AI

Article

Secure Internet Financial Transactions: A Framework Integrating Multi-Factor Authentication and Machine Learning

AlsharifHasan Mohamad Aburbeian [1,*] and Manuel Fernández-Veiga [2]

[1] Department of Natural, Engineering, and Technology Sciences, Arab American University, Ramallah P600, Palestine
[2] AtlanTTic Research Center, Universidade de Vigo, 36310 Vigo, Spain; mveiga@det.uvigo.es
[*] Correspondence: a.aburbeian@student.aaup.edu

Abstract: Securing online financial transactions has become a critical concern in an era where financial services are becoming more and more digital. The transition to digital platforms for conducting daily transactions exposed customers to possible risks from cybercriminals. This study proposed a framework that combines multi-factor authentication and machine learning to increase the safety of online financial transactions. Our methodology is based on using two layers of security. The first layer incorporates two factors to authenticate users. The second layer utilizes a machine learning component, which is triggered when the system detects a potential fraud. This machine learning layer employs facial recognition as a decisive authentication factor for further protection. To build the machine learning model, four supervised classifiers were tested: logistic regression, decision trees, random forest, and naive Bayes. The results showed that the accuracy of each classifier was 97.938%, 97.881%, 96.717%, and 92.354%, respectively. This study's superiority is due to its methodology, which integrates machine learning as an embedded layer in a multi-factor authentication framework to address usability, efficacy, and the dynamic nature of various e-commerce platform features. With the evolving financial landscape, a continuous exploration of authentication factors and datasets to enhance and adapt security measures will be considered in future work.

Keywords: multi-factor authentication; fraud detection; machine learning; face recognition; user-friendly system

check for updates

Citation: Aburbeian, A.M.; Fernández-Veiga, M. Secure Internet Financial Transactions: A Framework Integrating Multi-Factor Authentication and Machine Learning. *AI* **2024**, *5*, 177–194. https://doi.org/10.3390/ai5010010

Academic Editor: Ke-Lin Du

Received: 8 December 2023
Revised: 5 January 2024
Accepted: 7 January 2024
Published: 10 January 2024

1. Introduction

FinTech is described as a new financial development that enhances and automates financial services [1]. Mobile wallets, online banking, and payment gateways that offer quick and easy services are examples of financial technologies [2]. The increasing use of such technologies has led to a rise in fraudulent transactions, which makes securing these transactions an issue [3]. Authentication is a procedure in which a user submits some form of credentials to prove identity [4]. The authentication techniques can be one of three categories: something you know (password), something you have (tokens, cards), and something you are (biometrics) [5]. A password has been widely used as a single-factor authentication technique to secure communication between two entities [6]. Although it is a straightforward and easy-to-implement mechanism, it is not sufficient because of its high ability to be revealed [7]. Sharing the password immediately compromises the account. Furthermore, unauthorized access can be gained using a rainbow table [8], a dictionary attack [9], or social engineering approaches [10]. Following the demonstration that authentication with one factor is unsuitable to offer safety, according to various security vulnerabilities [11], two-factor authentication was suggested to enhance security in which a user must provide two credentials for authentication purposes [12–14]. A powerful authentication mechanism, according to the European Union (EU) regulation [15], requires the employment of two or more factors from separate groups to verify users. NIST publications [16] show a link between the degree of safety and the number of authentication elements. Since then, multi-factor authentication (MFA) has presented a greater degree of security [17] by forcing users to provide multiple authentication credentials (more than two) when requesting access to an online system [18,19].

Machine learning (ML) is a branch of artificial intelligence that teaches machines to learn from given data to possess the ability to identify patterns and take actions without the need for human interaction [20]. Since it can handle big data to provide predictions and classifications [21], many recent studies have used ML methods to solve real-world challenges [22–29]. One of these challenges is financial fraud detection, also called credit card fraud detection. Financial fraud is described as unlawful deception that is done to make money [30]. ML has enormous ways to handle financial fraud detection, which include but are not restricted to "intelligent decision engines, artificial neural networks, random forests, naive Bayes, support vector machines, decision trees, logistic regression, and k-nearest neighbor" [31–34].

To secure Internet financial transactions, this study proposes a framework that incorporates ML and MFA.

This study's importance derives from its ability to do the following:

- First: offering a model that can be implemented in the banking sector, e-commerce purchasing websites, and online payment systems.
- Second: using ML as part of MFA will achieve the highest possible security.
- Third: it shows the best way to use MFA conveniently.
- Fourth: provide a comprehensive analysis of the most appropriate ML algorithms and training methods to use in combination with MFA for online transactions.
- Fifth: the possibility of modifying the ML algorithm to comply with the requirements of any electronic system and integrating this algorithm with MFA to provide secure access to data.

Many studies implement an MFA schema to secure online transactions. For example, the authors of [35] utilized a combination of personal identification number (PIN), one-time password (OTP), and global positioning system (GPS). According to a predetermined space between the user's smart tool and the present payment tool, their framework was designed to either approve or deny the payment. Another framework to secure wireless payment systems was proposed by [36]; they used username–password, transaction identification code (TIC), and SMS. TICs are payment identifiers provided by financial organizations to their customers. This code is like OTP, except it provides more secure transaction authentication. Each TIC code is only used once, and then an encryption/decryption mechanism is used for storing TICs as secret codes on mobile devices. The user can quickly select a TIC from a saved list of TICs to begin a secure online transaction utilizing mobile phones. Based on risk assessment criteria, ref. [37] used a layered MFA architecture. The model developed consists of five levels each of which includes one or more authentication elements such as possession, knowledge, or biometric-based elements. The model was improved by including control information components in the last two layers to accommodate layering requirements. Another study [38] used a PIN, device-specific ID, and voice recognition to secure a mobile money application called MPESA. The system stored the mentioned credentials in a database and used them to confirm the identity of the user when performing transactions. Password, OTP, and fingerprint were utilized to secure electronic payment systems [39–42]. Firstly, the user logged into the system using the password, and when the user went to the transferring page, the system asked for fingerprint verification. Finally, after submitting the transaction details, the system sent an OTP to finish the process successfully. Another similar study was proposed in [43], their approach included a PIN, OTP, and face recognition schema. Firstly, the bank collected the user's data such as PIN code, phone number, and user face picture. Secondly, the user needed to sign in with a PIN code and facial picture. When the facial characteristics and PIN were confirmed, the system presented a menu from which the user needed to select a service. Finally, for the process to be performed successfully, the system produced an OTP and sent it to the user's phone for verification. Adding more layers of authentication is conducted by [44].

Four factors were utilized to secure the grid environment consisting of a password, user ID, biometrics, and the user's current location. The addition of the fourth component improved the security standards necessary for large distributed systems such as Banking Grid settings. All mentioned studies in this section [31–40] proposed a different MFA approach without the utilization of ML. Another important point is that these MFA systems do not address the importance of whether the system is convenient for users or not, which affects the usability and attitude toward using such a system. The user's negative feelings towards MFA were mentioned and proved by many studies in the literature [11,45–47].

Other researchers went further and tested using ML with the authentication approach. For example, ref. [48] proposed a two-factor authentication in which the user firstly logs in using his username and password; as a next step, they used neural networks for face recognition. Another two-factor authentication schema based on radio-frequency identification (RFID), IOT, and ML for the attendance system was conducted by [49]. For the initial phase of verification, a microcontroller, GSM module, RFID tag, and RFID reader were utilized. For the second verification, a camera with the "Multi-task Cascaded Convolutional Network (MTCNN)" model was utilized. Students were given attendance if both were satisfactory. In [50], two ML classifiers to analyze user behavior as an authentication schema were deployed; after the user logged in, the authors applied the random forest and k-nearest neighbor to analyze the player's behavior when playing a specific game using two fingers, and they used the collected data to ensure the user authenticity as a continued authentication schema. One type of the MFA that adjusts to the risk profile of the users is called risk-based authentication. To determine the user's degree of risk, ref. [51] determines the authentication techniques that may affect user confidentiality by designing a risk engine that integrates with the system. This engine looks at the user's historical login logs and deploys machine learning techniques to create an appropriate pattern and risk level for authentication factors for every user. To establish a safe and easy authentication method, ref. [52] also utilized risk-based authentication and MFA. They developed two separate libraries, one for backend servers and one for Android applications. The server-side library of the study included an ML risk engine. The choice of authentication elements was informed by the risk levels that this machine learning engine determined using user-specific information including Internet Protocol (IP) addresses, device types, and access times.

