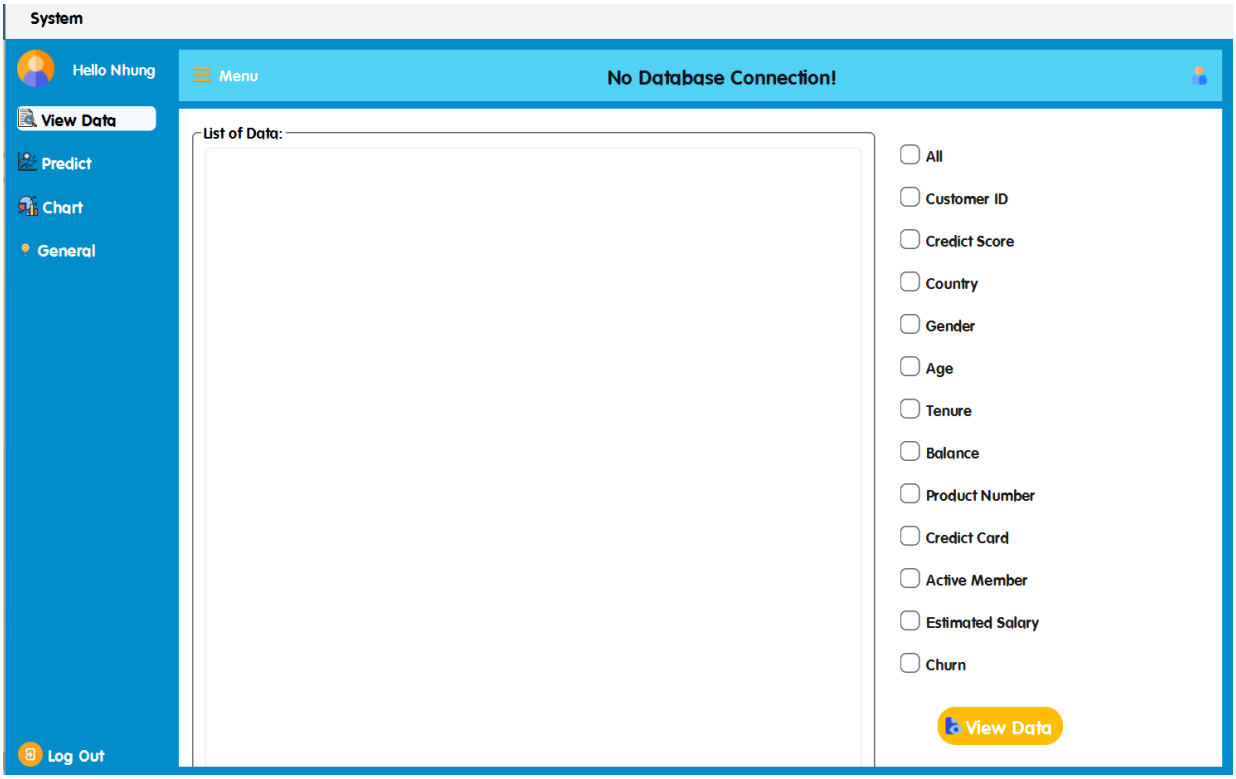

Figure 4.6 Interface to view data details

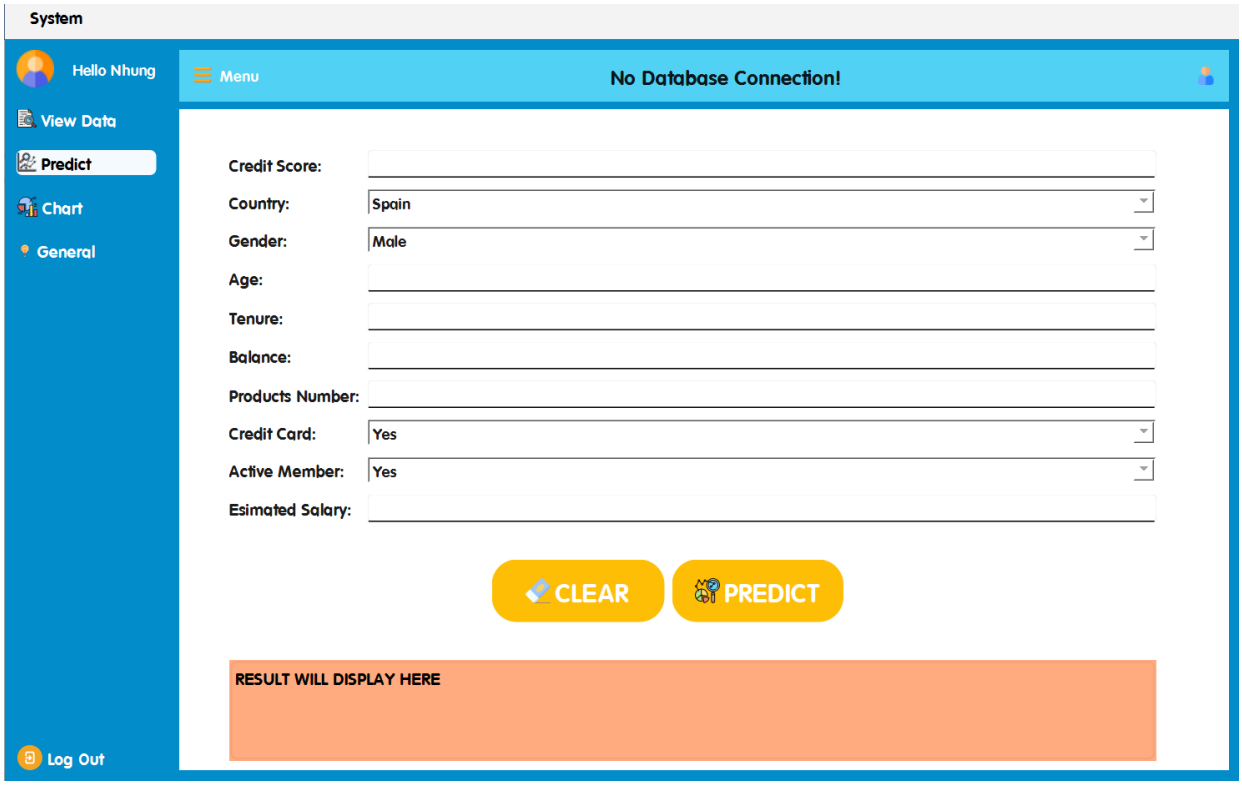

Figure 4.7 Churn prediction interface

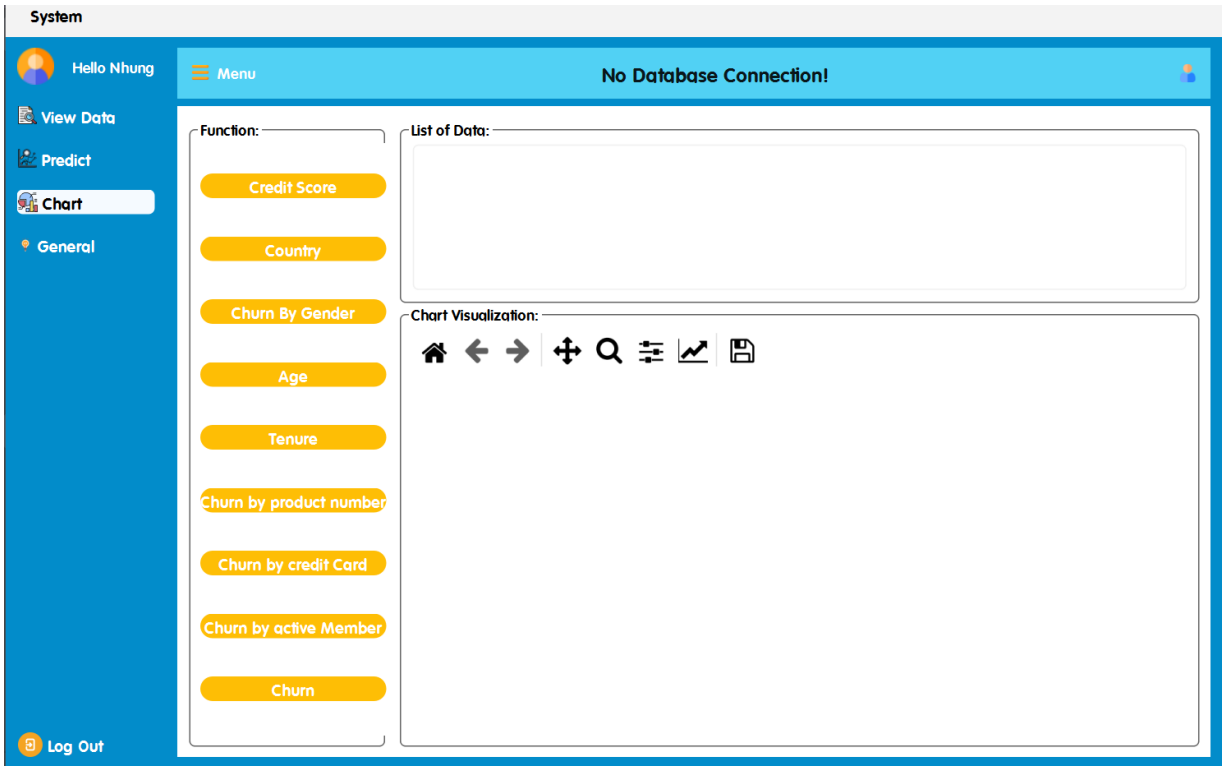

Figure 4.8 Chart display interface

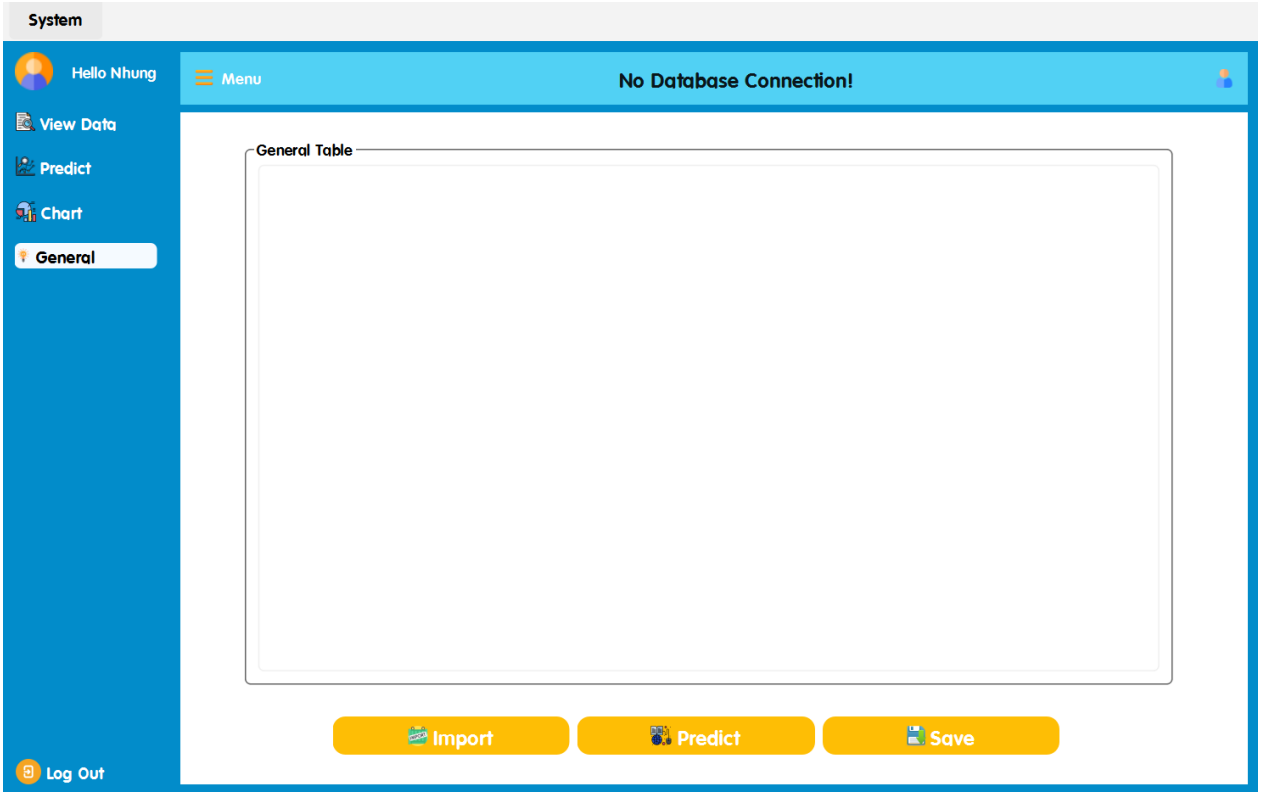

Figure 4.9 General Display Interface

- **List of operations and processing on the interface**

No	Operation	Event	Handle
1	Select *actionConnect_Database* in *menuSystem*		Show Connect Database interface
2	Select *actionSave_New_Trained_Model* in *menuSystem*		Save the model which was just trained by the user to the device
3	Select *actionLoad_Trained_Model* in *menuSystem*		Call and load the previously trained

			model
4	Select *actionExit_2* in *menuSystem*		Close the software
5	Select *pushButtonMenu*		Switch between QWidget IconName and QWidget IconOnly
6	Select *pushButtonView*		Show QWidget ViewData
7	Select *pushButtonPredict*		Show QWidget Predict
8	Select *pushButtonChart*		Show QWidget Chart
9	Select *pushButtonLogOut* or *pushButtonLogOutIcon*		Close the current interface, display the Login interface
In case the connection to the database has not been successful			
10	Operations on *QWidget ViewData, QWidget Predict, QWidget Chart, QWidget Infor*		Untreated
The case has successfully connected to the database			

