IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 7, 2020, accepted January 20, 2020, date of publication February 6, 2020, date of current version February 14, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2972009

A Multiple Classifiers System for Anomaly Detection in Credit Card Data With Unbalanced and Overlapped Classes

SURAYA NURAIN KALID[1], KENG-HOONG NG[1], GEE-KOK TONG[1], AND KOK-CHIN KHOR[2]
[1]Faculty of Computing and Informatics, Multimedia University, Cyberjaya 63100, Malaysia
[2]Lee Kong Chian Faculty of Engineering Science, Universiti Tunku Abdul Rahman, Kajang 43000, Malaysia

Corresponding author: Kok-Chin Khor (kckhor@utar.edu.my)

This work was supported by the Ministry of Higher Education Malaysia under Grant FRGS/1/2019/SS01/MMU/03/11.

ABSTRACT Frauds and default payments are two major anomalies in credit card transactions. Researchers have been vigorously finding solutions to tackle them and one of the solutions is to use data mining approaches. However, the collected credit card data can be quite a challenge for researchers. This is because of the data characteristics that contain: (i) unbalanced class distribution, and (ii) overlapping of class samples. Both characteristics generally cause low detection rates for the anomalies that are minorities in the data. On top of that, the weakness of general learning algorithms contributes to the difficulties of classifying the anomalies as the algorithms generally bias towards the majority class samples. In this study, we used a Multiple Classifiers System (MCS) on these two data sets: (i) credit card frauds (CCF), and (ii) credit card default payments (CCDP). The MCS employs a sequential decision combination strategy to produce accurate anomaly detection. Our empirical studies show that the MCS outperforms the existing research, particularly in detecting the anomalies that are minorities in these two credit card data sets.

INDEX TERMS Anomaly detection, credit card, multiple classifiers.

I. INTRODUCTION

Credit cards are widely used because they ease our daily transactions in many ways. However, banks need to take note of these issues seriously, i.e., (i) the intervention of unauthorised third parties – frauds, and (ii) the negligence of repayment by cardholders – default payments.

According to [1], the global credit card fraud losses have shown an uptrend, from USD 9.84 billion in the year 2011 to USD 27.69 billion in the year 2017. It is also reported that the worldwide credit card fraud is expected to reach a total of USD 31.67 million by the year 2020. The Malaysian banking sector also reported a total loss of RM 51.3 million in the credit card fraud in the year 2016 [2]. It was reported that in the year 2016, the outstanding balance of credit card holders in Malaysia is RM36.9 million and 12.8% of them failed to pay the minimum payment of the balance [3]. The Central Bank of Malaysia (Bank Negara Malaysia) also reported that the high outstanding balance by the credit card holders, has triggered an alarm to the Malaysian government [4].

Researchers have been vigorously finding ways to tackle both issues, including data mining. Data mining is not an option or a trend, but more of a necessity that the banking sector should invest in [5], [6]. However, banking data such as credit card fraud and default payment are quite of a challenge to data mining researchers. This is because the data usually exhibited characteristics: (i) unbalanced class distribution, and (ii) overlapping of class samples.

The size of the important classes in the data, i.e., fraud and default payment, are usually the minorities. Generally, it is easy for learning algorithms to find their regularities if they have sufficient records. But when their numbers are very small, finding their regularities becomes difficult and so as generalising their actual decision regions using learning algorithms [7]–[10]. It adds difficulty if their attribute values are overlapped by a large amount of normal transactions. In general, the performance of learning algorithms will be less affected if the minority classes are linearly separable, even though the data involved are highly unbalanced [11]–[13].

The associate editor coordinating the review of this manuscript and approving it for publication was Zhe Xiao.

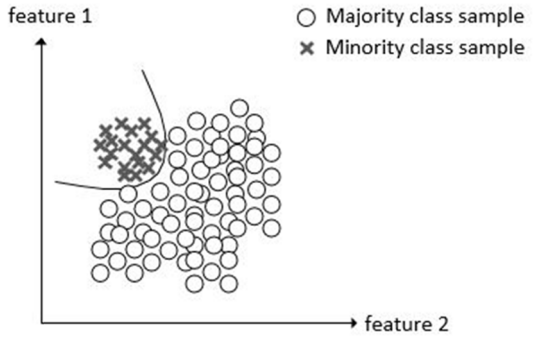

FIGURE 1. The decision boundary that separates the majority class and minority class samples.

Another aspect to take note is the weakness of learning algorithms in assessing its own classification capability [7], [11]–[14]. A commonly used metric to evaluate data mining results, which computes the number of correctly classified records, is the classification accuracy. The learning algorithms generally assume that positive and negative samples are roughly equal in data. Therefore, many learning algorithms aim to maximize accuracy, which lean more towards majority classes and against minorities. Subsequently, they are unlikely to produce satisfactory results when dealing with unbalanced data sets, especially the minority classes.

In a nutshell, the characteristics of the credit card data and learning algorithms have caused the problem of low anomaly detection rates. Therefore, single classifiers may not give good classification results. Hence, this research aims to design a Multiple Classifier System (MCS) for mitigating the low anomaly detection rate problem on the credit card data sets utilised in this study.

II. LITERATURE REVIEW

A. UNBALANCED CLASS DISTRIBUTION

A data set is unbalanced in its class distribution when one or more classes have a much greater number of samples than the other classes [17], [18]. In reality, the number of anomalies are very much fewer than normal transactions in the credit card data.

The unbalanced class distribution problem has been the focus of many researchers [19]–[22]. This is due to the high probability of producing errors in classification. The unbalanced class distribution is as illustrated in Fig. 1. The curved line that separates between two classes is called the decision boundary, which separates the region of different classes [23].

Classifying an unbalanced data set may lead to the low classification rate problem [24]–[26]. According to [27], general learning algorithms are able to give high accuracies on balanced data sets, but not on unbalanced data sets. Many researchers did comparative studies using some popular learning algorithms on unbalanced data sets [21], [28], [29]. In the next section, some selected popular single learning algorithms shall be discussed. A brief explanation on why they are weak towards unbalanced data sets shall be given.

