Fraud Detection in Credit Cards using Logistic Regression

Hala Z Alenzi[1], Nojood O Aljehane[2]
Department of Computer Science
Tabuk University, Tabuk City
Kingdom Saudi Arabia

***Abstract*—Due to the increasing number of customers as well as the increasing number of companies that use credit cards for ending financial transactions, the number of fraud cases has increased dramatically. Dealing with noisy and imbalanced data, as well as with outliers, has accentuated this problem. In this work, fraud detection using artificial intelligence is proposed. The proposed system uses logistic regression to build the classifier to prevent frauds in credit card transactions. To handle dirty data and to ensure a high degree of detection accuracy, a pre-processing step is used. The pre-processing step uses two novel main methods to clean the data: the mean-based method and the clustering-based method. Compared to two well-known classifiers, the support vector machine classifier and voting classifier, the proposed classifier shows better results in terms of accuracy, sensitivity, and error rate.**

Keywords—Classifier; logistic regression; accuracy; smoothing; artificial intelligence; cross validation

I. Introduction

According to the definition of fraud [1], the aim of fraud is to achieve personal or financial gain through deception. Based on this, fraud detection and prevention are the two significant methods for avoiding loss due to fraud. Fraud prevention is the proactive technique for avoiding the occurrence of fraudulent acts, and fraud detection is the technique for the detection of fraudulent transactions by fraudsters [2]. A variety of payment cards, including credit, charge, debit, and prepaid cards, are currently widely available. They are the most popular means of payment in some countries [3]. Indeed, advances in digital technologies have paved the way for changes in how we handle money, especially for payment methods that have changed from being a physical activity to a digital activity using electronics means [4]. This has revolutionized the landscape of monetary policy, including the business strategies and operations of both large and small companies. Credit card fraud is the fraudulent use of credit card details to buy a product or service. These transactions can be physically or digitally performed [5]. In physical transactions, the credit card is physically present. On the other hand, digital transactions take place over the internet or telephone. A cardholder normally provides their card number, card verification number, and expiration date through a website or telephone call. With the rapid rise in e-commerce over the past few years, credit card use has increased tremendously [1,3].

In Malaysia, the number of transactions performed through credit cards in 2011 was approximately 317 million, and this number increased to 447 million in 2018 [4]. In 2015, global credit card fraud reached a record of $21.84 billion, as reported by [2]. The number of fraud cases has been rising with the increased use of credit cards. While various verification methods have been implemented, the number of fraud cases involving credit cards has not been significantly decreased [6]. The potential for substantial monetary gains, combined with the ever-changing nature of financial services, creates a wide range of opportunities for fraudsters [7]. Funds from payment card fraud are often used in criminal activities that are hard to prevent, e.g., to support terrorist acts [8]. The internet is where fraudsters prefer to be because they are able to conceal their location and identity. The recent increase in credit card fraud has directly hit the financial sector hard. Losses due to credit card fraud mainly impact merchants because they bear all expenses, including the fees from their card issuer, administrative fees and other charges [9]. All the losses are borne by the merchants, leading to increases in the prices of goods and decreases in discounts. Hence, reducing this loss is highly important. An effective fraud detection system is required to minimize the number of cases of fraud.

A. Motivation

The use of credit cards to perform financial transactions at banks or other institutions is a common action in light of the currently available technology. Online payments (or any other online transactions) bring benefits to companies and individuals in terms of the convenience, velocity, and flexibility of performing daily duties [10,11]. The work in [12] presented a statistical analysis related to the usage of credit cards over five years (from 2006 to 2010). This reflected the huge dependency on credit cards by both people and organizations. To take advantage of advanced technologies, companies try to use advanced techniques to provide high-quality services to customers. Automation can be seen as the best solution for attracting more customers and consequently collecting more financial gain [13]. The process of converting a manual system to a fully automatic on, as found in smart cities, is not without risk.

B. Problem Statement

According to [14], it is estimated that 10,000 transactions take place via credit cards every second worldwide. Owing to such a high transaction frequency, credit cards have become the primary targets of fraud. Indeed, since the Diners Club released its first credit card in 1950, credit card companies have been fighting against fraud [15]. Every year, billions of dollars are lost directly because of credit card fraud. Fraud cases occur under different conditions, e.g., transactions at points of sale (POSs) or transactions made online or over the telephone, i.e., card-not-present (CNP) cases or transactions with lost and stolen cards. In this way, the credit card fraud in 2015 alone amounted to $21.84 billion, with issuers bearing $15.72 billion of the cost [16]. Based on information from the European Central Bank, in 2012, the majority (60%) of fraud stemmed from CNP transactions, and another 23% stemmed from POS terminals. The value of fraud is high globally and locally in Malaysia. The volumes of credit, debit, and charge cards were 383.8 million, 107.6 million, and 4.1 million, respectively, in 2016 and increased to 447.1 million, 245.7 million, and 5.2 million, respectively, in 2018 [9]. The overall percentage of fraudulent payments (i.e., with credit, debit, and charge cards) was 0.0186% in 2016 and increased by 37.6% to 0.0256% in 2018 [17]. The potential for huge monetary gains combined with the ever-changing nature of financial services provides opportunities for fraudsters. In Malaysia, 1,000 card transactions occur every minute. Fraud directly impacts merchants and financial institutions because they incur all the costs. An increase in fraud affects customers' confidence in using electronic payments [18].

Many surveys have shown that the increase in the dependence on credit cards to perform financial transactions is accompanied by an increasing rate of fraud, as seen in [1,3]. The increasing capabilities of the attackers or the hackers have accentuated the problem since these people can exploit security gaps to obtain sensitive information about users or their credit information to perform malicious activities, such as fraud [4,5]. To define this problem accurately, Fig. 1 shows the general scenario of performing credit card fraud.

As shown in Fig. 1, the attacker can perform malicious activities on many sides of the online process. To solve this problem, a fraud detection system is needed. Artificial intelligence (AI) is defined as the research field that aims at performing machine learning to obtain an intelligent machine that can perform tasks on behalf of the user. This can be done through two main steps: training and testing. AI is employed to build systems for fraud detection, such as classification-based systems [19,6,7,8], clustering-based systems [17,20,21], neural network-based systems [18,22,23], and support vector machine-based systems [9]. Although AI-based systems can perform well, they suffer from some critical issues. First, the term "imbalanced data" refers to unbalanced data used for training, where one class of the data is dominated by the other (i.e., the majority of data belong to one class and the rest belong to the other). This negatively affects the accuracy of detection [24,25]. Second, the term "noisy data" refers to the existence of outliers within the data employed for training. Outliers can be seen outside of the normal context of the data. This issue also leads to poor detection accuracy [26,16]. Third, the concept of drift means that the behaviour of the client changes, resulting in changes in the data stream when dealing with online data detection in real time [15,14].

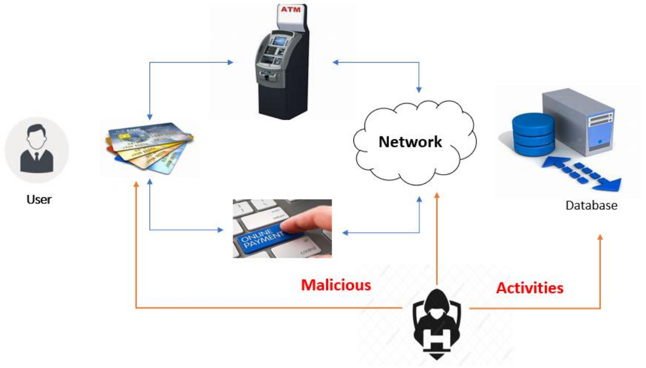


Fig. 1. General Scenario of Online Fraud.