The research gap is the shortage of knowledge on the potential of merging ML techniques with the MFA approach to raise the safety of Internet financial transactions. The utilization of the MFA schema without addressing the fact of negative feelings toward MFA systems is not the best way to secure financial transactions. To enhance security, ML techniques have been extensively employed in isolation, but integrating their applications with MFA has not received much attention. Some studies talk about this possible combination; for example, ref. [51,52] utilized ML for ranking authentication factors, denoting which one may be vulnerable. In [50], ML is used for continuous checking for user authenticity by evaluating user actions when using the system. In this study, the detection of fraud access happens after the user signs in to the system, and this is not a sufficient way either. Meanwhile, in [48,49], ML was deployed to enhance the face recognition quality of the users.

This research uses an ML model as an embedded layer of security in the MFA framework. Our system utilizes two stages of security, fingerprint and OTP were deployed to authenticate users in the first stage. In the second, the ML model classifies the current process and asks for a third factor (face recognition) in the scene of fraud. In this way, legitimate users interact with a two-factor authentication system to complete a purchase. Therefore, this research will bridge the gap in the literature by integrating MFA with ML to gain a secure and easy-to-use system.

2. Materials and Methods

2.1. System Architecture

In this study, we propose a framework to secure online transactions. This framework can be compatible with any e-commerce platform in which users use their mobile device or tablet to perform purchases. The system components are shown in Figure 1.

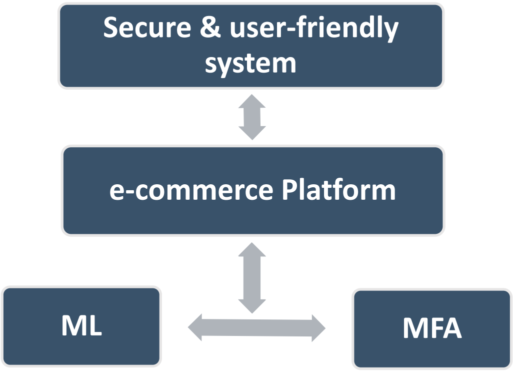


Figure 1. System architecture.

As shown in Figure 1, the system operates through three main parts. Firstly, authentication factors like fingerprint and OTP verify the user. Then, the ML model analyzes transaction data to spot potential fraud. Finally, the information goes to the e-commerce platform (website or app) where users can securely conduct their financial transactions. This streamlined process ensures a strong and easy-to-use mechanism.

2.2. Methodology Used to Secure Internet Transactions

This study's approach is demonstrated in Figure 2.

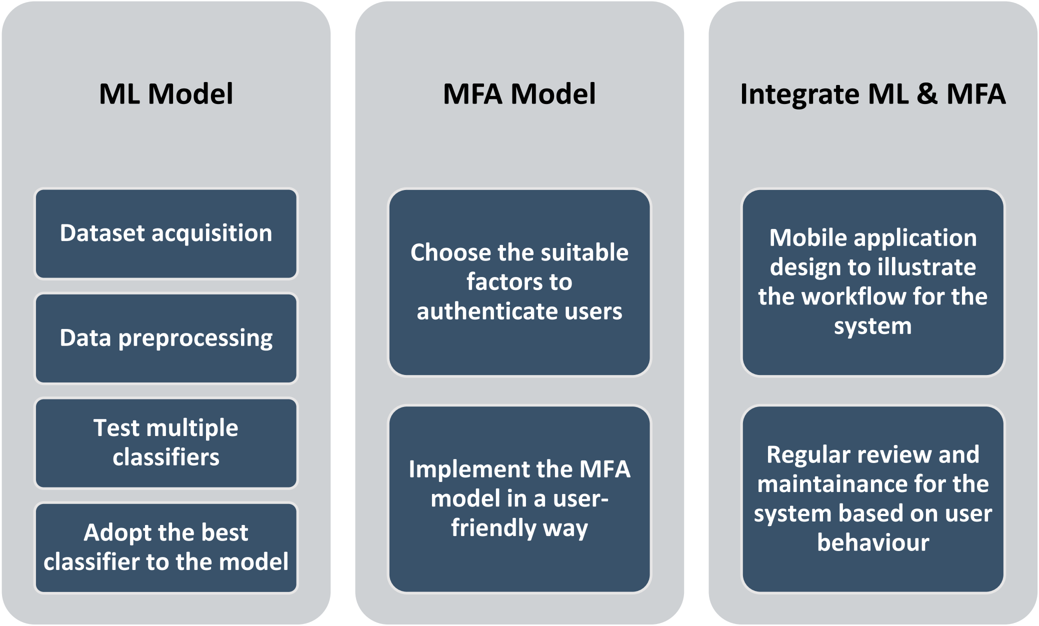

ML Model
Dataset acquisition
Data preprocessing
Test multiple classifiers
Adopt the best classifier to the model
MFA Model
Choose the suitable factors to authenticate users
Implement the MFA model in a user-friendly way
Integrate ML & MFA
Mobile application design to illustrate the workflow for the system
Regular review and maintainance for the system based on user behaviour

Figure 2. Methodology.

As seen in Figure 2, the components of our methodology include three main categories: First, the ML part, which consists of a credit card fraud dataset obtained using an open-source site; the dataset will be discussed in the next section. After performing the preprocessing phase for the dataset, we test different classifiers and use the best one in our model. Second, in the MFA part, we choose the suitable factors to authenticate users and determine the model architecture to gain a feasible MFA implementation. Finally, e-commerce application screens were designed; this design utilizes the integration of ML and MFA for security purposes.

2.3. ML Phase

This section will illustrate the journey to build the ML model. The roadmap starts with the dataset acquisition. After performing the dataset cleaning, different classifiers were tested to build the model. We will discuss the experiment environment, dataset, data preprocessing, and a justification for the chosen ML algorithms in different sections.

2.3.1. Experiment

An HP laptop "Intel(R) Core (TM) i5-10210U CPU @ 1.60GHz 2.11GHz" was used for the testing. The experiment's code was written in Python, and an Anaconda Jupyter Notebook V3 was used to conduct it.

The experiment's purpose was to create a trustworthy detection model with precise classification and identification capabilities. Data for testing and training were taken out of the dataset. To address the imbalanced dataset problem and avoid bias when implementing multiple classifiers, we utilized an oversampling strategy. Finally, to find the optimal settings, we conducted a grid search and standard scaler to achieve the best precision feasible.

2.3.2. Dataset

Using the URL "https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud (accessed on 5 August 2023)", the dataset was downloaded from the Kaggle website. It involves credit card transactions carried out by customers around Europe in September 2013. The dataset entails 31 features: Time, V1–V28, Amount, and Class. All features only have numeric variables; most of these features were transformed using the principal component analysis (PCA) technique. The original characteristics of the data were hidden from the dataset owner due to confidentiality concerns. The PCA transformation is a statistical strategy that aids the dimensionality reduction of big and complicated data. Essentially, the PCA is a calculation of a new coordinate system for the data subspace such that the projecting along each axis has the maximum value for the residual variance. By keeping a subset of the PCA coefficients, we identify the axis that explains most of the variance in the original data. This method aims to facilitate the examination of data by ML models [53]. Table 1 shows a dataset sample.

As shown in Table 1, the only components that did not change by PCA are Time and Amount. The "Time" feature indicates the interval of time in seconds between each payment. The value of the payment is represented by the feature "Amount". It indicates the amount of money involved in each payment. The response feature "Class" has a value of 0 or 1 (0 represents legitimate and 1 represents fraud). This binary categorization is essential to train the ML model to distinguish between legal and illegal transactions. The features "V1–V28" represent various aspects related to financial transaction characteristics and user behavior. There is no more information about the exact description of these variables according to user confidentiality.

Table 1. Dataset sample. Time represents the time in seconds that elapsed for every transaction; V1–V28 represents PCA-transformed numerical variables; Amount represents the transaction amount; Class represents the classification for each transaction (non-fraud is 0, fraud is 1).