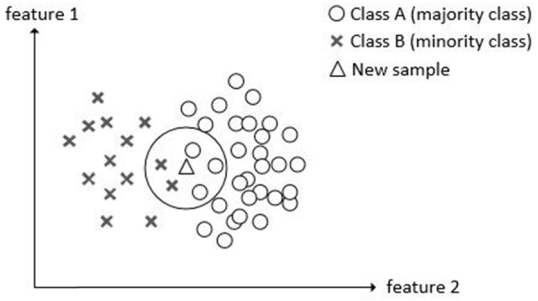

FIGURE 2. The illustration of finding the class label for a new sample.

B. LEARNING ALGORITHM

1) NAÏVE BAYES

NB is a simple, yet powerful learning algorithm that uses the probabilistic method to classify data samples. It assumes that every attribute is conditionally independent of the other attributes [30]. It will predict whether a data sample belongs to one class or another based on the Bayesian Theorem as per (1).

$$P(C \mid X) = \frac{P(X \mid C)\, P(C)}{P(X)} \tag{1}$$

Let X be a data sample (evidence) that is described by multiple attributes. The probability of X belongs to a class C is calculated as $P(C|X)$. $P(C)$ is the initial or prior probability while $P(X|C)$ is the likelihood or the probability that the sample data is observed. $P(X)$ is the evidence with a constant value and therefore can be omitted.

According to [31], NB is weak against unbalanced data sets as it biases towards the majority class. As illustrated in Fig. 2, given two classes of different class distribution (class A 20: class B 10) and a new sample that needs to be classified.

Firstly, NB will find the prior probability P(C) of each class; it is assumed that the new sample will belong to Class A as it has more samples as compared with Class B. Secondly, the likelihood is calculated based on the number of samples of each class that is within the vicinity (within the circle) of the new sample. Lastly, the posterior probability is calculated by combining the result of the prior probability and the likelihood of the new sample (refer to (2) and (3)).

$$Posterior\ prob.: Class\ A = \frac{20}{30} \times \frac{3}{20} = 0.100 \tag{2}$$

$$Posterior\ prob.: Class\ B = \frac{10}{30} \times \frac{2}{10} = 0.067 \tag{3}$$

The new sample is classified as Class A because of the largest posterior probability. Using this example, it shows that NB is likely bias towards the majority class.

2) C4.5

C4.5 is a popular learning algorithm that uses a divide-and-conquer method to build a decision tree from a training set [32]–[34]. It is popular due to its ability to produce good classification results in a much shorter time. To improve classification performance, it prunes small and deep nodes

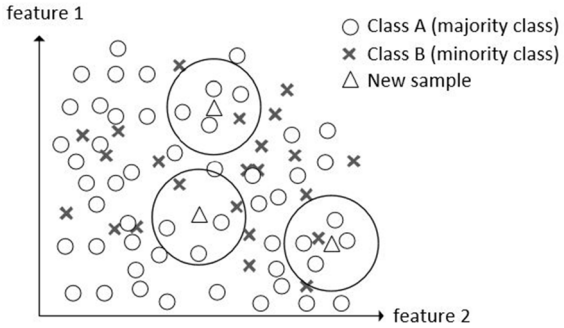

FIGURE 3. The illustration of finding the class label for a new sample using KNN.

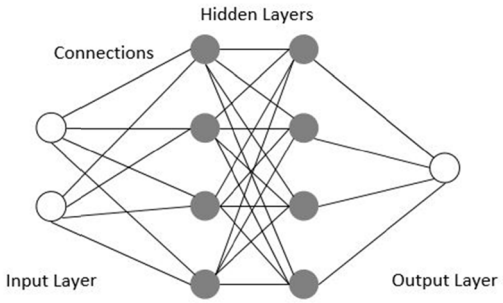

FIGURE 4. The illustration of an ANN model that contains an input layer, hidden layers, and an output layer.

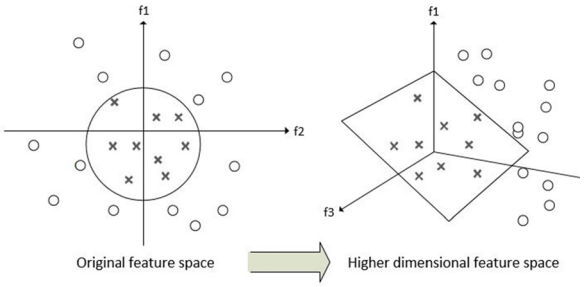

FIGURE 5. SVM transforms an original feature space into a higher dimensional feature space for finding a better decision boundary.

in the preliminary tree caused by the 'noises' contained in training samples. The advantage of pruning is that it will decrease the risk of 'over-fitting' [33], [35], [36]. Over-fitting refers to a classifier that learns a training data, the details as well as noises, too well. Due to the inability of such classifier in generalising the training data well, the classifier is weak in classifying new or unknown data.

Avoiding over-fitting gives a more precise classification for unknown data [37], [38]. Nevertheless, the pruning process can also be a disadvantage to unbalanced data sets. Removing 'noises' from such data set may also remove small and deep nodes of the preliminary tree that belong to a minority class, thus reducing the coverage for a precise classification [14], [39]–[41].

3) K-NEAREST NEIGHBOUR (KNN)

KNN builds the classifier's function by majority vote of its local neighbouring data points [42], [43]. Fig. 3 shows how KNN identifies the class of a new sample. Suppose the number of neighbours, $k = 5$, and the Euclidean distance is the distance measure. The KNN classifier will find the nearest five samples to the new sample. The Euclidean distance between the target sample (x) and the new sample (y) in an n-dimensional space is calculated using the measure as per (4), where p is 2.

$$L_p = (\sum_{i=1}^{n} |x_i - y_i|^p)^{1/p} \tag{4}$$

Out of five neighbours, the samples of class A is more than the samples of class B. Therefore, the new sample is classified as class A. This example shows that KNN is likely bias towards the majority class. The chance of the new sample to be classified as class B is relatively low as compared with Class A.