C. Research Questions

On the basis of the empirical evidence, the following research questions are developed to guide this study and meet its objectives.

- How can a fraud detection system be built using AI that can deal with imbalanced data effectively?
- How can we smooth (or clean) the data before using it for training the machine to ensure high detection accuracy?
- How can the system detect fraud by adapting to the behaviour of the user?

D. Contributions

The contributions of this work can be summarized as follows:

- An AI-based system for fraud detection is proposed. The system uses logistic regression to build a classifier called the LogR classifier. The LogR classifier has the ability to deal with imbalanced data and adapt to the behaviour of the user by employing the cross-validation technique.
- To ensure high accuracy detection, two main methods are used to clean the data. The mean-based method deals with missing values, and the clustering-based method deals with outliers.
- Extensive experiments are conducted to train and test the proposed classifier using a standard database.

E. Structure of the Paper

The rest of this work is organized as follows. Section II reviews the related work. Section III describes the proposed artificial intelligence system in detail. In Section IV, the metrics used are presented for evaluation purposes. Section V presents the experiments and discusses the results in light of a comparison with similar approaches. Finally, the paper is concluded in Section VI.

II. Related Work

This section first provides a brief background about the research domain. Then, the related work is presented in detail.

A. Background

The background refers to the credit card research field in terms of the intersection of multiple research sectors. This field can be viewed as the intersection of four main domains, as illustrated in Fig. 2.

The definitions of the domains and terms that are applied in this study are listed below.

Artificial Intelligence (AI): It can be defined as the science that addresses the methods used for training machines to mimic the brains of humans. In other words, machines can be used to make decisions on behalf of human users. In this context, data mining tasks, such as classification, clustering, applying association rules, and using neural networks, are employed [2].

Financial Systems: These can be defined as the systems that are used to convert manual transactions into digital transactions. In this context, the term "transaction" denotes any financial activity that may be performed by a user based on a specific system [27].

Chip Industry: This term refers to the manufacturing of chips to store critical information on the card of the user. The information acts as a key to trigger any transaction. However, the chip is programmed to match some passwords to allow access to financial interfaces [28].

Internet of Things (IoT): It can be defined as a collection of devices connected via a network. The devices vary from small devices with low processing power (such as watches) to large devices high processing power, such as mobile devices. Using IoT devices to perform financial transactions is vital, especially in light of the goal of shifting toward smart cities [29].

B. Groups of AI-based Techniques

Artificial intelligence (AI) is defined as enabling machines to make decisions on behalf of human users. In this context, data mining tasks, such as classification, clustering, applying association rules, and using neural networks, are employed [2]. In addition, AI is employed to build systems for fraud detection, such as classification-based systems [19,6,7,8], clustering-based systems [17,20,21], neural network-based systems [18,22,23] and support vector machine-based systems [9].

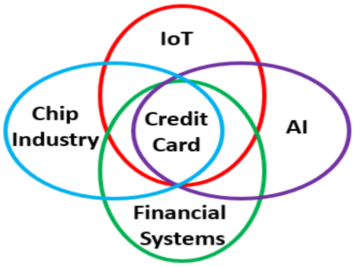


Fig. 2. The Intersection of Credit Card Research and other Research Fields.

The techniques employed to construct credit card fraud detection systems using AI can be categorized into four main groups. This idea is shown in Fig. 3.

1) Classification-based systems: The authors in [19] tried to achieve two main objectives in their work: (1) enhancing the accuracy of the classifications output by credit card fraud detection systems and (2) lowering the response times of these systems. To achieve the first goal, the authors proposed a hybrid model that fuses two classifiers to generate a new (or enhanced) one. The first classifier used is the K-means classifier, which deals with overlapping data because such data cause poor accuracy. The second classifier is the artificial bee colony algorithm (ABC), which is used to enhance the performance of the system. The first classifier forms the first level, and the second classifier forms the second level of the classification process proposed in the same model. The database used in this work was generated by using the C# programming language, where the number of instances was 100,000. In addition, 12 features were selected to include in the training phase. The selected features were based on a rule engine.

Moreover, previous systems suffered from problems in real-time environments [6]. These are problems in the context of credit card fraud detection. Such problems include imbalanced data, noisy data, and the concept of drift. The authors applied the bag creation technique to solve the data problems; this technique involves performing the sampling process on the collected data in real time. To clean the data, they applied naïve Bayes networks for the effective manipulation of noisy data. An incremental learning-based method was presented to address the concept of drift. The data set used in this work is summarized in Table I.

The strength of this study is the enhancement in performance achieved by using Spark to implement the system in parallel. In addition, the reduction in cost is considered an important feature of this system, and this was achieved by employing naive Bayes networks in the process of classification. The weakness of the proposed system is that it does not manipulate cyclic recurrences that may be included in the concept of drift. Cyclic recurrences refer to cyclic repetitions in the distributions of data.

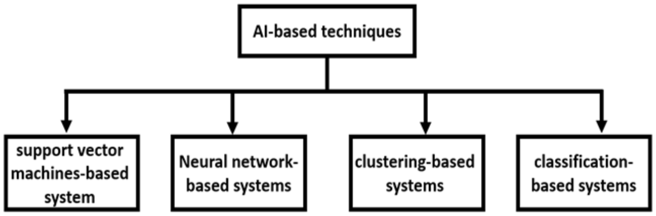

AI-based techniques
support vector machines-based system
Neural network-based systems
clustering-based systems
classification-based systems

Fig. 3. Categories of AI-based Techniques for Fraud Detection.

TABLE I. USED DATA SET [6]

Start day	End day	Instances	%Fraudulent Transactions
July 2004	September 2004	0.3 million	3.74%

The authors in [7] evaluated the current fraud detection system with regard to credit card transactions. The problem is that there are two stages for automatic classification: real-time (RT) and near-real-time (NRT). They focused on the NRT stage by using a rule-based classification technique that considers the final evaluation of the human element of fraud. The authors did not improve the design of the system, discover any new rules, or improve the arithmetic efficiency of individual rules. Instead, they manipulated the rules to form a decision-making system to improve both the accuracy and the performance. The key idea is to calculate the contribution of each rule involved in the system. Calculating the contribution of a rule depends on the difference between two values, which are (1) the performance of the system when the rule is used and (2) the performance of the system without using the rule. The degree of performance improvement is high if the rule is not redundant and is low if it is redundant with other rules or rule groups. For the measurement of performance, the precision, recall and F-score metrics were employed. A real database, which consists of 359,862 records provided by some industrial partners, was used for the training phase.

The authors in [8] addressed credit card fraud detection. In this study, the authors relied on the fact that "the features of the financial transactions in institutions change over time". This shows that the problem of credit card fraud detection should be considered in real time. Therefore, they converted this problem into real working transactions. In terms of artificial intelligence, the class should not be provided to the classifier immediately during the training stage. The key idea of the proposed approach is to follow a strict strategy that has three main steps: (1) analysing the real conditions under which the real transactions are performed; (2) employing these conditions to train the classifier using two main data sets; and (3) testing the classifier after the training stage is completed and supporting it by using the feedback of the users (their interactions) to improve the accuracy of the classifier. Table II summarizes the dataset used.

2) Clustering-based systems: To address the problem of detecting credit card fraud through transactions, the authors in [17] dealt with the problem of online shopping fraud and the concept of drift. They proposed a strategy consisting of four stages: (1) based on both the previous transaction data and the information of the cardholders, they used the clustering method to divide the cardholders into different groups for the purpose comparing their behaviours; (2) they proposed a sliding window strategy to group the transactions in each group to extract the behavioural patterns for each cardholder; (3) they trained a set of classifications for each group to measure behavioural patterns; and (4) they used a group of classifiers by training them on cardholder behaviours and output the highest behaviour pattern. A feedback mechanism was used to solve the concept of drift problem. Four dataset simulators were generated to manually create the data sets.