Time	**V1**	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**
0.0	−1.359807	−0.072781	2.536347	1.378155	−0.338321	0.462388	0.239599
0.0	1.191857	0.266151	0.16648	0.448154	0.060018	−0.082361	−0.078803
1.00	−1.358354	−1.340163	1.773209	0.37978	−0.503198	1.800499	0.791461
1.00	−0.966272	−0.185226	1.792993	−0.863291	−0.010309	1.247203	0.237609
2.00	−1.158233	0.877737	1.548718	0.403034	0.407193	0.095921	0.592941
V8	**V9**	**...**	**V21**	**V22**	**V23**	**V24**	**V25**
0.098698	0.363787	...	−0.018307	0.277838	−0.110474	0.066928	0.128539
0.085102	−0.255425	...	−0.225775	−0.638672	0.101288	−0.339846	0.16717
0.247676	−1.514654	...	0.247998	0.771679	0.909412	−0.689281	−0.327642
0.377436	−1.387024	...	−0.1083	0.005274	−0.190321	−1.175575	0.647376
−0.270533	0.817739	...	−0.009431	0.798278	−0.137458	0.141267	−0.20601
V26	**V27**	**V28**	**Amount**	**Class**			
−0.189115	0.133558	−0.021053	149.62	0			
0.125895	−0.008983	0.014724	2.69	0			
−0.139097	−0.055353	−0.059752	378.66	1			
−0.221929	0.062723	0.061458	123.5	0			
0.502292	0.219422	0.215153	69.99	0			

2.3.3. Data Preprocessing

The dataset includes a count of 285,299 transactions. The dataset is highly imbalanced. Approximately 0.172% of all transactions are fraudulent, accounting for 492. Using an imbalanced dataset directly may generate several unexpected behaviors. Positive samples (the fraudulent class) are very likely to be incorrect, and the algorithm will have a bias towards forecasting the negative class [54]. In this situation, using over- or under-sampling techniques is the best course of action. The technique that is most frequently employed in the literature is the synthetic minority over-sampling (SMOTE) technique [55–59]. Following a random selection of neighbors from the k-nearest neighbors, for each chosen neighbor, a single sample would be produced in their approximate direction to create the necessary number of synthetic samples. Then, the distance between the feature vector being examined and its neighbor is calculated, that distance is multiplied by an arbitrary digit from 0 to 1, and that difference is added to the feature vector being examined. After performing the SMOTE oversampling approach, the count for each class was 283,253 transactions with a distribution of 50:50. Finally, we split the dataset into training data (80% of transactions with a count of 453,204) and testing data (20% of transactions with a count of 113,302). After that, the dataset became ready to test multiple supervised ML classifiers.

2.3.4. The Choice of ML Classifiers

Credit card fraud detection is an issue of binary categorization (0 is legal and 1 is fraud). To overcome this issue, different ML approaches were implemented in the literature [54–57]. This investigation deploys multiple supervised ML algorithms: naive Bayes, decision trees, logistic regression, and random forest. Notably, the random forest was selected because of its ensemble-based strategy, which is a preferred method according to its capability to deal with complicated, high-dimensional data and avoid overfitting [58]. Decision trees, known for their interpretability, are useful for deciphering the thought process that leads to fraudulent behavior [59]. The inclusion of logistic regression was made possible by the way it offers modeling simplicity and efficiency while perfectly aligning with binary classification jobs. Despite its simplicity, naive Bayes has been successful in handling the categorical data that is frequently encountered in fraud detection scenarios [60]. We examined different algorithms to locate the most accurate one. These algorithms were chosen due to their effectiveness and simplicity, making them easier and faster when integrating them with the MFA framework.

2.4. MFA Phase

The choice of authentication factors was carefully considered. A strong security architecture was built using three authentication factors: username–password (user can activate fingerprint), OTP, and face recognition. These criteria were chosen because of their distinct advantages and capacity to offer a multi-layered security solution. Numerous online systems frequently use username–password combinations, which offer a fundamental level of protection. Additionally, using OTPs provides a further level of dynamic security, ensuring that a time-sensitive code is required to access the system [61]. Face recognition as the third authentication factor makes use of biometric technology to further strengthen security [62]. Utilizing the capabilities of contemporary biometric authentication technology, the research tries to achieve a balance between user familiarity and increased safety.

2.5. Combining MFA with ML

After building the ML model and determining the authentication factors to use in the final MFA model, we combine both in the final MFA framework. This section will discuss the hardware and software tools, the workflow of our proposed framework, and the e-commerce platform (application) that has been designed.

2.5.1. Experiment

The processing power needed for the phases of application design and development was supplied by the same device used in building the ML model. The "Adalo" website was the key piece of software used to customize the application displays. Adalo is an empowering no-code platform that lets people and companies create web and mobile applications without requiring a deep understanding of coding. It enables users to visually build and customize application components with its simple drag-and-drop functionality [63].

2.5.2. Proposed Framework

This part will discuss the system's working principle, which is illustrated in Figure 3.

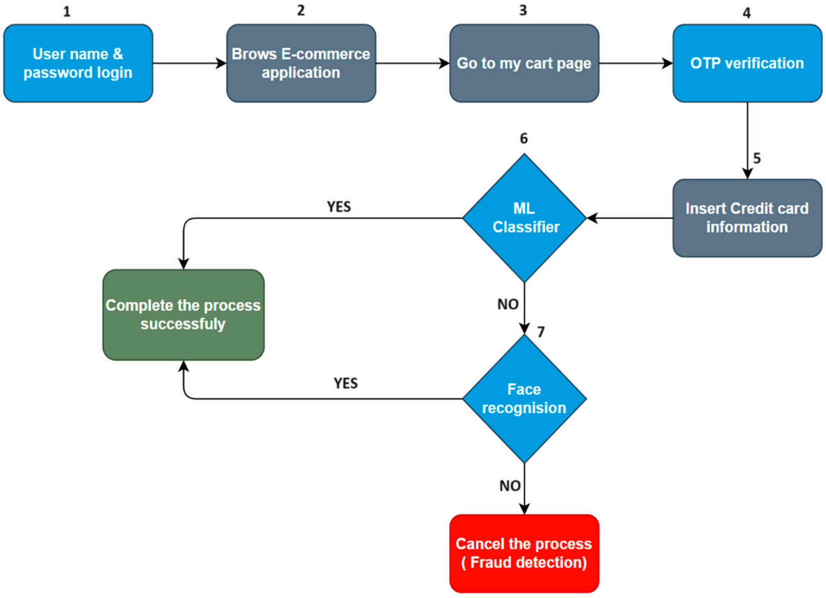

1
User name & password login
2
Brows E-commerce application
3
Go to my cart page
4
OTP verification
5
Insert Credit card information
6
ML Classifier
YES
Complete the process successfuly
NO
7
Face recognision
YES
NO
Cancel the process (Fraud detection)

Figure 3. Framework working principle.

As seen in Figure 3, the user first will sign in to the application using his credentials: username and password. After registration, the user can enable the fingerprint API to sign in to the system. When finished with browsing different products and choosing what to buy, the user has to go to the cart page and perform a purchasing process. Before redirecting the user to the credit card information page, an OTP verification will be delivered to the user's phone. At this stage, the ML model will classify this payment as fraudulent or legitimate. If the classifications were fraud, the user will be asked for face recognition authentication to complete the purchasing process successfully. Otherwise, the process will be canceled.

2.5.3. Application Design

Android e-commerce application screens were designed to make the idea simple to comprehend. The MFA framework was successfully implemented due to the design of the application panels. A user-centered approach was used during the design process to guarantee user-friendliness and ease of use. The design approach includes a logical and clear screen flow during the purchase process. The location of authentication elements was given special consideration to reduce user friction while ensuring high security. The design of the application is shown in the results part.

3. Results

The results will be divided into two parts: ML results, which show the supervised classifiers implemented in this study, and mobile application results, which show the application screens' design with the MFA implementation principle.

3.1. ML Results

3.1.1. Confusion Matrix

A statistic called the confusion matrix gives information about the groups that were correctly and incorrectly categorized. The confusion matrix produces a two-by-two matrix as its output, which shows the values of true positive (TP), true negative (TN), false positive (FP), and false negative (FN). TP and TN indicate that the positive and negative predictions made by the model are accurate. FP and FN denote a false prediction done by the model [64]. Figure 4 presents the confusion matrix results.

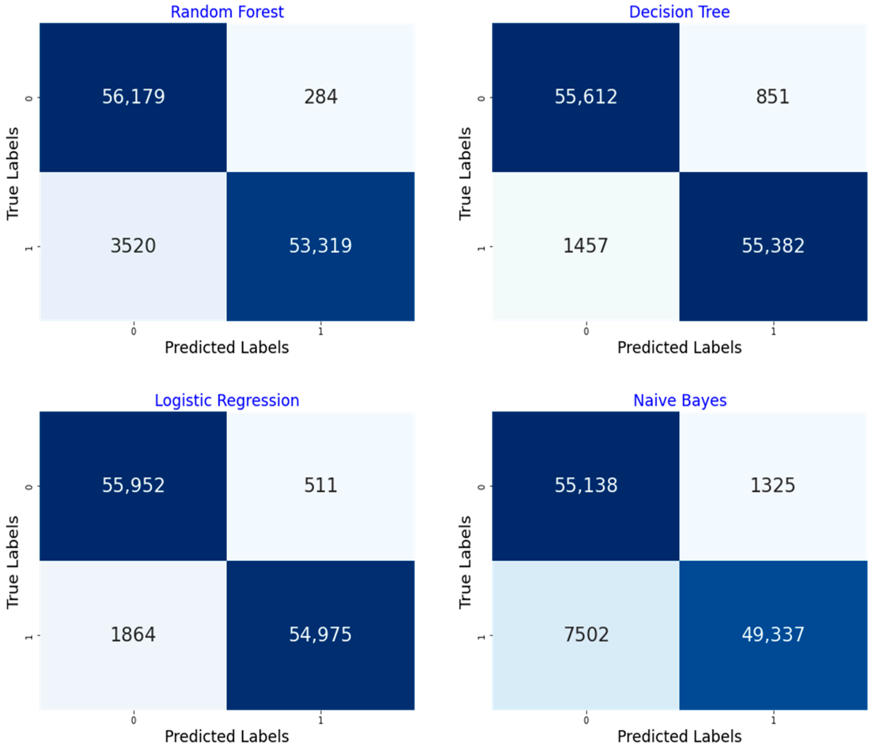


Figure 4. Confusion matrix results for implemented classifiers.