4) ARTIFICIAL NEURAL NETWORK (ANN)

ANNs are made up of simple and highly interconnected nodes that respond to inputs by the dynamic state of the nodes [42], [44]. It is made up of layers, i.e., input layer, hidden layer, and output layer, as shown in Fig. 4. These layers are formed using interconnected 'nodes' that are associated with activation functions.

ANNs contain some forms of learning rules that modify the weights of the connections according to the input patterns – learning by examples [42], [45]. Similar to other single classifiers, ANNs are also biased towards majority classes when they involve unbalanced data sets [26], [46]. Due to overwhelming samples of majority classes, samples of minority classes will be imperceptible to ANNs.

5) SUPPORT VECTOR MACHINE (SVM)

SVM solves classification and regression problems. As shown in Fig. 5, SVM plots each sample as a point in n-dimensional space, where n is the number of attributes in a data set. The value of each attribute will be the value of a particular coordinate. Then, SVM will identify the best hyper-plane that is able to differentiate classes [44].

SVM is powerful in getting the best decision boundary between classes. However, SVM does not work well with data sets that contain unbalanced class distribution, noises and overlapping class samples. The parameters in SVM can be altered to make the classifier more immune to noises and to work well for balanced data sets. But when it involves unbalanced data sets, minority class samples may consider as noises. Therefore, minority class samples will be ignored completely by SVM [47].

C. OVERLAPPING OF CLASSES

The other factor that contributes to the low classification rate is the overlapping class samples. Overlapping happens when

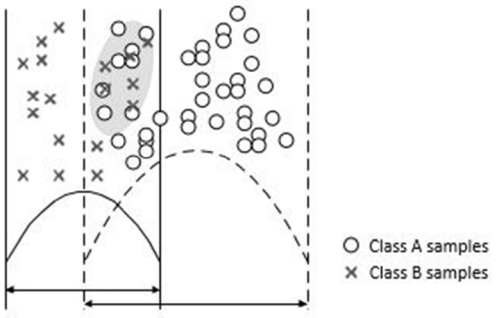

FIGURE 6. The unbalanced classes and overlapping samples that may lead to a low classification rate.

the samples are located too close to the decision boundary of classes and overlapped with each other [20], [21].

Based on a systemic study using a set of artificially generated data sets prepared by [48], the study result showed that the degree of overlapping class samples had a strong correlation with unbalanced class distribution. Fig. 6 shows the relationship between unbalanced class distribution and overlapping class samples. When the samples of two classes are not well distributed, some samples may overlap with each other. The grey area in Fig. 6 indicates the samples that are overlapped.

Another reason of overlapping is because the samples in both classes share almost that same value of attributes. Such overlapping causes difficulties for a classifier to classify the samples and may eventually lead to a low classification rate [49], [50]. In the next section, solutions to the unbalanced class distribution and the overlapping class samples shall be discussed based on the previous work conducted by other researchers.

D. MULTIPLE CLASSIFIERS SYSTEM (MCS)

Many researchers attempted different solutions to address the low classification rate problem caused by the two problems, as discussed in the earlier section [51]–[54], [83], [84]. One of the solutions is MCS. MCS is the combination of a set of classifiers to produce a better prediction. MCS employs various decision combination strategies that are able to produce a more robust, reliable, efficient recognition and accurate classification [55]–[59]. There are three combination strategies when employing MCS: (i) sequential combination, (ii) parallel combination, and (iii) hybrid combination.

1) SEQUENTIAL COMBINATION

Using the sequential combination, two or more single classifiers process the input data in a sequential manner. As shown in Fig. 7, the output resulted from a single classifier will then be used as the input data for the subsequent single classifiers.

Usually, simple classifiers are utilised first, followed by more accurate and complex classifiers [55]. However, this order can be reversed depending on the needs of the design. When the prior classifier is unable to accurately classify one of the class samples, then the sample is given to the next classifier for further classification [58]. An example

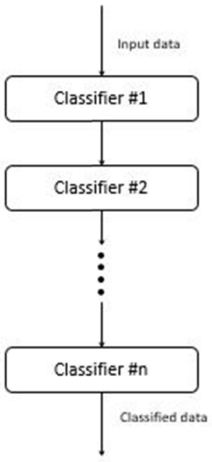

FIGURE 7. The sequential combination of MCS. The MCS passes data from one classifier to another.

of learning algorithm that uses sequential combination is Boosting [60].

Reference [61] used AdaBoost on a credit card fraud data set. The data set is highly unbalanced with only 0.173% fraud transaction. The authors used Naïve Bayes as the base classifier. Upon completing the experiment, they obtained 0.999, 0.825 and 0.981 for accuracy, true positive rate and true negative rate, respectively. However, accuracy is not a good performance measure as compared to the other two measures when involving unbalanced data sets. This is because the accuracy measure favours the majority class.

Reference [62] used a fine-tuned boosting ensemble approach, which is known as XGBoost, to solve a classification problem. The problem was to decide on the granting of loan application. The data sets used in the experiments were German credit, Australian credit, Taiwan credit, P2P landing data set A, and P2P landing data set B. Three of them are unbalanced and the other two are approximately balanced in class distribution. The classifier's performance was measured using accuracy and the highest accuracy obtained was 0.879. Using accuracy in their work showed that the authors did not explicitly address the issues of unbalanced data and overlapping class samples. However, the authors suggested to integrate XGBoost into MCS to further improve the classification performance. MCS is one of the recommended approaches to consider in data mining research when involving unbalanced data sets.