11	Select *pushButtonViewData* in *QWidget ViewData*	Don't select checkBox	Show warning
		Select *checkBoxAll* and another checkBox	Show warning
		Select different checkBoxes (not Select *checkBoxAll*) or Select only *checkBoxAll*	Show data in *tableWidgetViewData*
12	Select *pushButtonResultPredict* in *QWidget Predict*	Leave any lines blank in *lineEditCreditScore, lineEditAge, lineEditTenure, lineEditBalance, lineEditProductNumber, lineEditSalary*	Show warning

		Enter all 6 lines *lineEditCreditScore*, *lineEditAge*, *lineEditTenure*, *lineEditBalance*, *lineEditProductNumber*, *lineEditSalary*	Show prediction results in *plainTextEditResult*
13	Select *pushButtonResultClear* in *QWidget Predict*		Delete values in lines (if any)
14	Select *pushButtonCreditScore* / *pushButtonCountry* / *pushButtonChurnbyGender* / *pushButtonAge* / *pushButtonTenure* / *pushButtonProduct* / *pushButtonChurnbyCreditCard* / *pushButtonChurnbyActiveMember* / *pushButtonChurn*		Show data and corresponding graphs in *tableWidget* and *verticalLayout_7*
15	Select *pushButtonImport*		Select file to import

	in *QWidget Infor*		data, data that is successfully added will appear in the *tableWidgetView*
16	Select *pushButtonPredictcsv* in *QWidget Infor*	There is data in *tableWidgetView*	Show prediction results in *tableWidgetView*
		There isn't data in *tableWidgetView*	Show warning
17	Select *pushButtonSave* in *QWidget Infor*	There is data in *tableWidgetView*	Save data in the *tableWidgetView*
		There isn't data in *tableWidgetView*	Show warning

Table 4.6 List of operations and processing on the interface

4.2 Evaluate Applicability

The software has provided the following features:

- The software uses a machine learning model trained on historical data about customer behavior to identify customers at high risk of leaving the bank.
- Through analysis, banks can group customers according to their level of churn risk, helping banks focus resources on the customer group with the highest risk.

- Create detailed reports and visual graphs to help banking analysts understand and analyze desired results. Provides powerful data analytics tools that help managers make decisions based on data rather than on gut feeling or experience.

CHAP 5: CONCLUSION, LIMITATIONS AND RECOMMENDATIONS

5.1 Conclusion

Models predicting the likelihood of customers leaving banks using machine learning methods such as XGBoost has achieved encouraging results. Specifically, this model has been evaluated at about 90% accuracy in identifying customers at risk of leaving. This result not only helps bank managers come up with more effective customer retention strategies but also assists in optimizing customer care services, thereby improving their business efficiency. However, the group's research also encountered some limitations. Model accuracy can still be further improved through the use of larger and more diverse data, as well as the incorporation of other advanced machine learning techniques. In addition, implementing the model into practice also requires continuous adjustment and monitoring to ensure optimal effectiveness.

Overall, the research has achieved its initial goal of building and applying a machine learning model to predict customer churn, and at the same time provide useful recommendations for banking managers.

5.2 Limitations and recommendations

5.2.1 Limitations

User interface (UI) and user experience (UX) are not really good: The interface is not really eye-catching, the charts are not really clear, etc.

Usage data may lack diversity, not covering enough important factors that influence customers to leave the bank (additional time factor to be able to calculate the likelihood of customers leaving the bank within a certain period of time)

5.2.2 Recommendations

Expand the data set by integrating other data sources such as service usage behavior, social network data, and customer feedback history to enrich the data set.

Improved user interface (UI) and user experience (UX) to help users more easily engage with and understand predictions.

Provides APIs so businesses can integrate the software with their other systems.

Use Ensemble Methods to improve model and software performance.

Expand to other industries/fields:

- Insurance companies can use this software to predict and reduce customer cancellation rates.
- Telecommunications companies can use models to predict and intervene early with customers at risk of leaving their services.
- Analyze shopping behavior and predict customers who may stop purchasing to provide promotions or special offers.