The authors in [20] proposed a clustering-based method. In this study, the fraud detection problem in ecommerce is manipulated and may be exploited by hackers who are highly skilled. The methods proposed to address such problems suffer from low accuracy and effectiveness. In addition, the methods used for detecting fraud may make some mistakes in identifying fraudulent transactions. The reason behind such shortcomings is that the proposed approaches focus on order analysis rather than anything else. Motivated by these facts, the authors proposed a method that focuses on the hackers themselves. The key idea is to extract some recognized features, such as the address of delivery, customer name, and methods of payment, and then, based on these features, the similarity among the attackers is calculated. Based on these similarities, the attackers are grouped in some clusters for detection. A main feature of their proposed method is that two current methods, agglomerative clustering and sampling, are selectively used in a reasonable amount of time for recursively grouping orders into small clusters. The dataset used for the training process was inspired by the Zalando website. This website periodically receives approximately 29 million orders (some of them are normal and others are fraudulent).

The authors in [21] tried to evaluate the detection problem by extracting the general pattern of the dataset to represent the fraud. In other words, the enhancement of the clustering methods relies only on the clusters used; this technique is called general enhancement. The authors proposed an approach that enables the application of local enhancement as well as general enhancement for fraud detection in financial transactions. They proposed the “Hierarchical Clusters-based Deep Neural Networks (HC-DNN)” method that uses the anomalous features of hierarchical clusters that are pretrained based on an autoencoder as the initial weights for neural networks. In detail, the data are grouped based on abnormal features that refer to fraud. These features are then used as the initial weights for the input layers of neural networks, as shown in Fig. 4.

TABLE II. DETAILS OF THE DATASET USED [8]

Id	End day	Instances	Features	%Fraudulent Transactions
2013	2014-01-18	21'830'330	51	0.19%
2014-2015	2015-05-31	54'764'384	51	0.24%

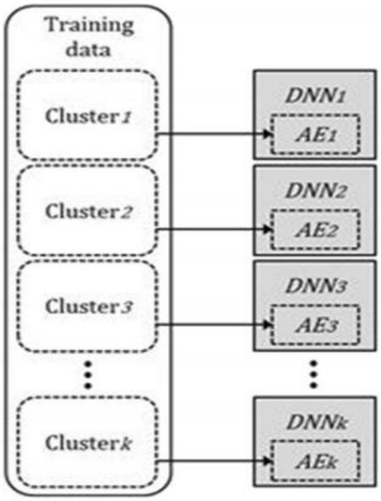


Fig. 4. Key Idea of the HC-DNN Method [21].

The authors used a dataset containing 19,505 records, including fraudulent and non-fraudulent records. The dataset is skewed and consists of 19,313 non-fraudulent and 192 fraudulent cases. Some preprocessing steps were performed on the data to mitigate the negative impact of the imbalanced data before using them for actual training.

3) Neural network-based systems: The authors in [18] discussed issues related to increasing fraud detection in online shopping transactions and payments, especially those related to credit cards. To detect credit card fraud, they proposed a neural network-based system. It uses back prorogation to enhance the output of the neural network so that the error (the difference between the actual or desirable value and the output of the neural network) is distributed back by adjusting the weights of the inputs. The strategy followed in this work can be summarized through the following steps:

a) A new Neuroph Project was created in Neuroph Studio using the Java programming language.

b) The actual perceptron network was constructed.

c) The training data set was prepared.

d) The training process was started by considering the desired value (the accuracy of fraud detection) set by an expert in the field.

e) The trained network was tested.

The data used for training were collected from a data mining blog. It includes 20000 active credit card holders with transactions spanning more than six months. The authors in [22] proposed a "Convectional Neural Network CNN" in their work. Similar to previous works, the problem studied was how to detect a pattern that represents fraudulent transactions. In their method, the CNN forms a classifier that takes features of the transactions as inputs. The features are extracted from each transaction and stored in a feature matrix. The classifier has the ability to deal with imbalanced data based on the sampling technique. The key idea behind the sampling technique is to use higher than normal costs to generate fraudulent transactions. Fig. 5 illustrates the general scenario of the CNN model.

The data used includes more than 260 million credit card transactions in one year. Approximately four thousand transfers are listed as fraudulent, and the remainder are legal. A hybrid fraud detection system was proposed in [23]. The key idea is to use neural networks as classifiers. Since the network needs to update the weights of the input layer, a swarm optimization method was employed for this purpose. Finally, the model was tested and evaluated. Fig. 6 illustrates the general structure of the proposed system, which is called the "Particle Swarm Optimization Auto-associative Neural Network (PSOAANN)".

4) Support vector machine-based systems: The authors in [9] used a support vector machine (SVM) to improve the accuracy of the classifier in the process of detecting fraud in credit card transactions. The key idea behind using an SVM is to split the features that represent transactions, where these features are used for the clustering process. In other words, the data are cleaned initially. Then, the features of transactions are extracted. Third, the features are measured to calculate the similarity among them. To isolate the features as much as possible, the SVM is used. Fourth, the K-means clustering algorithm is used to cluster the data based on the isolated (i.e., as far as possible) features. The classifier is then trained on the clusters. The classifier that deals with fraudulent transactions is used to detect fraud. The database used for training contains 5310 records in total. Among them, 490 records are fraudulent data and 4820 are non-fraudulent data, and 1174 characteristic variables are included.

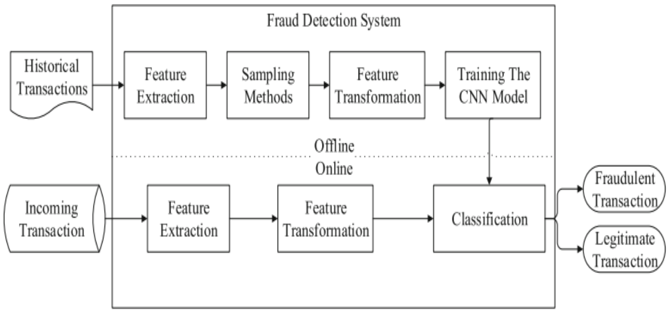


Fig. 5. General Scenario of the Fraud Detection System Proposed in the Work in [22].

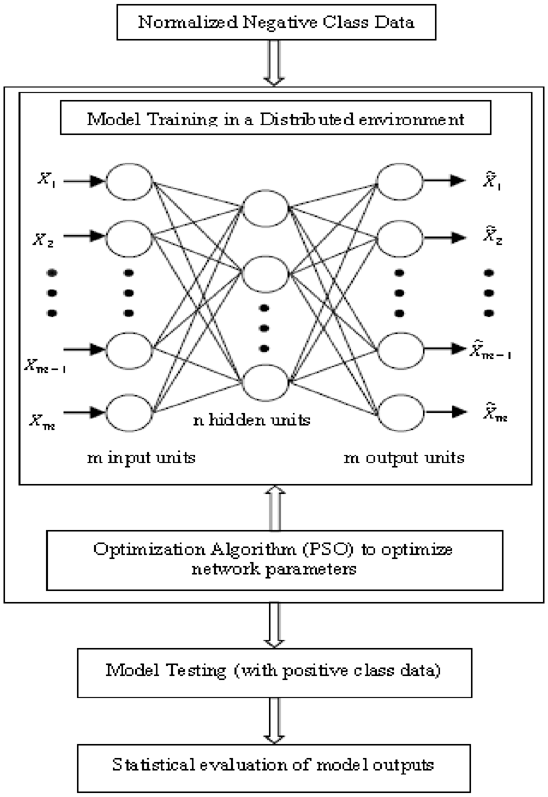

Normalized Negative Class Data
Model Training in a Distributed environment
X_1
X_2
X_{m-1}
X_m
$\hat{X}_1$
$\hat{X}_2$
$\hat{X}_{m-1}$
$\hat{X}_m$
n hidden units
m input units
m output units
Optimization Algorithm (PSO) to optimize network parameters
Model Testing (with positive class data)
Statistical evaluation of model outputs

Fig. 6. Structure of the PSOAANN-based System [23].