Reference [79] also used boosting algorithm, AdaBoost. M1. The authors boosted three different classifiers, which were Multilayer Perceptron (MLP), Radial Basis Function (RBF), and Naïve Bayes (NB), on the Taiwan credit card default payment data. To handle the unbalanced data set issue, the authors reduced the size of majority class samples by using random under sampling.

2) PARALLEL COMBINATION

The same data are processed by multiple single classifiers using the parallel combination, where each classifier is

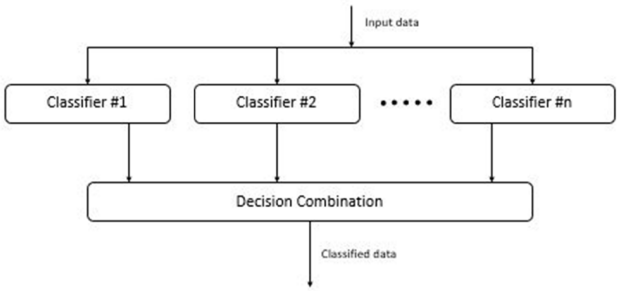

FIGURE 8. The parallel combination of the MCS. All the single classifiers involved are independent from each other.

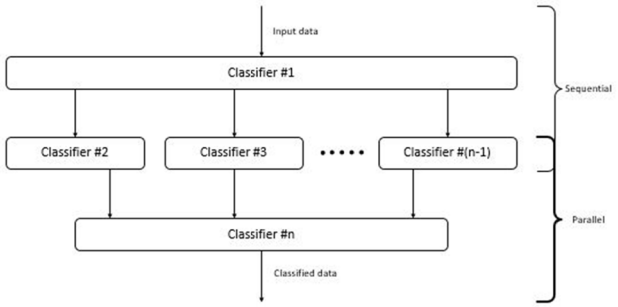

FIGURE 9. The MCS that utilises the hybrid combination. The sequential and parallel combinations are put into one architecture.

independent from the others. The output from all the single classifiers will be combined to get a final decision, as shown in Fig. 8. An example of learning algorithm that uses parallel combination is Bagging [60].

Reference [63] used Bagging with Random Forest as the base classifier. The data set used was about credit card default payment from a Taiwan bank. The data set is unbalanced with 28% of it categorised as default-payment. The author obtained 0.816 for accuracy, 0.371 for TPR, and 0.764 for AUC.

Reference [64] used the bagged decision trees to classify new transactions as fraudulent or legitimate type. The data set used in the experiment was the UCSD-FICO competition credit card data set. Out of 97,707 instances, the data set contains only 2.9% of fraudulent instances. The authors did some comparison with other single classifiers and the result showed that bagging approach outperformed other single classifiers.

The authors of [78] used Random Forest, an ensemble method. Random Forest uses the same combination strategy as Bagging. Similar to [63], credit card default payment data set from a Taiwan bank was used. In the experiment, the authors applied the Correlation Based Feature Selection (CFS) technique to reduce the data set dimension with the purpose of improving classification accuracy. Based on the experiment result, the authors obtained a good TPR f 0.816.

In the study by [81], the credit card fraud data set used was obtained from a European bank with an unbalanced class distribution and overlapping class samples. The data set is highly unbalanced with only 492 fraud transactions (0.173%). The authors first partitioned and clustered the data set. Then, a Random Forest framework, with C4.5 as the base classifier, was trained using the resulted clusters. The final result was determined based on the majority vote. The authors used AUC as the performance measure. Based on the experiment result, the author obtained an AUC value of 0.965.

The study by [82] used the similar data set used as [81]. The data set was bootstrapped such that each resulted data set had a balanced class distribution. Then, the authors used an ensemble of Deep Belief Network and applied to each of the bootstrapped samples. The author used the performance measures, i.e., accuracy, TPR, TNR, and AUC and obtained 0.906, 0.818, 0.995, and 0.978, respectively.

3) HYBRID COMBINATION

The hybrid combination puts the sequential and parallel combinations into one architecture. Fig. 9 illustrates the hybrid combination.

In this case, the input data is fed to the first classifier. The output from the first classifier will be the input to several parallel classifiers. Then, a single combination function or classifier will merge the output of the individual parallel classifiers.

Reference [65] used the hybrid combination to classify credit or loan applicants into good and bad applicants. The data set used in the experiment is a German credit data set. The data set has 20 attributes with 700 Good and 300 Bad Applicant data. The hybrid combinations used are a two-level voting scheme: level I – AdaBoost approach, and level II – single classifier approach. The authors used accuracy as the performance measure. Both level I and level II had achieved an average accuracy of 76.33% and 78.33%, respectively. However, the evaluation measure is not suitable as the study involved the unbalanced data set.

Reference [80] also employed a hybrid model using a combination of AdaBoost and majority voting. ANN and NB were used as the base classifiers for AdaBoost that employed sequential combination. Then, the final result was obtained using majority voting, which was done in parallel. Similar to [81], the credit card fraud data set obtained from the European bank was used in this experiment. Upon completing the experiment, the authors were able to obtain an accuracy of 0.999, a TPR of 0.789 and a TNR 0.999.

E. EVALUATION MEASURES

Given a binary class problem, general learning algorithms assume that the classes involved are approximately balanced and attempt to maximise its accuracy regardless of classes [66], [67]. With a balanced data set, accuracy is the suitable measure to evaluate the performance of a classifier. However, when it involves an unbalanced data set, learning algorithms may bias towards the majority class [68]–[70]. This may lead to a high accuracy in overall, but a poor classification rate for the minority class.

In this study, the True Positive Rate (TPR) was used to evaluate the performance of the proposed MCS. By using

TABLE 1. The attributes of the Credit card fraud (CCF) data set.

No.	Attribute(s)	Description
1	TIME	The seconds elapsed between each transaction and the first transaction in the dataset
2, 3 - 29	V1, V2…V28	Not disclosed
30	AMOUNT	The transaction amount
31	CLASS	Fraud transaction is 1 otherwise 0

TPR as the evaluation metric, we were able to identify the classification rate for both majority and minority classes. In this study, we focused on the minority classes: (i) the credit card frauds and (ii) the credit card default payments.