Based on Figure 4, the random forest classifier strikes a correct prediction of TP = 56,179 and TN = 53,319, and the count of inaccurate forecasts was FP = 284 and FN = 3520. The results of the decision tree classifier were TP = 55,612, TN = 55,382, FP = 851, and FN = 1457. The logistic regression results gain a total of TP = 55,952, TN = 54,975, FP = 511, and FN = 1864. While the naive Bayes classifier results were TP = 55,138, TN = 49,337, FP = 1325, and FN = 7502.

3.1.2. Classification Report

Because of working with an extremely skewed dataset, testing the algorithm and just demonstrating its accuracy is insufficient to demonstrate its dependability. This led to the utilization of precision, recall, and F1 score metrics to assess results. Accuracy is a measure that indicates how a classifier predicts results correctly. It can be computed by dividing the total number of precise estimates by the overall predictions (Equation (1)) [27].

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})} \tag{1}$$

Precision is a measure of the number of accurate positive guesses. It can be computed by dividing the number of TP by the sum of FP and TP (Equation (2)) [27].

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \tag{2}$$

The percentage of favorable instances where the classifier correctly predicted is called recall. It can be computed by dividing the total of the TP and FN by the TP (Equation (3)) [27].

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \tag{3}$$

Recall and precision are combined into one metric called the F1 score. It is a technique for figuring out the harmonic mean, which works better for ratios than the conventional mean (Equation (4)) [27].

$$\text{F1 score} = 2 \times \frac{(\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})} \tag{4}$$

The classification report results for each classifier are presented in Table 2.

Table 2. Classification report results for implemented classifiers.

Classifier	Accuracy	Class	Precision	Recall	F1 Score	Support
Random Forest	96.717%	0	0.94	0.99	0.97	56,463
		1	0.99	0.94	0.97	56,463
Decision Tree	97.881%	0	0.97	0.99	0.98	56,463
		1	0.99	0.97	0.98	56,463
Logistic Regression	97.938%	0	0.97	0.99	0.98	56,463
		1	0.99	0.97	0.98	56,463
Naive Bayes	92.354%	0	0.88	0.98	0.93	56,463
		1	0.97	0.87	0.92	56,463

As stated in Table 2, decision tree and logistic regression obtained approximately the same degree of accuracy. The decision tree gained an accuracy of 97.881%, with precision, recall, and F1 scores of 97%, 99%, and 98% consequently for class 0 (legitimate transaction) and precision, recall, and F1 scores of 99%, 97%, and 98% consequently for class 1 (fraud transaction). Logistic regression gained an accuracy of 97.938%, with a precision of 97%, recall of 99%, and F1 scores of 98% for class 0 and a precision of 99%, recall of 97%, and

F1 score of 98% for class 1. The random forest accuracy was 96.717%, and the naive Bayes accuracy was 92.354%.

3.1.3. The ROC Curve

The ROC curve is a visual depiction that indicates the capability to identify problems of a binary classification system by drawing the rate of true positives against the rate of false positives. AUC values range from 0 to 1, where 0.5 denotes a classifier that is no more successful than a wild guess and 1 denotes perfect performance [65]. Figure 5 shows the ROC results.

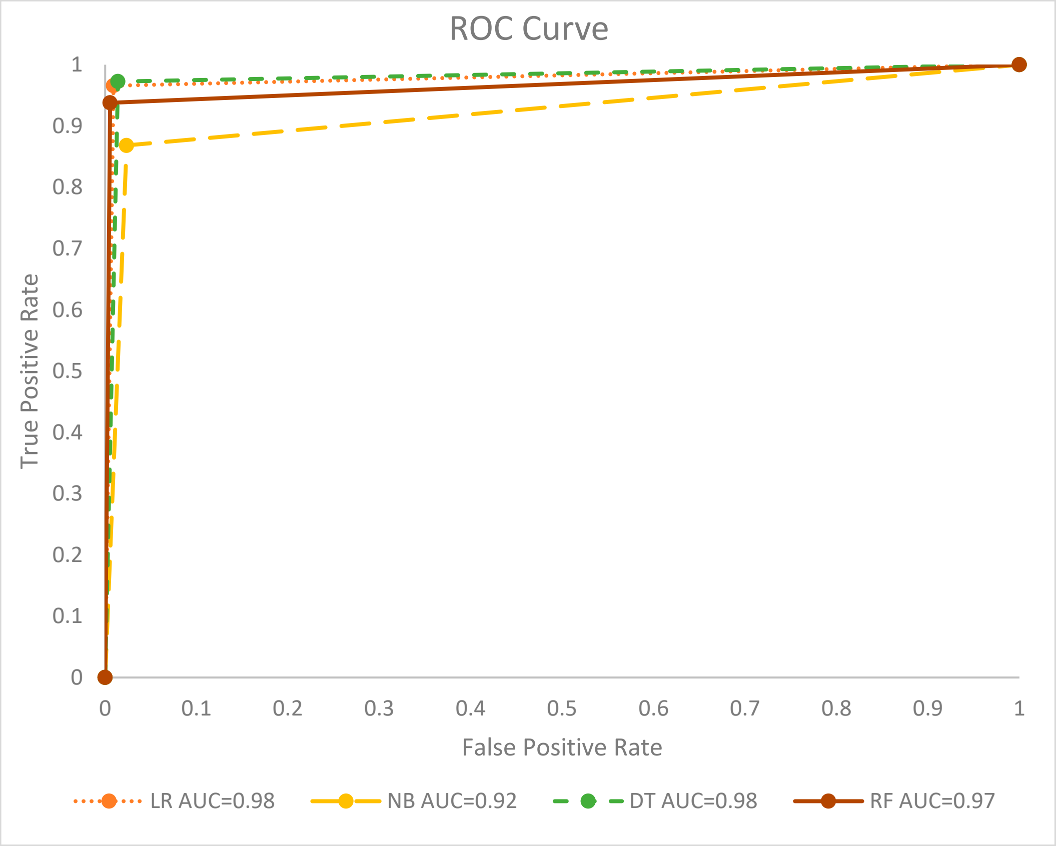
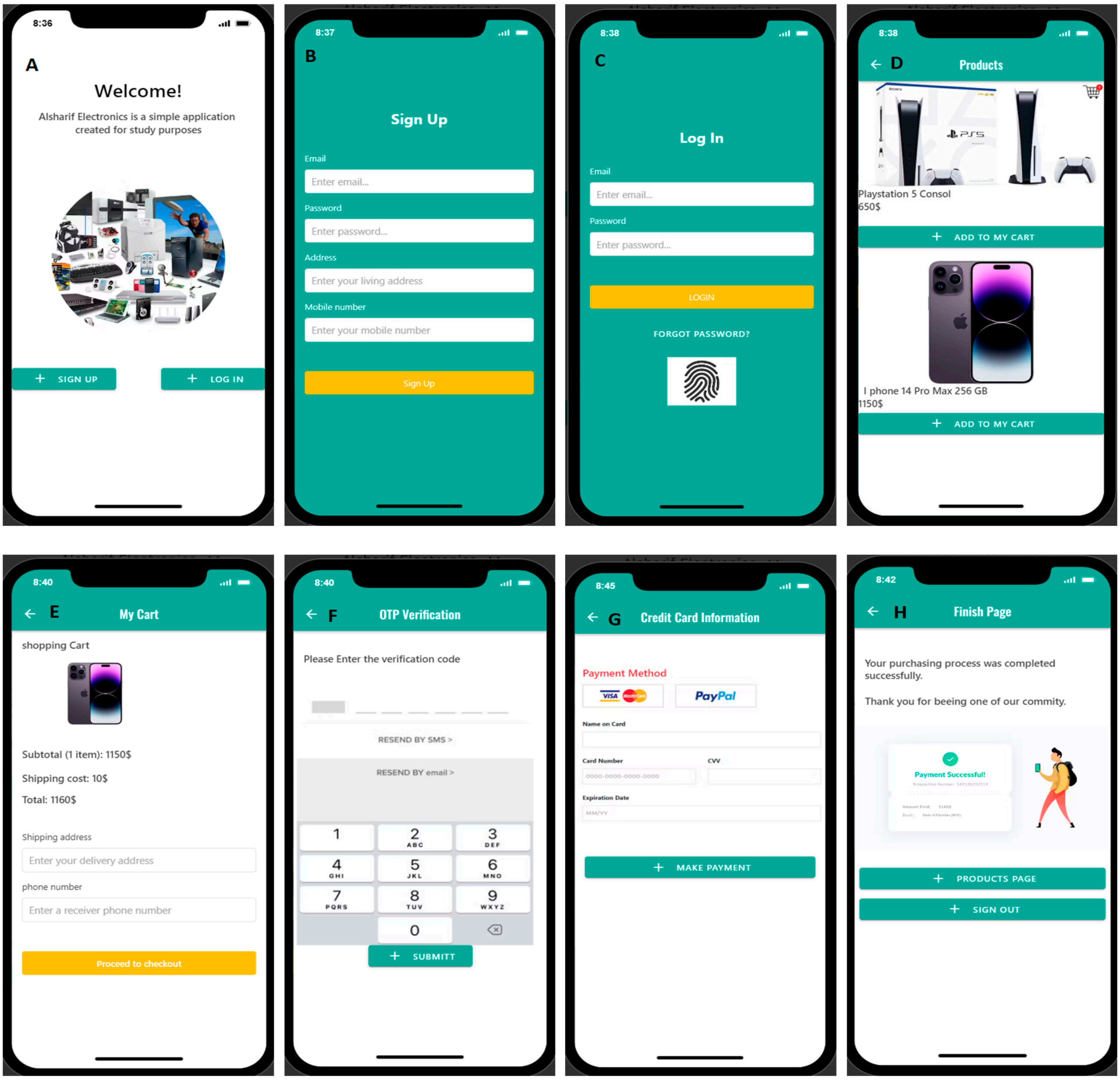