CHAP 6: FUTURE DIRECTIONS

6.1 Handling Missing data

Missing data (or missing values) is defined as the data value that is not stored for a variable in the observation of interest. To handle missing data, there are two main approaches:

- Deletion: This involves removing rows or columns with missing values (use "dropna" of pandas library)
  - One is to remove the entire information column with missing data from the model building process. This method is simple but has certain limitations. If there are too many columns with missing data and we remove these columns, there will be no information left for building the model.
  - The second way is to remove the missing row from the training set. This also has the same limitation as above, so each row loses a bit of data. The second limitation is that when we encounter a new data point that the model needs to predict, we cannot simply remove that data point and still have to predict a certain value.
- Imputation: This replaces missing values with estimates.
  - Mean/Median/Mode Imputation: Replace missing entries with the average (mean), middle value (median), or most frequent value (mode) of the corresponding column
    - With numeric data, the two most common and simple ways are to fill missing values with mean or median of non-missing values. The

decision to use the mean or median should be considered based on the data before or after handling outliers. The scikit-learn library with the class ‘sklearn.impute.SimpleImputer’ is often used for this task or use ‘fillna()’ of pandas library.

  - With categorical data, since the average value cannot be calculated, the commonly used approaches are to fill in the most frequently occurring value (strategy='mode') or treat the very fact of missing as a special value (strategy='constant') with the special value passed through the fill_value parameter (sklearn.impute.SimpleImputer) or using the 'fillna()' method of the pandas library.

- Forward and Backward Fill: Replace missing values with the previous or next non-missing value in the same variable.These fill methods are particularly useful when there is a logical sequence or order in the data, and missing values can be reasonably assumed to follow a pattern. The method parameter in fillna() allows to specify the filling strategy, and here, it’s set to ‘ffill’ for forward fill and ‘bfill’ for backward fill.
- Use Advanced Techniques like K-Nearest Neighbors (KNN): Estimate missing values by finding similar data points using KNN. This method can preserve data integrity. (KNNImputer of sklearn library) or use a Simple Regression model to fill the data. But, sometimes, using regression to fill the data can cause the data to be overfit.

6.2 Proposed automatic model evaluation

Instead of using the method of manually comparing evaluation indicators between models: XGBoost, Logistic Regression, Decision Tree, Random Forest as above to evaluate and choose the best model, you can use cross_validate in “sklearn.model_selection” in the “scikit-learn” library in Python to automate this selection process.

Cross-Validation is just a method that simply reserves a part of data from the dataset and uses it for testing the model (Validation set), and the remaining data other than the reserved one is used to train the model. If a given model does not perform well on the validation set then it’s gonna perform worse when dealing with real live data. This notion makes Cross-Validation probably one of the most important concepts of machine learning which ensures the stability of models.

In the library mentioned above, for the purpose of performing cross validation, both K-Fold Cross-Validation and cross_validate methods are popular methods commonly used. However, K-Fold Cross-Validation still has to be done manually during the training and evaluation process. Furthermore, if cross_val_score is used in K_Fold, the results will only return scores for one evaluation index, which is true. This is only suitable when only a single evaluation index is needed. Meanwhile, if cross_validate is used, it performs every step from data splitting to training and evaluation automatically. Easily calculate multiple evaluation metrics at the same time, from which you can have a more comprehensive view of the model's performance.

In summary, instead of having to evaluate the indicators manually and potentially leading to bias, using cross_validate is the recommended method for upcoming research articles to evaluate the most effective and suitable model. merge with your dataset.

MATERIALS

Link Google Drive

REFERENCES

1. Dr.Chitra và B.Subahini. (2013), *"Data Mining Techniques và its Applications in Banking Sector"*, International Journal of Emerging Technology và Advanced Engineering, Volume 3 (Issue 38), 219-226.

2. Roberts, John H (2000), *"Developing new rules for new markets"*, Journal of the Academy of Marketing Science, 28, 31-44.

3. Reichheld, Frederick F and W Earl Sasser (1990), *"Zero defections: quality comes to services"*, Harvard business review, 68(5), 105-111.

4. Sanjay Kumar (2024), *"Balancing act: The pros and cons of machine Learning algorithms",* LinkedIn.

5. Rabiloo (2021), *"Các phương pháp đánh giá mô hình học máy, học sâu (Machine learning & Deep learning)".*

6. Saini, A. (2021), *"What is Decision Tree? [A Step-by-Step Guide]",* Analytics Vidhya.

7. Sruthi, E. R. (2021), "*Understand Random Forest algorithms with examples",* Analytics Vidhya.

8. guest_blog (2018), "*Introduction to XGBoost algorithm in machine learning*", Analytics Vidhya.