III. METHODOLOGY

A. DATA SETS OVERVIEW AND THEIR INHERITED PROBLEMS

Two credit card data sets were utilised in this study to demonstrate the challenges: (i) overlapping class samples and (ii) unbalanced class distribution. The general learning algorithms have difficulty in handling these two issues and caused low detection rates for minority classes.

The first data set is the Credit card fraud (CCF) data set released by [71]. Credit card fraud is an act of gaining unlawful advantage such as performing a variety of unauthorised transactions using the victim's credit card account [72], [73]. This data set was formed during a big data mining and fraud detection research between Worldline and the Machine Learning Group of Université Libre de Bruxelles (ULB). The data set contains transactions made by European credit card holders. The data set has a total of 284,807 transactions and it is highly unbalanced with only 492 fraud transactions (0.173%). Further, the data set has a total number of 31 attributes, as shown in Table 1. Unfortunately, due to confidentiality, the details of certain attribute are not disclosed. Most attributes of the CCF data set exhibit the scenarios as follows. Fig. 10 (a) shows the pairwise relationship of attributes V20 and V14. The red crosses are samples of Class 1 (frauds), while the blue crosses represent samples of Class 0 (normal transactions). The class distribution is clearly unbalanced and the samples of both classes are overlapped. Fig. 10 (b) also displays the same scenario where the plot involves attributes V24 and V12.

The second data set is the credit card default payment (CCDP) data set [74], [75]. Default credit card payment refers to the failure of a credit card holder in performing the minimum amount of credit card repayment within the agreed period [76], [77]. This data set contains the payment data of credit card holders of a Taiwan bank from April 2005 to September 2005. The CCDP data set is also slightly unbalanced with a ratio of approximately 3:1 (non-default payment: default payment). It has a total of 30,000 payment instances. 23,364 instances belong to 'no' class (non-default payment next month), and 6,636 of them belong to the 'yes' class (default payment next month). This means that there are only 28% of default payment instances out of the whole

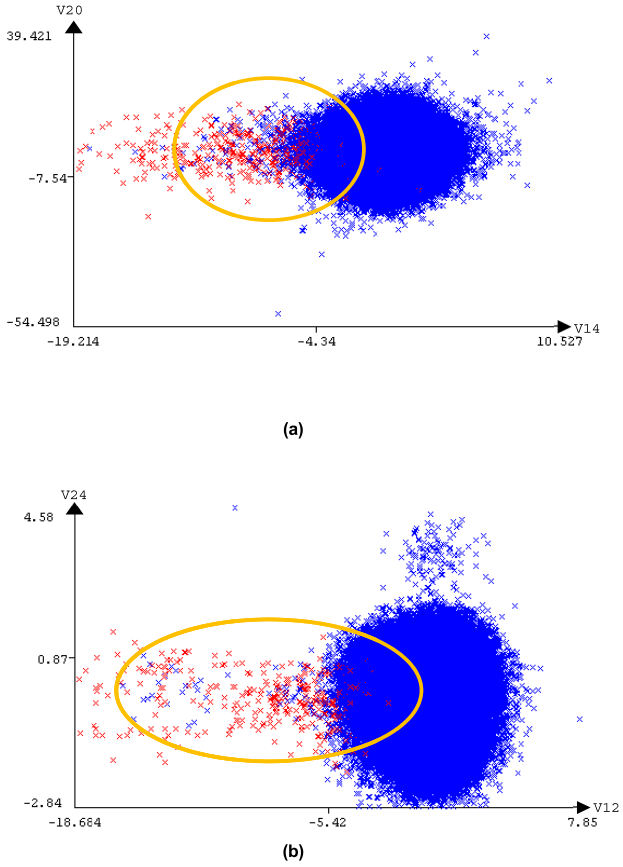

FIGURE 10. The visualisation of the CCF data set showing the unbalanced class distribution and the overlapping class problem between attributes (a) V20 vs. V14, and (b) V24 vs. V12.

payment data. This data set has a total of 25 attributes. The detail of each attribute is described in Table 2.

Fig. 11 (a) shows the scatter plot of attributes "Repayment status in April" and "Amount of previous statement in April". The red crosses are samples of class 'yes' (non-default payment), while the blue crosses are samples of class 'no' (default payment). The distribution of both classes is clearly unbalanced and we can see that the blue crosses are overwhelmed by the red crosses. Fig. 11 (b) shows the scatter plot of attributes "Bill statement in August" and "Repayment status in August".

Fig. 11 (a) and (b) show the overlapping samples of majority and minority classes. It is expected to be difficult for classifiers to accurately detect the minority classes (class 1 and class 'yes').

B. TACKLING THE PROBLEM USING MCS

MCS is the combination of predictions from a set of classifiers to produce a better prediction. MCS employs decision combination strategies that are able to produce a more robust, reliable, efficient recognition and accurate classification [55]–[58], [85].

Basically, there are three different combination strategies when employing MCS: (i) sequential, (ii) parallel, and (iii) hybrid. Among the three combination strategies of MCS,

TABLE 2. The attributes of the CCDP data set.