Figure 5. The ROC curve results (all classifiers).

The AUC for the logistic regression and decision tree classifiers was 0.98, as seen in Figure 5. It indicates that 98% of positive cases are accurately classified as positive and 98% of negative examples as negative by the classifier. Additionally, it demonstrates that the classifier has a low rate of false positives or the percentage of times it mistakenly classifies negative cases as positive. In contrast, the AUC of random forest and naive Bayes was 0.98, 0.97, and 0.92, respectively.

3.2. Mobile Application Results

At this stage, we are focused on developing the user interface for an Android application. The primary objective is to create screens and visual elements that effectively convey the principles of our proposed security model. Figure 6 provides a clear visual representation of the security measures required to complete a purchase through the application.



Alsharif Electronics is a simple application created for study purposes

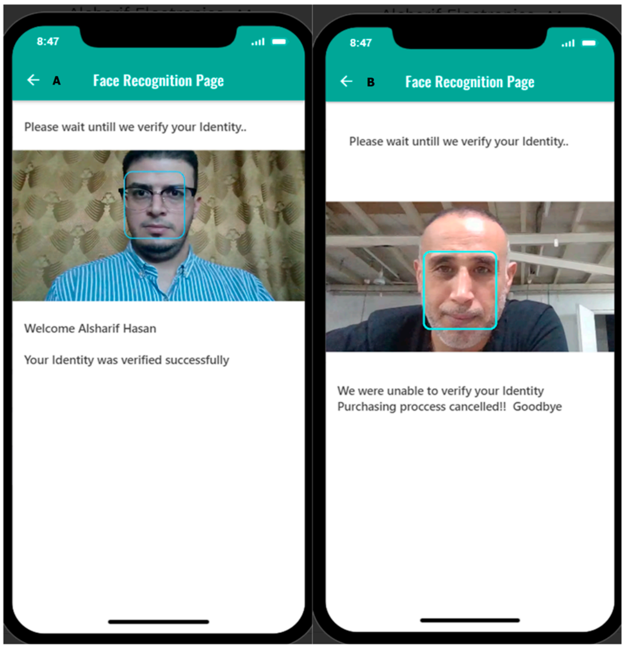
Your purchasing process was completed successfully.

Thank you for beeing one of our commity.

Please Enter the verification code

Figure 6. Mobile application screens: (**A**) welcome page; (**B**) sign-up page; (**C**) login page; (**D**) products page; (**E**) cart page; (**F**) OTP verification page; (**G**) credit card information page; (**H**) finish page.

As shown in Figure 6, (A) the user has to choose to go to the login screen or sign-up screen. In the sign-up screen of Figure 6, (B) the user will make an account by providing some information such as e-mail address, password, address, and mobile number. After making an account, the user can sign in (C) using the username and password or he can activate the fingerprint to browse the application; this step is the first authentication factor. Figure 6D shows the products offered in the e-commerce application in which the user can browse and add any product to the cart. After finishing browsing, the user will proceed to the My Cart page (Figure 6E) to revise the chosen products and the amount of the transaction; the user must enter the shipping address and mobile number to communicate with the delivery company. When finishing this step and pressing proceed to check out, an OTP will be delivered to the user's phone. The user must enter the sent number into the screen (Figure 6F) (second authentication factor). After successful OTP verification, the user shall be routed to the credit card information page (Figure 6G). While the user enters the credit card information, the ML model will evaluate the purchasing process and classify it as fraudulent or legitimate. If the process is classified as legitimate, the process will be completed successfully, as shown in Figure 6H. Otherwise, the user will be asked for face recognition (third authentication factor). Successful face recognition (Figure 7A) will complete the purchasing process successfully. Otherwise (Figure 7B) the purchasing process will be canceled and the transaction will be classified as fraudulent.



Please wait untill we verify your Identity..

Welcome Alsharif Hasan

Your Identity was verified successfully

Please wait untill we verify your Identity..

We were unable to verify your Identity
Purchasing proccess cancelled!! Goodbye

Figure 7. Face recognition verification: (**A**) successful face recognition; (**B**) unsuccessful face recognition.

The application shown in Figures 6 and 7 considers crucial factors: user perceptions and experiences about the usability and efficacy of the multi-layer security strategy. If the security measures are too complex, usability issues might arise, which might frustrate users and possibly cause resistance. However, usability can be improved by a simple and well-designed system. User trust in security measures is an essential requirement for effectiveness, and problems like false positives or negatives may harm that faith. The deployment of secured authentication factors and an accurate ML model is crucial to maintain efficiency.

4. Discussion

As shown in the results section, the logistic regression outperformed the other classifiers. Based on Figure 4, it gains a total of 110,966 accurate predictions. This demonstrates how well the model classified transactions with accuracy. The high percentage of accurate predictions attests to the model's dependability in differentiating between fraud and legitimate payments. However, the 2336 inaccurate predictions show when the algorithm misclassified transactions, highlighting the trade-off between overall accuracy and error rates.

The results of the accuracy, precision, recall, and F1 scores are presented in Table 2. The logistic regression classifier gains an accuracy of 97.938%. For class 1, the classifier gains a precision of 99%, indicating a low level of false positives, and a recall of 97%, effectively detecting transactions of fraud. The F1 score of 98% represents a balanced performance in recognizing fraudulent transactions. Overall, the logistic regression model showcases robust accuracy and well-balanced precision, recall, and F1 scores for both instances, making it a highly effective choice for the binary classification task. These results were too close to the decision trees results, which gained an accuracy of 97.881%, with precision, recall, and F1 scores of 99%, 97%, and 98% correspondingly for group 1.

The AUC values vary from 0 to 1, where 1 represents a perfect score and 0.5 represents an arbitrary guess classifier. According to Figure 5, both logistic regression and decision trees strike an AUC of 0.98. This denotes that these classifiers have an excellent capability to distinguish between different instances. The model's predictions are highly accurate, and it performs well across different thresholds.

To compare our results with other studies, Table 3 mentions some of the studies that used the same dataset and the same ML algorithms to solve the financial fraud issue.

Table 3. Related studies' results; LR: logistic regression; NB: naive Bayes; DT: decision trees; RF: random forest.

Reference	Year	Classifier	Accuracy	Precision	Recall
[34]	2013	LR NB	0.54 0.97	0.38 0.97	0.58 0.95
[66]	2016	DT	0.90	0.83	0.83
[67]	2018	LR DT NB	0.96 0.96 0.97	- - -	- - -
[68]	2018	LR DT RF NB	0.94 0.90 0.94 0.90	0.95 0.91 0.95 0.91	0.95 0.91 0.95 0.91
[69]	2019	LR DT RF	0.97 0.97 0.99	0.98 0.98 0.99	- - -
[70]	2020	LR DT RF	0.90 0.91 0.95	0.92 0.90 0.96	0.93 0.92 0.95
[71]	2022	LR DT RF	0.96 0.77 0.85	0.98 0.77 0.93	0.93 0.76 0.78
[72]	2023	LR RF	0.69 0.64	0.59 0.77	0.82 0.55
[73]	2023	RF NB	0.99 0.99	0.99 0.99	0.99 0.99
[74]	2023	DT RF	0.51 0.84	0.38 0.87	0.75 0.81

Depending on Table 3, we can notice that our results demonstrate its superiority over other related studies. Some studies, refs. [69,73], gain an accuracy of 0.99, which indicates that their algorithms are overfitted and need further enhancements. Despite the good results, other studies in the table do not use different metrics to improve the quality of their investigation, like showing results for both classes or using the ROC curve.