III. Proposed Approach

This section describes the proposed approach in detail. Fig. 7 illustrates the steps of the proposed approach.

As shown in Fig. 7 above, there are nine steps, starting with the selection of the database and ending with the use of the classifier in real-life situations. The reason behind selecting logistic regression to build the classifier is related to its efficiency of detecting frauds based on its ability to isolate the data that belong to different binary classes.

A. Selecting the Database

This work uses a standard dataset that is available on the internet [30]. The dataset contains transactions made using credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred over two days, where we have 492 fraudulent cases out of 284,807 transactions. The dataset is highly unbalanced, and the positive class (fraudulent cases) accounts for 0.172% of all transactions. Fig. 8 shows the selection step in the implemented programme represented by "Load DB".

As shown in Fig. 8, the loading of the data is competed, and the size of the dataset can be seen.

To explore the data contained in this data set, Fig. 9 shows the data exploration options that can be chosen.

As shown in Fig. 9, there are 6 views of the used data set. This enables us to clearly explore the database. In terms of exploring the database, Fig. 10 and 11 show two examples of data exploration.

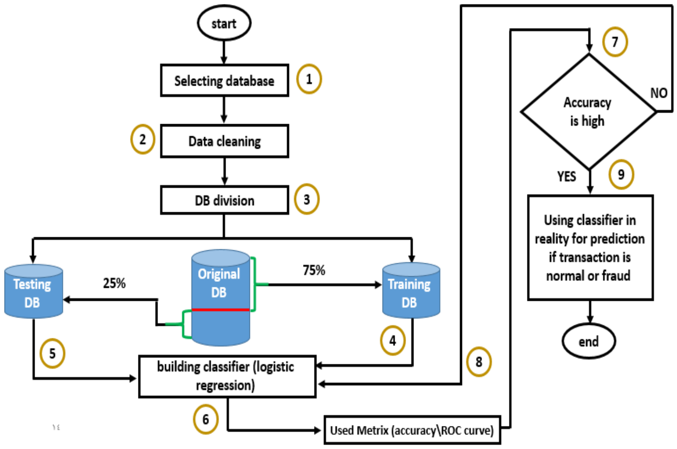
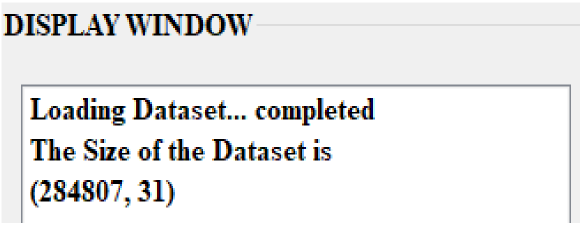

start
Selecting database
1
2
Data cleaning
DB division
3
Testing DB
25%
Original DB
75%
Training DB
4
5
building classifier (logistic regression)
8
6
Used Metrix (accuracy\ROC curve)
7
Accuracy is high
NO
YES
9
Using classifier in reality for prediction if transaction is normal or fraud
end

Fig. 7. Flow Chart of the Proposed Approach.



DISPLAY WINDOW

Loading Dataset... completed
The Size of the Dataset is
(284807, 31)

Fig. 8. Loading the used Dataset.

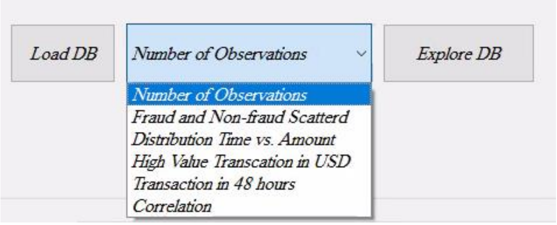


Fig. 9. Interface for Selecting (or Loading) the Data Set.

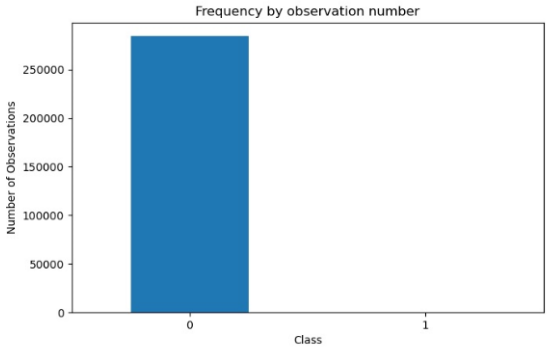


Fig. 10. Data Exploration based on the Observation Number.

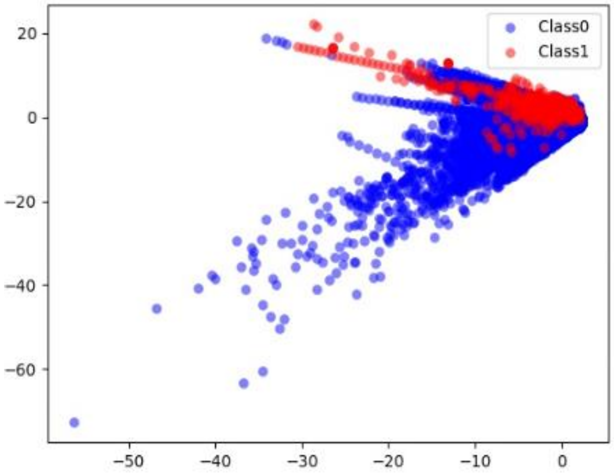


Fig. 11. Data Exploration based on the Two Main Classes of the Data.

B. Data Cleaning

The goal of this step is to clean the data and prepare it for the training phase of the classifier. In general, data in reality are noisy. Therefore, a cleaning step is necessary. In the context of the data cleaning process, the procedure is as follows:

1) Fill in the missing values. A missing value means that a cell of a given record is empty due to an mistake during entry.

2) Solve any inconsistencies. This means that if there is a collision in the data, this collision must be resolved.

3) Remove any outliers. Outliers refer to abnormal values (i.e., very high values or very low values).

Fortunately, most of the data used in the data set are cleaned except for some missing values and outliers. The mechanism that is used for handling the missing values depends on the mean (mathematical operation) since the data are numbers. Fig. 12 illustrates to the process of filling in a missing value.

For the handling of outliers, a clustering-based method is employed in this work. The key idea is to create three clusters (one for the normal data, a second one for high values, and a third for low values). After grouping the data into the clusters, the last two clusters (i.e., those that contain outliers) are deleted. Fig. 13 illustrates the mechanism of outlier removal.

C. Database Division

In this step, the database is divided into training and testing databases. The goal of the training database is to construct the classifier (model), while the goal of the testing database is to test (evaluate) the built classifier. In this work, the cross-validation method is used to divide the database, which is divided into 10 parts, as shown in Fig. 14.

As shown in Fig. 14, the database is divided into 10 parts (i.e., the value of $k = 10$ in the cross-validation method). In the first iteration ($k = 1$), the first nine parts are considered a training set, while the last part of the database is considered a testing set. In the second iteration ($k = 2$), both the first eight parts and the tenth part are considered as a training set, while the ninth part of the database is considered a testing set. This process continues until the last iteration ($k = 10$), where the first part is the testing set and the last nine parts are the training set.

Fig. 15 illustrates a sample of the code execution process based on the cross-validation method when clicking on the “Split DB” button.