No.	Attribute(s)	Description
1	ID	The ID of each client
2	LIMIT_BAL	Amount of given credit in NT dollars (includes individual and family/supplementary credit
3	SEX	Gender (1=male, 2=female)
4	EDUCATION	(1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown)
5	MARRIAGE	Marital status (1=married, 2=single, 3=others)
6	AGE	Age in years
7	PAY_0	Repayment status in September, 2005 (-1=pay duly, 1=payment delay for one month, 2=payment delay for two months, … 8=payment delay for eight months, 9=payment delay for nine months and above)
8	PAY_2	Repayment status in August, 2005 (scale same as above)
9	PAY_3	Repayment status in July, 2005 (scale same as above)
10	PAY_4	Repayment status in June, 2005 (scale same as above)
11	PAY_5	Repayment status in May, 2005 (scale same as above)
12	PAY_6	Repayment status in April, 2005 (scale same as above)
13	BILL_AMT1	Amount of bill statement in September, 2005 (New Taiwan (NT) dollar)
14	BILL_AMT2	Amount of bill statement in August, 2005 (NT dollar)
15	BILL_AMT3	Amount of bill statement in July, 2005 (NT dollar)
16	BILL_AMT4	Amount of bill statement in June, 2005 (NT dollar)
17	BILL_AMT5	Amount of bill statement in May, 2005 (NT dollar)
18	BILL_AMT6	Amount of bill statement in April, 2005 (NT dollar)
19	PAY_AMT1	Amount of previous payment in September, 2005 (NT dollar)
20	PAY_AMT2	Amount of previous payment in August, 2005 (NT dollar)
21	PAY_AMT3	Amount of previous payment in July, 2005 (NT dollar)
22	PAY_AMT4	Amount of previous payment in June, 2005 (NT dollar)
23	PAY_AMT5	Amount of previous payment in May, 2005 (NT dollar)
24	PAY_AMT6	Amount of previous payment in April, 2005 (NT dollar)
25	default payment next month	Default payment (1=yes, 0=no)

we expect the sequential combination to perform well. Using the sequential combination, the output of the first classifier will be an input of the subsequent classifier. This means that the same piece of sample will be classified more than once. Taking credit card fraud data set (CCFD) as the example, our primary concern will be the frauds that are misclassified as normal transactions. Therefore, the sample classified as normal by the first classifier will be fed to the subsequent classifier for re-classification purpose. Further, having one classifier for each class is an advantage. Since each of the classes shall be handled by one classifier, then it will mitigate the effect of the unbalanced class distribution and

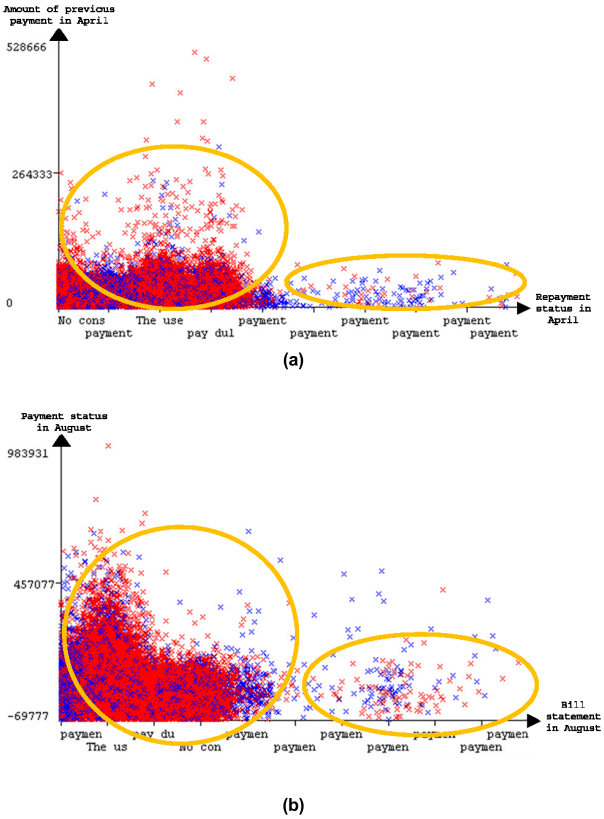

FIGURE 11. The visualisation of the CCDP data set showing the unbalanced class distribution and overlapping class problem between attributes (a) "Amount of previous statement in April" vs. "Repayment status in April", and (b) "Repayment status in August" vs. "Amount of bill statement in August".

overlapping classes. Employing the MCS with sequential combination will therefore reduce the overall misclassification rate and improve the detection rate, particularly for the minority class. Algorithm 1 shows the pseudocode of the proposed MCS that utilises the sequential combination scheme.

The credit card data set shall be classified by classifier *C1* (Line 1, Algo. 1). Classifier *C1* is the expert classifier to classify majority class samples. During the classification, if the sample is classified as 1 or 'yes', then the sample will be stored in the data set *F_ds*. Conversely, if the sample is classified as 0 or 'no', then the sample will be stored in a different data set called *N_ds* (Line 2 – 8, Algo. 1). Our primary concern for this project is the samples of class 1 or 'yes' that are misclassified as 0 or 'no'. To address the concern, we will re-classify the data by feeding *N_ds* into classifier *C2* (Line 9, Algo. 1). Classifier *C2* is the expert classifier to classify minority class samples. During the classification, if the sample is classified as 1 or 'yes', then the sample will be stored in data set *F_ds*. If the sample is classified as 0 or 'no', then the sample will remain in the data set *N_ds* (Line 10 – 14, Algo. 1). Subsequently, we combine *F_ds* and *N_ds* into one data set called *C_ds* (Line 15, Algo. 1).Finally, a confusion matrix as per Table 3 shall be generated based on *C_ds* (Line 16 – 19, Algo. 1). Once the confusion matrix

Algorithm 1 *Detect_Anomaly* (*CC_db*)

Input: The credit card data set, *CC_db* that comprises of *x* features for each credit card transaction
Output: A confusion matrix of classified data