According to Figure 6, the mobile application design demonstrated the usefulness of combining MFA processes with ML. It was carefully built through the combination of hardware and software. The application panels showed an easy-to-use interface that led users through a safe transaction procedure, from creating an account and choosing a product to implementing multi-layered authentication during transactions.

The suggested framework can be tailored to suit various e-commerce platforms and payment systems by integrating platform-specific authentication techniques and modifying the data collection procedure, integrating security measures as the first layer of security. Building the ML models is included in the second layer. Crucial actions include customizing these layers to the specifications of each platform, modifying application design by applicable rules, and ensuring compliance with security and laws. Retraining the ML model using user profiles after a period will improve accuracy and enable data to be tuned to maximize model efficiency. The efficacy of the framework is further improved by feedback mechanisms, iterative enhancements, and continuous monitoring across a variety of online transaction scenarios.

Integrating MFA and ML in securing Internet financial transactions may face many challenges, some of these challenges are as follows:

1. Authentication factors: potential user resistance or discomfort with the chosen methods presents one challenge when choosing MFA for security purposes. Sometimes people feel that multi-factor authentication is too complicated and annoying to use, which can cause resistance or lower user acceptance of such systems. The challenge is to choose a secure factor to authenticate users along with taking into consideration the ease of use.
2. Data availability: one major obstacle is the absence of necessary data. Organizations may choose to hide financial transaction data according to privacy and security considerations, and the needed datasets may not be publicly accessible due to the sensitivity of this data.
3. Data quality: this study is impacted by the quality of the accessible data. Data that is missing, incorrect, unbalanced, or inconsistent can make ML models and authentication systems less effective and possibly produce biased or incorrect results. In particular, most of the available datasets are transformed using the PCA transformation technique.
4. Technical limitations: technical barriers, such as compatibility issues or limited storage capacity, and processing speed may restrict the power to handle, process, and store big data efficiently.

To address these challenges, this research implements the following strategies:

1. Suitable MFA implementation: a user-centric strategy was used to gain adaptable, and secure system implementation. The adaptive implementation of the MFA system led to interaction with only two factors when put into practice. A third factor is required if the ML algorithm classifies the transaction as fraud. This preserves strong security standards while simultaneously improving usability.
2. Data cleaning and preprocessing: using techniques to remove errors and deal with unbalanced datasets that could affect the models' accuracy in cleaning and preparing data. This was conducted successfully and discussed in Section 2.3.3 (data preprocessing).
3. Replication: conducting the ML analysis at different times to confirm and guarantee the reliability and consistency of the results while reducing the influence of anomalies or errors.
4. Algorithm and analysis suitability: using the right statistical techniques and ML algorithms to analyze the data while taking hardware constraints into account. Identifying and evaluating the best algorithms for the particular use case of safe financial transactions was conducted carefully. The implemented ML algorithms were simple and accurate to overcome the hardware limitations and facilitate the integration of ML and MFA into one model.

Our proposed framework does not conflict with the same studies in the literature [75–79]. In future work, implementing this framework in real-world applications will be valuable, using other biometrics or adding more security layers may offer additional safety to the framework.

5. Conclusions

This study aims to propose a framework to secure Internet financial transactions by integrating MFA and ML. Our framework overcomes the previous work in the literature by adding more layers of security while offering a user-friendly system. Taking advantage of the abilities of ML and making it work as an embedded layer of security within an MFA framework was the strength and distinction of this study.

Many supervised ML algorithms were investigated to build an ML model that can accurately identify illegal payments. The implemented algorithm's accuracy varied from 92.354 to 97.938%. The logistic regression algorithm was the best one, with AUC, precision, recall, and F1 scores of 0.98, 0.99, 0.97, and 0.98, respectively.

The username–password, fingerprint, OTP, and face recognition were deployed in the MFA model to authenticate users. E-commerce application screens were designed to offer a better understanding of the proposed framework and show how users will interact with the system easily and securely.

The results of this investigation show that many domains of security can be greatly improved and refined by incorporating ML techniques as a core component of MFA. However, this study is only the beginning of a larger and more thorough investigation into this kind of integration, highlighting the need for additional research that looks into various authentication factors across various datasets to balance security and usability.

Author Contributions: A.M.A. designed the methodology, built the model, processed the data, interrupt results, and wrote the first draft of the manuscript; M.F.-V. helped in the interpretation of the results, helped in the data visualization, revised and edited the manuscript, helped in the experimental work, and supervision. All authors have read and agreed to the published version of the manuscript.

Funding: This work received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Schueffel, P. Taming the Beast: A Scientific Definition of Fintech. *J. Innov. Manag.* **2016**, *4*, 32–54. [CrossRef]
2. Ul, B.; Khan, I.; Olanrewaju, R.F.; Mehraj Baba, A.; Langoo, A.A.; Assad, S. A Compendious Study of Online Payment Systems: Past Developments, Present Impact, and Future Considerations. *IJACSA Int. J. Adv. Comput. Sci. Appl.* **2017**, *8*, 256–271. [CrossRef]
3. Cherif, A.; Badhib, A.; Ammar, H.; Alshehri, S.; Kalkatawi, M.; Imine, A. Credit Card Fraud Detection in the Era of Disruptive Technologies: A Systematic Review. *J. King Saud. Univ. Comput. Inf. Sci.* **2023**, *35*, 145–174. [CrossRef]
4. Meneses-Claudio, B.; Huamaní, E.L.; Yauri-Machaca, M.; Meneses-Claudio, J.; Perez-Siguas, R. Authentication and Anti-Duplication Security System for Visa and MasterCard Card. *Repos. Inst. UTP* **2022**, *10*, 1–5. [CrossRef]
5. Nandalwar, P.J.I.; Gaikwad, P.V.; Kulkarni, P.S. A Survey and Comparison on User Authentication Methods. *Int. J. Innov. Eng. Res. Technol.* **2016**, *3*, 1–7.
6. Bonneau, J.; Herley, C.; Van Oorschot, P.C.; Stajano, F. Passwords and the Evolution of Imperfect Authentication. *Commun. ACM* **2015**, *58*, 78–87. [CrossRef]
7. Gunson, N.; Marshall, D.; Morton, H.; Jack, M. User Perceptions of Security and Usability of Single-Factor and Two-Factor Authentication in Automated Telephone Banking. *Comput. Secur.* **2011**, *30*, 208–220. [CrossRef]
8. Ah Kioon, M.C.; Wang, Z.S.; Deb Das, S. Security Analysis of MD5 Algorithm in Password Storage. *Appl. Mech. Mater.* **2013**, *347–350*, 2706–2711. [CrossRef]
9. Wang, D.; Wang, P. Offline Dictionary Attack on Password Authentication Schemes Using Smart Cards. In *Information Security: 16th International Conference, ISC 2013, Dallas, Texas, November 13–15, 2013, Proceedings*; Springer International Publishing: New York, NY, USA, 2015; Volume 7807, pp. 221–237.
10. Heartfield, R.; Loukas, G. A Taxonomy of Attacks and a Survey of Defence Mechanisms for Semantic Social Engineering Attacks. *ACM Comput. Surv. (CSUR)* **2015**, *48*, 1–39. [CrossRef]
11. Wang, D.; He, D.; Wang, P.; Chu, C.H. Anonymous Two-Factor Authentication in Distributed Systems: Certain Goals Are Beyond Attainment. *IEEE Trans. Dependable Secur. Comput.* **2015**, *12*, 428–442. [CrossRef]
12. Petsas, T.; Tsirantonakis, G.; Athanasopoulos, E.; Ioannidis, S. Two-Factor Authentication: Is the World Ready? Quantifying 2FA Adoption. In Proceedings of the 8th European Workshop on System Security, EuroSec 2015, Bordeaux, France, 21 April 2015. [CrossRef]
13. Schneier, B. Two-Factor Authentication. *Commun. ACM* **2005**, *48*, 136. [CrossRef]
14. Dumortier, J. Regulation (EU) No 910/2014 on Electronic Identification and Trust Services for Electronic Transactions in the Internal Market (EIDAS Regulation). *SSRN Electron. J.* **2016**, *48*, 1–39. [CrossRef]