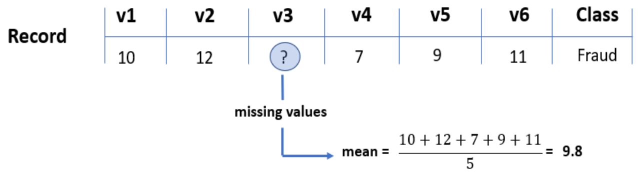


Fig. 12. Mean-based Mechanism for Handling Missing Values.

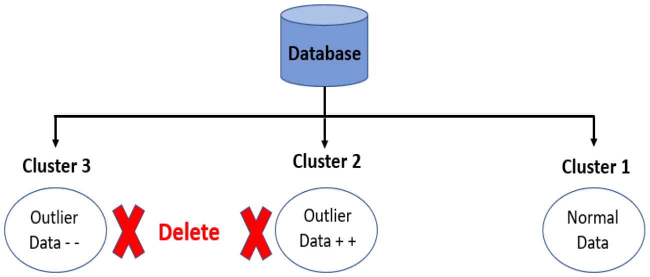


Fig. 13. Mechanism of Outlier Removal.

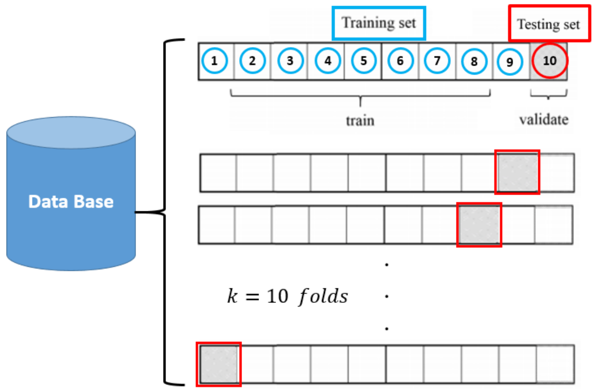
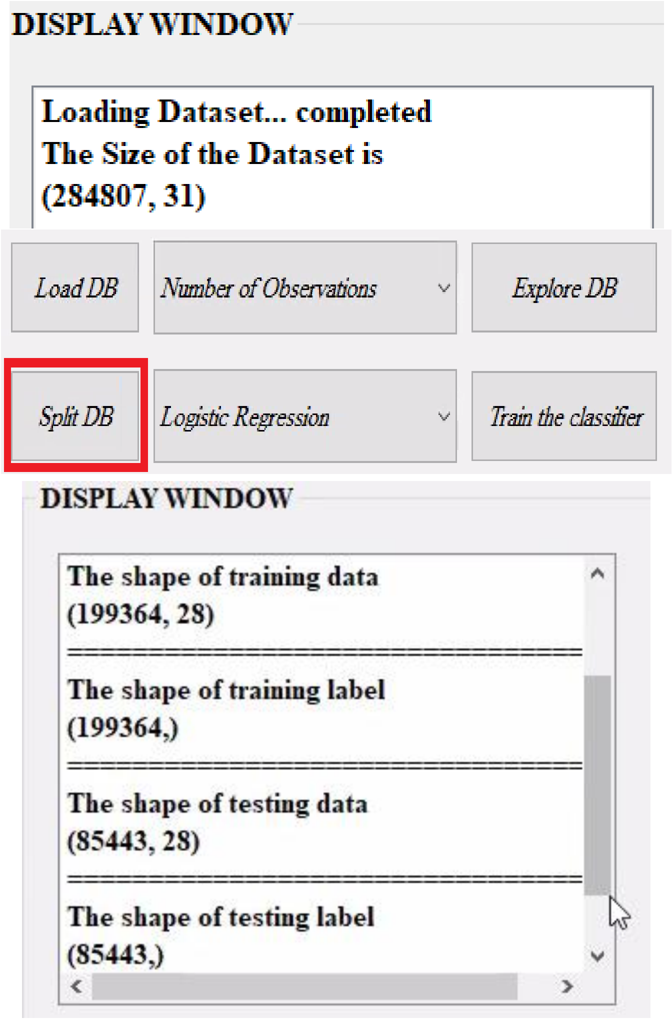


Fig. 14. Division of the Database based on Cross Validation.



```
Loading Dataset... completed
The Size of the Dataset is
(284807, 31)
```

```
The shape of training data
(199364, 28)
========================================
The shape of training label
(199364,)
========================================
The shape of testing data
(85443, 28)
========================================
The shape of testing label
(85443,)
```

Fig. 15. Results of the Division Process.

D. Building the Classifier

In the context of building the classifier, logistic regression is employed. Logistic regression is more advanced than linear regression. The reason for this is that linear regression cannot classify data that are widely distributed in a given space, as shown in Fig. 16.

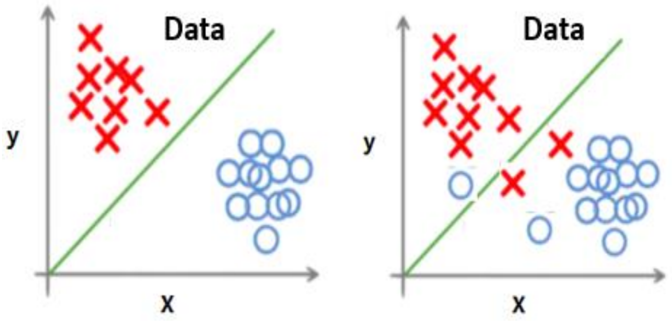


Fig. 16. The Limitation of Linear Regression.

As shown in Fig. 16, on the left side, the linear regression has the ability to classify the data, where the line can divide the given data into two main categories (or classes). The right side of Fig. 16 illustrates the limitation of linear regression. When the data overlap, the line cannot divide the data into two clear classes. This limitation is overcome by logistic regression. Fig. 17 provides a visual comparison between the linear regression and the logistic regression methods for the purpose of highlighting this limitation.

Logistic regression has the following advantages [32]:

1) Logistic regression is easier to implement than linear regression and is very efficient to train.

2) It makes no assumptions about the distributions of classes in the feature space.

3) It can easily be extended to multiple classes (multinomial regression).

4) It is very efficient for classifying unknown records.

The logistic regression equation can be obtained from the linear regression equation. The mathematical steps to obtain logistic regression equations are given below:

The equation of the straight line can be written as:

$$y = a_0 + a_1 \times x_1 + a_2 \times x_2 + \cdots a_k \times x_k \qquad (1)$$

In logistic regression, y can be between 0 and 1 only, so we divide the above equation by $(1 - y)$:

$$\frac{y}{1-y} \; |0 \; for \; y = 0 \; and \; \infty \; for \; y = 1 \qquad (2)$$

As a result, the logistic regression equation is defined as:

$$\log \left[\frac{y}{1-y}\right] = a_0 + a_1 \times x_1 + a_2 \times x_2 + \cdots a_k \times x_k \qquad (3)$$

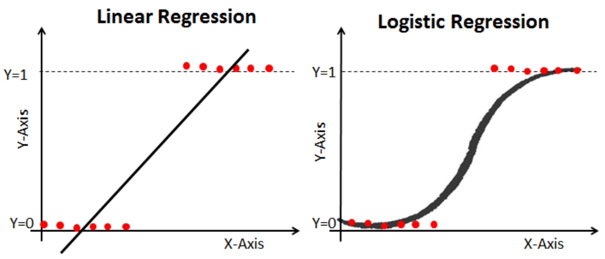


Fig. 17. A Visual Comparison between Linear and Logistic Regression [31].

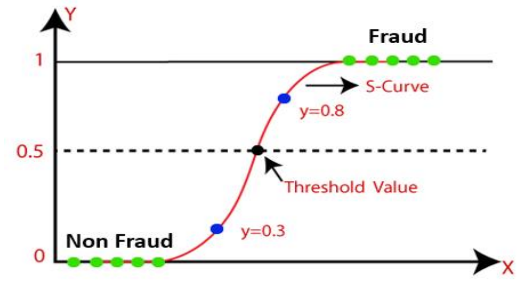


Fig. 18. The Concept of Logistic Regression Classification [33].