1. classify *CC_db* with classifier *C1* // *C1* classifier to classify // majority class samples
2. for each transaction, *s* in the *CC_db* do
3. if *C1*. class(*s*) is equal to 1 or 'yes' // it is a fraud transaction
4. assign *s* to *F_ds* // *F_ds* is a data set to keep 1 or 'yes' data
5. else {*C1*. Class(*s*) is or 'no'} // it is a normal transaction
6. assign *s* to *N_ds* // *N_ds* is a data set to keep or 'no' data
7. end if
8. end for
9. classify *N_ds* with classifier, *C2* // *C2* classifier to classify minority // class samples
10. for each transaction, *sn* in the *N_ds* do
11. if *C2*. class(*sn*) is equal to 1 or 'yes'
12. assign *sn* to *F_ds*
13. end if
14. end for
15. combine *F_ds* and *N_ds* assign to *C_ds* // *C_ds* are combination of // dataset *F_ds* and *N_ds*
16. *actual* = *C_ds* //actual class
17. *predicted* = *C_ds* //predicted class
18. *result* ← confusion matrix (*actual,predicted*)
19. return *result*
20. *TPRmin* ← Calculate TPRmin(*result*) // calculate the TPR for // minority class samples
21. *TPRmaj* ←Calculate TPRmaj(*result*) // calculate the TPR for // majority class samples
22. return *TPRmin, TPRmaj*

TABLE 3. The Confusion matrix for the classified data.

(Predicted) yes/1	(Predicted) no/0	
TP	FN	(Actual) yes/1
FP	TN	(Actual) no/0

is obtained, the True Positive Rate (TPR) for both majority and minority classes shall be calculated using (5) and (6) (Line 20 – 22, Algo. 1).

To implement Algorithm 1, we need to identify the expert classifiers, *C1* and *C2*. An experiment had been conducted to identify them. A few popular single classifiers, namely, Naïve Bayes (NB), C4.5, Random Forest, Random Tree, Logistic Regression (LR), Multilayer Perceptron (MLP), and IBk were tested on both CCF and CCDP data sets. The experiments were evaluated using TPR. The TPR for both majority and minority classes can be calculated using (5) and (6).

$$TPR\,(minority) = \frac{Num.\ of\ detected\ frauds\ or\ default\ payments}{Total\ frauds\ or\ default\ payments} \quad (5)$$

$$TPR\,(majority) = \frac{Num.\ of\ detected\ non\ frauds\ or\ non\ default\ payments}{Total\ non\ frauds\ or\ non\ default\ payments} \quad (6)$$

Apart from TPR, Area under the ROC Curve (AUC) were also used as the performance metric. In general, AUC tells the capability of a classifier in differentiating classes. The closer the AUC value to 1, the better a classifier is in distinguishing between classes.

TABLE 4. The experimental results of using single classifiers. on both CCF and CCDP data sets.

CCF Data set					
Classifier	**TPR (0)**	**FPR (0)**	**TPR (1)**	**FPR (1)**	**AUC**
Naïve Bayes	0.978	0.171	**0.829**	0.022	0.960
C4.5 (C:0.025 M:2)	1.000	0.222	0.778	0.000	0.871
C4.5 (C:0.0325 M:2)	**1.000**	0.217	0.783	0.000	0.879
Random Forest	1.000	0.224	0.776	0.000	0.951
Random Tree	1.000	0.232	0.768	0.000	0.884
Logistic	1.000	0.376	0.624	0.000	0.975
Multilayer Perceptron	1.000	0.191	0.809	0.000	0.955
K-Nearest Neighbours	1.000	0.213	0.787	0.000	0.891
CCDP Data set					
Classifier	**TPR (no)**	**FPR (no)**	**TPR (yes)**	**FPR (yes)**	**AUC**
Naïve Bayes	0.634	0.281	**0.719**	0.366	0.745
C4.5 (C:0.25 M:2)	0.941	0.655	0.345	0.059	0.677
C4.5 (C:0.09 M:2)	**0.955**	0.653	0.347	0.045	0.703
Random Forest	0.939	0.623	0.377	0.061	0.759
Random Tree	0.820	0.588	0.412	0.180	0.619
Logistic	0.952	0.642	0.358	0.048	0.767
Multilayer Perceptron	0.922	0.596	0.404	0.078	0.714
K-Nearest Neighbours	0.827	0.601	0.399	0.173	0.615

**C4.5 Parameters:*

- *C - The confidence factor that is used for pruning where the smaller the values, the more pruning will be incurred.*
- *M The minimum number of instances per leaf.*

As shown in Table 4, most classifiers including C4.5 were able to achieve the perfect TPR for the majority class (class 0) of the CCF data set. C4.5 also scored the highest TPR, 0.955, for the majority class (class 'no') of the CCDP data set. Therefore, we identified C4.5 as the *C1* as it is able to produce high TPRs for the majority class of both data sets.

On the other hand, NB obtained the highest TPR, 0.829, for the minority class (class 1) of the CCF data set. NB also obtained the highest TPR for the minority class (class yes), 0.719, of the CCDP data set. Therefore, we identified NB as C2 as it is able to produce high TPR for the minority class of both data sets.

We employed sequential combination in our proposed MCS, with C4.5 and NB as the first and second expert classifiers. The results shall be discussed in the next section.

IV. RESULTS & DISCUSSION

A. DETECTION RESULTS USING SINGLE CLASSIFIERS

Based on our studies in the literature review, it was found that single classifiers are weak against classifying data sets that contain unbalanced class distribution and overlapping classes. This is proven by our experiment using some popular single classifiers, as shown in Table 4.

The majority class of the CCF data set was classified perfectly, with a TPR of 1.000, by most of the single classifiers. As for the minority class, the TPRs were mostly just average. The highest TPR for class 1 is 0.829, which was achieved by using NB.

On the other hand, the TPRs for the majority class of the CCDP data set were quite promising, except for NB which

TABLE 5. The comparison between the proposed MCS and the other researchers' work on the CCF data set.

Work	Approach	Base Learner	Performance Evaluation		
			Accuracy	TPR	TNR
Our Approach	Multiple classifier	C4.5+NB	0.999	**0.872**	**1.000**
Randhawa et al. [80]	AdaBoost + Majority Voting	NN+NB	0.999	0.789	0.999
Sohony et al. [61]	Multiple classifier	RF+FFNN	0.999	0.867	-
Xenopoulos, P. [82]	Ensemble Learning	Deep Belief Network	0.906	0.818	0.995

TABLE 6. The comparison between the proposed MCS and the other researchers' work on the CCDP data set.