15. Burr, W.; Dodson, D.; Polk, W.T. Archived NIST Technical Series Publication Electronic Authentication Guideline. 2004. Available online: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-63ver1.0.2.pdf (accessed on 7 December 2023).
16. Ometov, A.; Bezzateev, S.; Mäkitalo, N.; Andreev, S.; Mikkonen, T.; Koucheryavy, Y. Multi-Factor Authentication: A Survey. *Cryptography* **2018**, *2*, 1. [CrossRef]
17. Kennedy, E.; Millard, C. Data Security and Multi-Factor Authentication: Analysis of Requirements under EU Law and in Selected EU Member States. *Comput. Law. Secur. Rev.* **2016**, *32*, 91–110. [CrossRef]
18. Dasgupta, D.; Roy, A.; Nag, A. Multi-Factor Authentication. In *Advances in User Authentication*; Deb, K., Gupta, R., Mehlhorn, K., Rao, V.R., Sharma, A., Eds.; Springer: Cham, Switzerland, 2017. [CrossRef]
19. Bell, J. What Is Machine Learning? *Mach. Learn. City Appl. Archit. Urban. Des.* **2022**, 209–216.
20. Cohen, S. The Evolution of Machine Learning: Past, Present, and Future. *In Artif. Intell. Deep. Learn. Pathol.* **2021**, 1–12. [CrossRef]
21. Hafez, M.M.; Redondo, R.P.D.; Vilas, A.F. A Comparative Performance Study of Naïve and Ensemble Algorithms for E-Commerce. In Proceedings of the ICENCO 2018—14th International Computer Engineering Conference: Secure Smart Societies, Cairo, Egypt, 29–30 December 2018; pp. 26–31. [CrossRef]
22. Lomba, E.; Severino, R.; Vilas, A.F. Work In Progress: Towards Adaptive RF Fingerprint-Based Authentication of IIoT Devices. In Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation, ETFA, Stuttgart, Germany, 6–9 September 2022. [CrossRef]
23. Redondo, R.P.D.; Vilas, A.F.; Merino, M.R.; Rodríguez, S.M.V.; Guijarro, S.T.; Hafez, M.M. Anti-Sexism Alert System: Identification of Sexist Comments on Social Media Using AI Techniques. *Appl. Sci.* **2023**, *13*, 4341. [CrossRef]
24. González-Soto, M.; Díaz-Redondo, R.P.; Fernández-Veiga, M.; Fernández-Castro, B.; Fernández-Vilas, A. Decentralized and Collaborative Machine Learning Framework for IoT. *Comput. Netw.* **2024**, *239*, 110137. [CrossRef]
25. Malta, S.; Pinto, P.; Fernandez-Veiga, M. Using Reinforcement Learning to Reduce Energy Consumption of Ultra-Dense Networks With 5G Use Cases Requirements. *IEEE Access* **2023**, *11*, 5417–5428. [CrossRef]
26. Paladino, L.M.; Hughes, A.; Perera, A.; Topsakal, O.; Akinci, T.C. Evaluating the Performance of Automated Machine Learning (AutoML) Tools for Heart Disease Diagnosis and Prediction. *AI* **2023**, *4*, 1036–1058. [CrossRef]
27. Abumohsen, M.; Owda, A.Y.; Owda, M. Electrical Load Forecasting Based on Random Forest, XGBoost, and Linear Regression Algorithms. In Proceedings of the 2023 International Conference on Information Technology: Cybersecurity Challenges for Sustainable Cities, ICIT, Amman, Jordan, 9–10 August 2023; pp. 25–31. [CrossRef]
28. Owess, M.M.; Owda, A.Y.; Owda, M. Decision Support System in Healthcare for Predicting Blood Pressure Disorders. In Proceedings of the 2023 International Conference on Information Technology: Cybersecurity Challenges for Sustainable Cities, ICIT, Amman, Jordan, 9–10 August 2023; pp. 62–67. [CrossRef]
29. Kulatilleke, G.K.; Mary, Q. Challenges and Complexities in Machine Learning Based Credit Card Fraud Detection. *arXiv* **2022**, arXiv:2208.10943.
30. Gaikwad, J.R.; Deshmane, A.B.; Somavanshi, H.V.; Patil, S.V.; Badgujar, R.A. Credit Card Fraud Detection Using Decision Tree Induction Algorithm. *Int. J. Innov. Technol. Explor. Eng. (IJITEE)* **2014**, 2278–3075.
31. Ng, A.Y.; Jordan, M.I. On Discriminative vs. Generative Classifiers: A Comparison of Logistic Regression and Naive Bayes. *Adv. Neural Inf. Process Syst.* **2001**, *14*, 841–848.
32. Rajak, I.; Mathai, K.J. Intelligent Fraudulent Detection System Based SVM and Optimized by Danger Theory. In Proceedings of the IEEE International Conference on Computer Communication and Control, IC4 2015, Indore, India, 10–12 September 2015. [CrossRef]
33. Awoyemi, J.O.; Adetunmbi, A.O.; Oluwadare, S.A. Credit Card Fraud Detection Using Machine Learning Techniques: A Comparative Analysis. In Proceedings of the IEEE International Conference on Computing, Networking and Informatics, ICCNI 2017, Lagos, Nigeria, 29–31 October 2017; pp. 1–9. [CrossRef]
34. Khattri, V.; Singh, D.K. Implementation of an Additional Factor for Secure Authentication in Online Transactions. *J. Organ. Comput. Electron. Commer.* **2019**, *29*, 258–273. [CrossRef]
35. Sanyal, S.; Tiwari, A.; Sanyal, S. A Multifactor Secure Authentication System for Wireless Payment. *Adv. Inf. Knowl. Process.* **2010**, *53*, 341–369. [CrossRef]
36. Mohammed, M.M.; Elsadig, M. A Multi-Layer of Multi Factors Authentication Model for Online Banking Services. In Proceedings of the 2013 International Conference on Computer, Electrical and Electronics Engineering: "Research Makes a Difference", ICCEEE 2013, Khartoum, Sudan, 26–28 August 2013; pp. 220–224. [CrossRef]
37. Chetalam, L.J. Enhancing Security of Mpesa Transactions by Use of Voice Biometrics. *Diss. United States Int. Univ. Africa*. 2018. Available online: https://api.semanticscholar.org/CorpusID:69847257 (accessed on 7 December 2023).
38. MacIej, B.; Imed, E.F.; Kurkowski, M. Multifactor Authentication Protocol in a Mobile Environment. *IEEE Access* **2019**, *7*, 157185–157199. [CrossRef]
39. Guma, A. Development of a Secure Multi-Factor Authentication Algorithm for Mobile Money Applications. Ph.D. Thesis, NM-AIST, Arusha, Tanzania, 2022. [CrossRef]
40. Scaria, B.A.; Karman Megalingam, R. Enhanced E-Commerce Application Security Using Three-Factor Authentication. In Proceedings of the 2nd International Conference on Intelligent Computing and Control Systems, ICICCS 2018, Madurai, India, 14–15 June 2018; pp. 1588–1591. [CrossRef]