In other words, the fraud class takes the value “1”, while the non-fraud class takes the value “0”. A threshold of 0.5 is used to differentiate between the two classes, as shown in Fig. 18.

E. Testing the Classifier

Since the cross-validation method divides the database into 10 parts, there are 10 testing data sets. Each testing data set is used to test one classifier (there are 10 classifiers). This in turn gives the model an advantage by allowing it to use the whole database for testing as well as for training. The testing process is tightly coupled with the accuracy of the model. Calculating the final accuracy involves calculating the accuracy of each classifier. Formally, let Acc_k^C denote the accuracy of a given trained classifier, as shown in Fig. 19.

Then, the final accuracy of the final classifier (ACC_F^C) is obtained based on the “average” mathematical operation.

$$ACC_F^C = \frac{\sum_{k=1}^{10} Acc_k^C}{k} \qquad (4)$$

F. Evaluating the Classifier

In general, a confusion matrix is an effective benchmark for analysing how well a classifier can recognize records of different classes [34]. The confusion matrix is formed based on the following terms:

1) True positives (TP): positive records that are correctly labelled by the classifier.

2) True negatives (TN): negative records that are correctly labelled by the classifier.

3) False positives (FP): negative records that are incorrectly labelled positive.

4) False negatives (FN): positive records that are mislabelled negative.

Table III shows the confusion matrix in terms of the TP, FN, FP, and TN values.

Relying on the confusion matrix, the accuracy, sensitivity, and error rate metrics are derived. For a given classifier, the accuracy can be calculated by considering the recognition rate, which is the percentage of records in the test set that are correctly classified (fraudulent or non-fraudulent). The accuracy is defined as:

$$Accuracy = \frac{(TP+TN)}{number \; of \; all \; records \; in \; the \; testing \; set} \qquad (5)$$

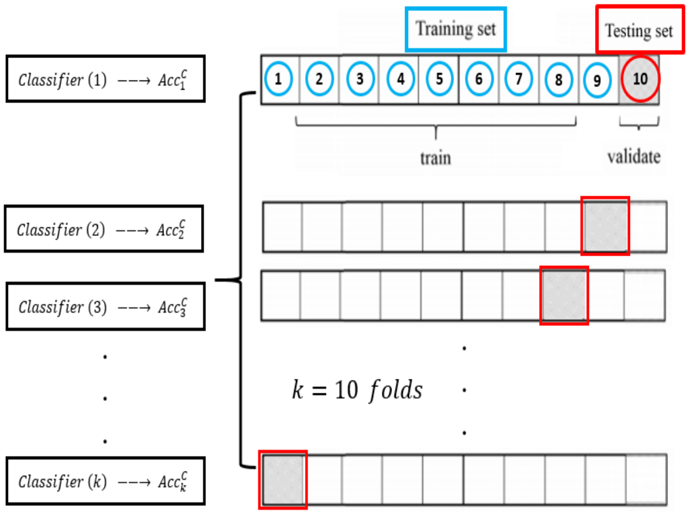

Training set
Testing set
$Classifier\ (1) \dashrightarrow Acc_1^C$
1
2
3
4
5
6
7
8
9
10
train
validate
$Classifier\ (2) \dashrightarrow Acc_2^C$
$Classifier\ (3) \dashrightarrow Acc_3^C$
$k = 10\ folds$
$Classifier\ (k) \dashrightarrow Acc_k^C$

Fig. 19. Classifiers with Corresponding Accuracies.

Mechanisms for accuracy-based evaluation. In this context, a higher accuracy corresponds to a better classifier output. The maximum value of the accuracy metric is 1 (or 100%), which is achieved when the classifier classifies the records correctly without any errors in the classification process.

Sensitivity refers to the true positive recognition rate. It is given by:

$$Sensitivity = \frac{TP}{P} \quad (6)$$

Mechanisms for sensitivity-based evaluation. In this context, a higher sensitivity corresponds to a better classifier output. The maximum value of the sensitivity metric is 1 (or 100%), which is achieved when the proportion of true positive cases equals the number of actual positive cases.

The error rate is defined as the ratio of mistakes made by the classifier during the prediction process. It is defined as:

$$eror\ rate = 1 - accuracy \quad (7)$$

Mechanisms for error rate-based evaluation. In this context, a higher accuracy corresponds to a worse classifier output. The maximum value of the accuracy metric is 1 (or 100%), which is achieved when the classifier classifies all the records incorrectly (i.e., the accuracy is zero).

G. Examining the Value of the Accuracy

In this step, the final calculated accuracy is examined. If it is accepted, then the classifier can be used in real-life situations. Otherwise, the process of building the classifier has a problem, and then retraining the classifier is required.

TABLE III. CONFUSION MATRIX

Actual class (Predicted class)	**Confusion matrix**		
	C1	**¬ C1**	**Total**
C1	True positives (TP)	False negatives (FN)	TP + FN = P
¬ C1	False positives (FP)	True negatives (TN)	FP + TN = N

Security and privacy issues are highly stressed according to many studies [35-43] when using data in the artificial intelligence research field. This is because the data reflect the policies and sensitive issues of the institution in question (these are banks in our work when applying the proposed classifier in reality). Therefore, the privacy and security of data are not considered in this work, but they will be considered in future work.

IV. USED METRICS

Since the domain of this work is artificial intelligence, two types of metrics are used. They are AI-based metrics and performance-based metrics.

A. AI-based Metrics

In this context, the confusion matrix dominates the situation. In other words, the metrics that are derived from the confusion matrix are employed to measure the prediction accuracy of the classifier.

B. Performance-based Metrics

In this context, time dominates the situation. In other words, the total time ($ToTi$) required to build, train, and test the classifier is used as a benchmark. The $ToTi$ is given by:

$$ToTi = T_{pre} + T_{dbs} + T_{tr} + T_{ts} \quad (8)$$

where T_{pre} refers to the preprocessing time, T_{dbs} refers to the database splitting time, T_{tr} refers to the training time, and T_{ts} refers to the testing time. It is well known that the lower the total time is, the higher the degree of performance.

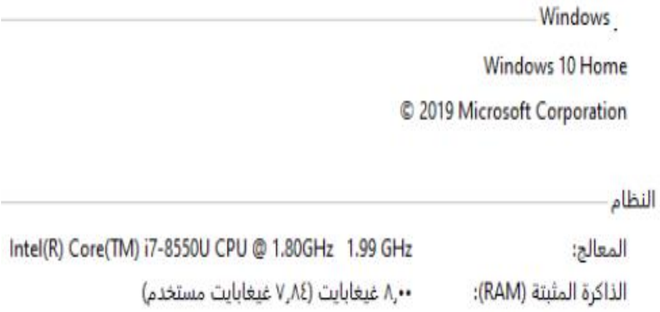
V. RESULTS AND DISCUSSIONS

This section is structured so that the specifications of the machine used to implement the proposed classifier are introduced. Then, the classifiers that are compared with the proposed classifier are described. Finally, the results are provided along with two discussions.

A. Setup

The system is performed on a machine that has the specifications summarized in Fig. 20.

The programming language used for the implementation of the classifier is Python.



Windows

Windows 10 Home

© 2019 Microsoft Corporation

النظام

المعالج: Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz 1.99 GHz

الذاكرة المثبتة (RAM): ٨,٠٠ غيغابايت (٧,٨٤ غيغابايت مستخدم)

Fig. 20. Specifications of the Machine used to Implement the Classifiers.

B. Selected Classifiers

Two classifiers are selected for a comparison with the classifier proposed in this work. They are the K-nearest neighbours (KNN) classifier and the voting classifier (VC). Below, a brief description of each selected classifier is presented.

Fig. 21 shows the fundamental steps required to build the voting classifier.

As shown in Fig. 21, there are many classifiers, and a voting step is required to produce the final output class. The voting step means that the final output of the classifier depends on the majority of the classes (predictions) that are generated by the classifiers. For example, there are three classifiers in Fig. 22. The final prediction is either Fraud (F) or Non-Fraud (NF). The voting process works as follows:

1) Obtain the outputs of the classifiers.

2) Calculate the number of classifiers that generate the F class (let us say 2 classifiers).

3) Calculate the number of classifiers that generate the NF class (let us say 1 classifier).

4) The majority is 2. Therefore, the final prediction is the F class.

Fig. 22 shows the fundamentals steps for building the KNN classifier.

As shown in Fig. 22, there are two clusters (one for fraudulent transactions and one for non-fraudulent transactions). Each cluster has a centre, which is represented numerally by (-1) for nonfraudulent transactions and (+1) for fraudulent transactions. For a given transaction, the KNN classifier processes the transaction and generates a corresponding number. Then, the distance between the generated value and the centre of each cluster is calculated. Finally, the transaction is assigned to the correct cluster (in the example, it is assigned to the non-fraud cluster).

C. Results

Since the cross-validation method is used to divide the database, we obtain ten sub-classifiers as mentioned previously. The process of calculating the final values of the AI-based metrics depends on the "average" mathematical operation. Table IV summarizes the obtained results.

Table V summarizes the comparison of the logistic regression (LogR)-based classifier with both the KNN-based classifier and the VC-based classifier.

Discussion. From Table V, it is obvious that the LogR classifier achieves the best values in terms of accuracy, sensitivity, and error rate. The reason behind this is related to the efficient preprocessing technique used to remove outliers and manipulate the missing values. In addition, cross validation ensures that the entire database is employed as both the training and testing data sets, and this in turn enhances the three metrics. The KNN classifier comes in second, and the VC classifier comes in third. This is because the KNN classifier includes a step related to calculating the distances between the value of the new transaction and the centres of clusters. This in turn reflects efficient processing in the prediction process compared to poor processing in the VC classifier (i.e., only calculating the majority).

For performance comparison purposes, the bare chart shown in Fig. 23 illustrates the values of the response time for all classifiers involved in the comparison.

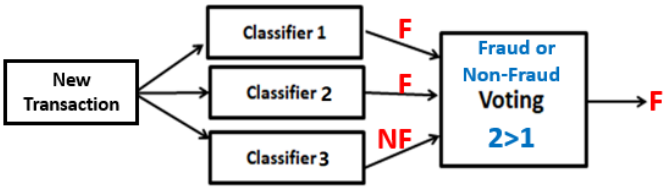


Fig. 21. Basic Concept of the Voting Classifier.

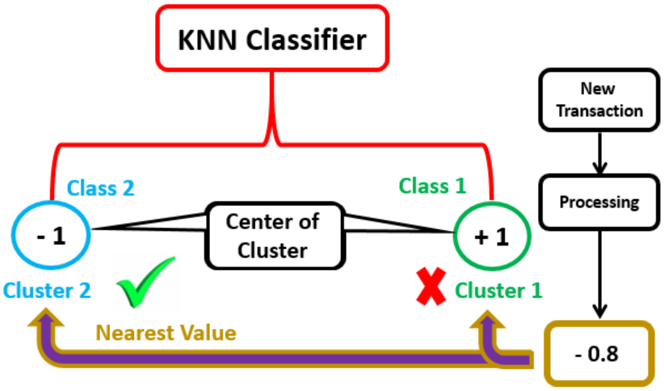


Fig. 22. Basic Concept of the KNN Classifier.

TABLE IV. EVALUATING THE PROPOSED CLASSIFIER

K-value	Accuracy	Sensitivity	Error rate
1	96%	97%	4%
2	98%	96%	2%
3	98%	97%	2%
4	96%	96%	4%
5	97%	98%	3%
6	96%	98%	4%
7	97%	96%	3%
8	98%	98%	2%
9	98%	98%	2%
10	98%	96%	2%
Average	97.2%	97%	2.8%

TABLE V. COMPARISON OF CLASSIFIERS

Classifier	Metrics		
	Accuracy	*Sensitivity*	*Error rate*
LogR classifier	97.2%	97%	2.8%
KNN classifier	93%	94%	7%
VC classifier	90%	88%	10%

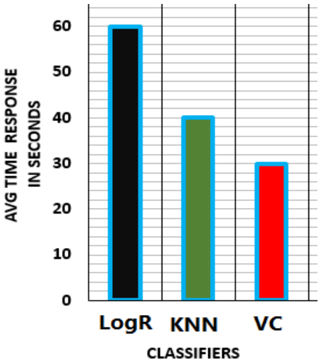


Fig. 23. Performances of the Three Classifiers.

Discussion. Fig. 23 shows that the VC classifier achieves the best performance. This is because it depends only on a simple mathematical operation (the sum operation) to determine the classes and generate the final output. The KNN classifier comes in the second in terms of its response time. That is because this classifier must perform additional mathematical operations related to calculating the distances between the new value and the centre of each cluster, and these operations in turn consumes more time. Compared to the previous classifiers, the LogR classifier performs the worst. The reason for this is that the time required for database division and training the sub-classifiers is very high. In other words, training and testing ten sub-classifiers logically takes time less than training and testing one classifier (i.e., the KNN and VC classifiers). However, although the response time of the LogR classifier is the longest, it achieves the best accuracy. From the point of view of detecting fraud (or security), accuracy more of a concern than performance. This issue will be taken into consideration in future work.

VI. Conclusion

The detection of credit card fraud is a vital research field. This is because of the increasing number of fraud cases in financial institutions. This issue opens the door for employing artificial intelligence to build systems that can detect fraud. Building an AI-based system to detect fraud requires a database to train the system (or classifier). The data in reality are dirty and have missing values, noisy data, and outliers. Such issues negatively affect the accuracy rate of the system. To overcome these problems, a logistic regression-based classifier is proposed. The data are first cleaned using two methods: the mean-based method and clustering-based method. Second, the classifier is trained based on the cross-validation technique (folds=10), which ensures that the whole database is used as both the training data set and testing data set. Finally, the proposed classifier is evaluated based on the accuracy, sensitivity, and error rate metrics. The proposed logistic regression-based classifier is compared to well-known classifiers, which are the K-nearest neighbours classifier and the voting classifier. The logistic regression-based classifier generates the best results (accuracy = 97.2%, sensitivity = 97%, and error rate = 2.8%).

Limitations. The performance of the proposed classifier suffers in terms of response time. In addition, it does not apply to data in real time.

Future work. In future work, we intend to enhance the performance and take the security and privacy of the data in real time into consideration.

References

[1] Yousefi, Niloofar, Marie Alaghband, and Ivan Garibay. "A Comprehensive Survey on Machine Learning Techniques and User Authentication Approaches for Credit Card Fraud Detection." arXiv preprint arXiv:1912.02629 (2019).

[2] Paschen, Jeannette, Jan Kietzmann, and Tim Christian Kietzmann. "Artificial intelligence (AI) and its implications for market knowledge in B2B marketing." Journal of Business & Industrial Marketing (2019).

[3] Abdallah, Aisha, Mohd Aizaini Maarof, and Anazida Zainal. "Fraud detection system: A survey." Journal of Network and Computer Applications 68 (2016): 90-113.

[4] Alladi, Tejasvi, et al. "Consumer IoT: Security vulnerability case studies and solutions." IEEE Consumer Electronics Magazine 9.2 (2020): 17-25.