41. Hassan, M.A.; Shukur, Z. A Secure Multi Factor User Authentication Framework for Electronic Payment System. In Proceedings of the 2021 3rd International Cyber Resilience Conference, CRC 2021, Langkawi Island, Malaysia, 29–31 January 2021. [CrossRef]
42. Zadeh, M.J.; Barati, H. Security Improvement in Mobile Baking Using Hybrid Authentication. 2019 The 3rd International Conference on Advances in Artificial Intelligence, Istanbul, Turkey, 26–28 October 2019; 2019; pp. 198–201. [CrossRef]
43. Jaspher, G.; Kathrine, W.; Kirubakaran, E. Four-Factor Based Privacy Preserving Biometric Authentication and Authorization Scheme for Enhancing Grid Security. *Int. J. Comput. Appl.* **2011**, *30*, 975–8887.
44. Krol, K.; Philippou, E.; De Cristofaro, E.; Sasse, M.A. "They Brought in the Horrible Key Ring Thing!" Analysing the Usability of Two-Factor Authentication in UK Online Banking. *arXiv* **2015**, arXiv:1501.04434. [CrossRef]
45. Dutson, J.; Allen, D.; Eggett, D.; Seamons, K. Don't Punish All of Us: Measuring User Attitudes about Two-Factor Authentication. In Proceedings of the 4th IEEE European Symposium on Security and Privacy Workshops, EUROS and PW 2019, Stockholm, Sweden, 17–19 June 2019; pp. 119–128. [CrossRef]
46. Arnold, D.; Blackmon, B.; Gibson, B.; Moncivais, A.G.; Powell, G.B.; Skeen, M.; Thorson, M.K.; Wade, N.B. The Emotional Impact of Multi-Factor Authentication for University Students. In Proceedings of the Conference on Human Factors in Computing Systems, New Orleans LA, USA, 29 April–5 May 2022; 2022. [CrossRef]
47. Eid Alanzi, T.; Naif Alatawi, M. A Secure Two-Factor Authentication Framework Based on Deep Learning. *J. Res. Sci. Eng. (JRSE)* **2022**, *4*, 2319–7064. [CrossRef]
48. Kariapper, R. Attendance System Using RFID, IoT and Machine Learning: A Two Factor Verification Approach. *J. Adv. Res. Dyn. Control. Syst.* **2021**, *12*, 3285–3297. [CrossRef] [PubMed]
49. Deridder, Z.; Siddiqui, N.; Reither, T.; Dave, R.; Pelto, B.; Vanamala, M.; Seliya, N. Continuous User Authentication Using Machine Learning and Multi-Finger Mobile Touch Dynamics with a Novel Dataset. In Proceedings of the 2022 9th International Conference on Soft Computing and Machine Intelligence, ISCMI 2022, Toronto, ON, Canada, 26–27 November 2022; pp. 42–46. [CrossRef]
50. Misbahuddin, M.; Bindhumadhava, B.; Dheeptha, B. Design of a Risk Based Authentication System Using Machine Learning Techniques. In Proceedings of the 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), San Francisco, CA, USA, 4–8 August 2017; pp. 1–6.
51. Rodrigues, A.R.L. Enhanced Multi-Factor Authentication for Mobile Applications. *Enhanc. Multi-Factor. Authentication Mob. Appl.* 2023. Available online: https://estudogeral.uc.pt/handle/10316/107821 (accessed on 7 December 2023).
52. Maćkiewicz, A.; Ratajczak, W. Principal Components Analysis (PCA). *Comput. Geosci.* **1993**, *19*, 303–342. [CrossRef]
53. Taha, A.A.; Malebary, S.J. An Intelligent Approach to Credit Card Fraud Detection Using an Optimized Light Gradient Boosting Machine. *IEEE Access* **2020**, *8*, 25579–25587. [CrossRef]
54. Carcillo, F.; Le Borgne, Y.A.; Caelen, O.; Kessaci, Y.; Oblé, F.; Bontempi, G. Combining Unsupervised and Supervised Learning in Credit Card Fraud Detection. *Inf. Sci.* **2021**, *557*, 317–331. [CrossRef]
55. Bin Sulaiman, R.; Schetinin, V.; Sant, P. Review of Machine Learning Approach on Credit Card Fraud Detection. *Hum.-Centric Intell. Syst.* **2022**, *2*, 55–68. [CrossRef]
56. Tiwari, P.; Mehta, S.; Sakhuja, N.; Kumar, J.; Singh, A.K. Credit Card Fraud Detection Using Machine Learning: A Study. *arXiv* **2021**, arXiv:2108.10005.
57. Aburbeian, A.M.; Ashqar, H.I. Credit Card Fraud Detection Using Enhanced Random Forest Classifier for Imbalanced Data. In Proceedings of the 2023 International Conference on Advances in Computing Research (ACR'23), Springer Links, Orlando, FL, USA, 25–26 May 2023; pp. 605–616.
58. Vargaftik, S.; Keslassy, I.; Orda, A.; Ben-Itzhak, Y. RADE: Resource-Efficient Supervised Anomaly Detection Using Decision Tree-Based Ensemble Methods. *Mach. Learn.* **2021**, *110*, 2835–2866. [CrossRef]
59. Itoo, F.; Meenakshi; Singh, S. Comparison and Analysis of Logistic Regression, Naïve Bayes and KNN Machine Learning Algorithms for Credit Card Fraud Detection. *Int. J. Inf. Technol.* **2021**, *13*, 1503–1511. [CrossRef]
60. Kaur, S.; Kaur, G.; Shabaz, M. A Secure Two-Factor Authentication Framework in Cloud Computing. *Secur. Commun. Netw.* **2022**, *2022*, 7540891. [CrossRef]
61. Adjabi, I.; Ouahabi, A.; Benzaoui, A.; Taleb-Ahmed, A. Past, Present, and Future of Face Recognition: A Review. *Electronics* **2020**, *9*, 1188. [CrossRef]
62. Muda, T.Z.T.; Wong, X.Y.; Teh, X.L. Designing a Mobile Apps: Savior. *J. Inf. Syst. Technol. Manag.* **2022**, *7*, 178–185. [CrossRef]
63. Zeng, G. On the Confusion Matrix in Credit Scoring and Its Analytical Properties. *Commun. Stat. Theory Methods* **2020**, *49*, 2080–2093. [CrossRef]
64. Davis, J.; Goadrich, M. The Relationship between Precision-Recall and ROC Curves. *ACM Int. Conf. Proceeding Ser.* **2006**, *148*, 233–240. [CrossRef]
65. Zhang, Y.; Yu, G.; Yang, D. Predicting Non-Performing Loan of Business Bank by Multiple Classifier Fusion Algorithms. *J. Interdiscip. Math.* **2016**, *19*, 657–667. [CrossRef]
66. Dighe, D.; Patil, S.; Kokate, S. Detection of Credit Card Fraud Transactions Using Machine Learning Algorithms and Neural Networks: A Comparative Study. In Proceedings of the 2018 4th International Conference on Computing, Communication Control and Automation, ICCUBEA 2018, Pune, India, 16–18 August 2018. [CrossRef]

67. Dhankhad, S.; Mohammed, E.A.; Far, B. Supervised Machine Learning Algorithms for Credit Card Fraudulent Transaction Detection: A Comparative Study. In Proceedings of the 2018 IEEE 19th International Conference on Information Reuse and Integration for Data Science, IRI 2018, Salt Lake City, UT, USA, 6–9 July 2018; pp. 122–125. [CrossRef]
68. Dornadula, V.N.; Geetha, S. Credit Card Fraud Detection Using Machine Learning Algorithms. *Procedia Comput. Sci.* **2019**, *165*, 631–641. [CrossRef]
69. Kumar Trivedi, N.; Simaiya, S.; Kumar Sharma, S.; Kumar Lilhore, U. An Efficient Credit Card Fraud Detection Model Based on Machine Learning Methods. *Int. J. Adv. Sci. Technol.* **2020**, *29*, 3414–3424.
70. Faraji, Z. A Review of Machine Learning Applications for Credit Card Fraud Detection with A Case Study. *SEISENSE J. Manag.* **2022**, *5*, 49–59. [CrossRef]
71. Mohsen, O.R.; Nassreddine, G.; Massoud, M. Credit Card Fraud Detector Based on Machine Learning Techniques. *J. Comput. Sci. Technol. Stud.* **2023**, *5*, 16–30. [CrossRef]
72. Homepage, J.; Unogwu, O.J.; Filali, Y. Fraud Detection and Identification in Credit Card Based on Machine Learning Techniques. *Wasit J. Comput. Math. Sci.* **2023**, *2*, 16–22. [CrossRef]
73. Aftab, A.U.; Shahzad, I.; Anwar, M.; Sajid, A.; Anwar, N. Fraud Detection of Credit Cards Using Supervised Machine Learning. *Pak. J. Emerg. Sci. Technol. (PJEST)* **2023**, *4*, 38–51. [CrossRef]
74. Ogbanufe, O.M.; Baham, C. Using Multi-Factor Authentication for Online Account Security: Examining the Influence of Anticipated Regret. *Inf. Syst. Front.* **2023**, *25*, 897–916. [CrossRef]
75. Tanwar, R.; Samadi, B.; Khodadadi, T.; Chaudet, C.; Bellavista, P.; Prakash Otta, S.; Panda, S.; Gupta, M.; Hota, C. A Systematic Survey of Multi-Factor Authentication for Cloud Infrastructure. *Future Internet* **2023**, *15*, 146. [CrossRef]
76. Braeken, A. Highly Efficient Bidirectional Multi-Factor Authentication and Key Agreement for Real-Time Access to Sensor Data. *IEEE Internet Things J.* **2023**, *10*, 21089–21099. [CrossRef]
77. Marasco, E.; Albanese, M.; Vamsi, V.; Patibandla, R.; Vurity, A.; Sriram, S.S. Biometric Multi-Factor Authentication: On the Usability of the FingerPIN Scheme. *Secur. Priv.* **2023**, *6*, e261. [CrossRef]
78. Ahmad, M.O.; Tripathi, G.; Siddiqui, F.; Alam, M.A.; Ahad, M.A.; Akhtar, M.M.; Casalino, G. BAuth-ZKP—A Blockchain-Based Multi-Factor Authentication Mechanism for Securing Smart Cities. *Sensors* **2023**, *23*, 2757. [CrossRef]
79. Carrillo-Torres, D.; Pérez-Díaz, J.A.; Cantoral-Ceballos, J.A.; Vargas-Rosales, C. A Novel Multi-Factor Authentication Algorithm Based on Image Recognition and User Established Relations. *Appl. Sci.* **2023**, *13*, 1374. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.