[5] Rahman, Rizwan Ur, et al. "Classification of Spamming Attacks to Blogging Websites and Their Security Techniques." Encyclopedia of Criminal Activities and the Deep Web. IGI Global, 2020. 864-880.

[6] Somasundaram, Akila, and Srinivasulu Reddy. "Parallel and incremental credit card fraud detection model to handle concept drift and data imbalance." Neural Computing and Applications 31.1 (2019): 3-14.

[7] Gianini, Gabriele, et al. "Managing a pool of rules for credit card fraud detection by a Game Theory based approach." Future Generation Computer Systems 102 (2020): 549-561.

[8] Dal Pozzolo, Andrea, et al. "Credit card fraud detection: a realistic modeling and a novel learning strategy." IEEE transactions on neural networks and learning systems 29.8 (2017): 3784-3797.

[9] Wang, Chunhua, and Dong Han. "Credit card fraud forecasting model based on clustering analysis and integrated support vector machine." Cluster Computing 22.6 (2019): 13861-13866.

[10] Deufel, Patrick, Jan Kemper, and Malte Brettel. "Pay now or pay later: A cross-cultural perspective on online payments." Journal of Electronic Commerce Research 20.3 (2019): 141-154.

[11] Co-Pending, U. S. "patent application No." US201514696366, filed on Apr 24 (2015).

[12] Hamid, N. R., and Aw Yoke Cheng. "A risk perception analysis on the use of electronic payment systems by young adult." WSEAS Transactions on Information Science and applications 10.1 (2013): 26-35.

[13] inc website (2020), online available : https://www.inc.com/guides/cust_tech/20909.html, access (10 March 2020).

[14] Janbandhu, Ruchika, Shameedha Begum, and N. Ramasubramanian. "Credit Card Fraud Detection." Computing in Engineering and Technology. Springer, Singapore, 2020. 225-238.

[15] Mittal, Sangeeta, and Shivani Tyagi. "Computational Techniques for Real-Time Credit Card Fraud Detection." Handbook of Computer Networks and Cyber Security. Springer, Cham, 2020. 653-681.

[16] Zou, Junyi, Jinliang Zhang, and Ping Jiang. "Credit Card Fraud Detection Using Autoencoder Neural Network." arXiv preprint arXiv:1908.11553 (2019).

[17] Jiang, Changjun, et al. "Credit card fraud detection: A novel approach using aggregation strategy and feedback mechanism." IEEE Internet of Things Journal 5.5 (2018): 3637-3647.

[18] Murli, Divya, et al. "Credit card fraud detection using neural networks." International Journal of Students' Research in Technology & Management 2.2 (2015): 84-88.

[19] Darwish, Saad M. "An intelligent credit card fraud detection approach based on semantic fusion of two classifiers." Soft Computing 24.2 (2020): 1243-1253.

[20] Marchal, Samuel, and Sebastian Szyller. "Detecting organized eCommerce fraud using scalable categorical clustering." Proceedings of the 35th Annual Computer Security Applications Conference. 2019.

[21] Kim, Jeongrae, Han-Joon Kim, and Hyoungrae Kim. "Fraud detection for job placement using hierarchical clusters-based deep neural networks." Applied Intelligence 49.8 (2019): 2842-2861.

[22] Fu, Kang, et al. "Credit card fraud detection using convolutional neural networks." International Conference on Neural Information Processing. Springer, Cham, 2016.

[23] Kamaruddin, Sk, and Vadlamani Ravi. "Credit card fraud detection using big data analytics: use of PSOAANN based one-class classification." Proceedings of the International Conference on Informatics and Analytics. 2016.

[24] Arun, C., and C. Lakshmi. "Class Imbalance in Software Fault Prediction Data Set." Artificial Intelligence and Evolutionary Computations in Engineering Systems. Springer, Singapore, 2020. 745-757.

[25] Thabtah, Fadi, et al. "Data imbalance in classification: Experimental evaluation." Information Sciences 513 (2020): 429-441.

[26] Maung, Ei Thinzar Win. Comparison of Data Mining Classification Algorithms: C5. 0 and CART for Car Evaluation and Credit Card Information Datasets. Diss. Unversity of Computer Studies, Yangon, 2020.

[27] Rike, James B. "Cylinder support system." U.S. Patent Application No. 29/641,843.

[28] Freund, Peter C. "Method and system for performing purchase and other transactions using tokens with multiple chips." U.S. Patent No. 10,282,536. 7 May 2019.

[29] Benamar, Lamya, Christine Balagué, and Zeling Zhong. "Internet of Things devices appropriation process: the Dynamic Interactions Value Appropriation (DIVA) framework." Technovation 89 (2020): 102082.

[30] Kaggle , website (2020). Avaliable : https://www.kaggle.com/mlg-ulb/creditcardfraud (access 22 July 2020).

[31] DataCamp , website (2020). Avaliable : https://www.datacamp.com/community/tutorials/understanding-logistic-regression-python (access 28 July 2020).

[32] Salillari, Denisa, and Luela Prifti. "Comparison Study of Logistic Regression Model for Albanian Texts." *Journal of Advances in Mathematics* 12.9 (2016): 6572-6575.

[33] javatpoint , website (2020). Avaliable : https://www.javatpoint.com/logistic-regression-in-machine-learning (access 22 July 2020).

[34] Mona Alfifi, Mohamad Shady Alrahhal, Samir Bataineh and Mohammad Mezher, “Enhanced Artificial Intelligence System for Diagnosing and Predicting Breast Cancer using Deep Learning” International Journal of Advanced Computer Science and Applications(IJACSA), 11(7), 2020. http://dx.doi.org/10.14569/IJACSA.2020.0110763

[35] Alrahhal, Mohamad Shady, Maher Khemakhem, and Kamal Jambi. "Agent-Based System for Efficient kNN Query Processing with Comprehensive Privacy Protection." International Journal Of Advanced Computer Science And ApplicationS 9.1 (2018): 52-66.

[36] Alrahhal, Mohamad Shady, et al. "AES-route server model for location based services in road networks." International Journal Of Advanced Computer Science And Applications 8.8 (2017): 361-368.

[37] Alrahhal, Mohamad Shady, Maher Khemakhem, and Kamal Jambi. "A SURVEY ON PRIVACY OF LOCATION-BASED SERVICES: CLASSIFICATION, INFERENCE ATTACKS, AND CHALLENGES." Journal of Theoretical & Applied Information Technology 95.24 (2017).

[38] Alrahhal, Mohamad Shady, Maher Khemekhem, and Kamal Jambi. "Achieving load balancing between privacy protection level and power consumption in location based services." (2018).

[39] Alrahhal, H.; Alrahhal, M.S.; Jamous, R.; Jambi, K. A Symbiotic Relationship Based Leader Approach for Privacy Protection in Location Based Services. ISPRS Int. J. Geo-Inf. 2020, 9, 408.

[40] Al-Rahal, M. Shady, Adnan Abi Sen, and Abdullah Ahmad Basuhil. "High level security based steganoraphy in image and audio files." Journal of theoretical and applied information technology 87.1 (2016): 29.

[41] Alluhaybi, Bandar, et al. "A Survey: Agent-based Software Technology Under the Eyes of Cyber Security, Security Controls, Attacks and Challenges." International Journal of Advanced Computer Science and Applications (IJACSA) 10.8 (2019).

[42] Fouz, Fadi, et al. "Optimizing Communication And Cooling Costs In Hpc Data Center." Journal of Theoretical and Applied Information Technology 85.2 (2016): 112.

[43] Alrahhal, Mohamad Shady, and Adnan Abi Sen. "Data mining, big data, and artificial intelligence: An overview, challenges, and research questions." (2018).