Work	Approach	Base Learner	Performance Evaluation		
			Accuracy	TPR	TNR
Our Approach	Multiple classifier	C4.5+NB	**0.930**	**0.840**	**0.955**
Xia et al. [62]	Boosting (XGBoost)	CART	0.694	-	-
Singh [63]	Bagging	RF	0.816	0.371	-
Venkatesth & Jacob [78]	RF Ensemble learning method	RF	-	0.816	-
Charleonnan [79]	Multiple Classifiers	MLP+RBF+NB	-	0.534	0.831

achieved only 0.634. As for the minority class, the TPRs were generally low, except for NB that scored an average TPR of 0.719.

In general, the single classifiers did not perform well in detecting the minority class in both CCF and CCDP data sets.

B. DETECTION RESULTS USING THE PROPOSED MCS

Table 5 shows the comparison between our approach and the other researcher' approach in classifying CCF data set. All the other researchers' work listed above used ensemble approaches in tackling the unbalanced class distribution. By using our proposed MCS, we managed to achieve the highest TPR of 0.872 for the minority class and outperformed the other researchers'' work. Our proposed approach also gave a good accuracy of 0.999 and a TNR of 1.000.

Our MCS was also tested on CCDP data set. Table 6 shows the comparison between our approach and the other researchers' work. We outperformed their work by obtaining the highest TPR for the minority class, which is 0.840. Our proposed approach also achieved an accuracy of 0.930 and a TNR of 0.955, which are better than their work.

In summary, we can conclude that our proposed approach is able to tackle the unbalanced class distribution and the overlapping class samples that exists in both credit card data sets.

V. CONCLUSION

Credit card is one alternative of cash payment. Some card holders may abuse their responsibility in credit card usage and repayment. Apart from that, credit card transaction is also prone to fraudulent where unauthorized parties perform illegal transactions using credit cards. Therefore, it is the responsibility of card issuers or the banks to find an effective way to reduce the cost that may incur when the issues above happen. One way to address these issues is via data mining. Due to the characteristics such as overlapping class samples and unbalanced class distribution that exist in credit card data sets, it gives challenges to data mining researchers. On top of that, the weakness of general learning algorithms also contributes to the difficulties of classifying the minority class, which is usually the important class, of the data sets.

This study proposed a MCS to tackle the issues as discussed above. Based on our analysis using single classifiers, we found that C4.5 is the expert in classifying the majority class samples and NB is the expert in classifying the minority class samples. Therefore, they were arranged sequentially in our proposed MCS to detect credit card anomalies. Our proposed MCS was evaluated using two different credit card data sets: CCF and CCDP. We have compared our work with the other researchers' work. The experimental results showed that the proposed MCS outperformed their work. In general, our proposed MCS demonstrates its superiority in handling the credit data sets that inherit the characteristics of overlapping classes and unbalanced class distribution. However, there are rooms to improve the TPR for the minority classes. We are looking into other MCS combination strategies for our future work, particularly the hybrid combination. Currently, researchers had attempted deep learning algorithms such as Long Short-term Memory (LSTM) and Deep Belief Networks for detecting anomalies in credit card transactions [86]. We are also considering combining the deep learning algorithms, as in the study of [87], for promising detection results.

REFERENCES

[1] *Nilsonreport*. Accessed: 2019. [Online]. Available: https://nilsonreport.com/upload/content_promo/The_Nilson_Report_10-17-2016.pdf
[2] *RM51.3mil in payment card fraud losses reported in 2016*. Accessed: 2019. [Online]. Available: https://www.thestar.com.my/news/nation/2017/08/03/rm422mil-in-credit-card-fraud-losses-reported-in-2016

[3] T. Lokman. *3.6 Million Credit Card Holders Have RM36.9 Billion Outstanding Balance*. NST Online. Accessed: Dec. 2019. [Online]. Available: https://www.nst.com.my/news/nation/2017/08/270620/36-million-credit-card-holders-have-rm369-billion-outstanding-balance

[4] Ceicdata. (2019). *Malaysia Credit Card Statistics*. Accessed: Dec. 2019. [Online]. Available: https://www.ceicdata.com/en/malaysia/credit-card-statistics

[5] A. O. Adewumi and A. A. Akinyelu, "A survey of machine-learning and nature-inspired based credit card fraud detection techniques," *Int. J. Syst. Assur. Eng. Manag.*, vol. 8, no. S2, pp. 937–953, Nov. 2017, doi: 10.1007/s13198-016-0551-y.

[6] S. Makki, "Fraud analysis approaches in the age of big data—A review of state of the art," in *Proc. IEEE 2nd Int. Workshops Found. Appl. Self Syst. (FASW)*, 2017, pp. 243–250, doi: 10.1109/fas-w.2017.154.

[7] G. M. Weiss, "Mining with rarity," *SIGKDD Explor. Newsl.*, vol. 6, no. 1, p. 7, Jun. 2004, doi: 10.1145/1007730.1007734.

[8] S. Maheshwari, J. Agrawal, and S. Sharma, "A new approach for classification of highly imbalanced datasets using evolutionary algorithms," *Int. J. Sci. Eng. Res.*, vol. 2, no. 7, pp. 1–5, 2011.

[9] Q. Dong, S. Gong, and X. Zhu, "Class rectification hard mining for imbalanced deep learning," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1851–1860. Accessed: Aug. 2018, doi: 10.1109/iccv.2017.205.

[10] J. Gao, L. Gong, J. Y. Wang, and Z. C. Mo, "Study on unbalanced binary classification with unknown misclassification costs," in *Proc. IEEE Int. Conf. Ind. Eng. Eng. Manage. (IEEM)*, Dec. 2018, pp. 1538–1542, doi: 10.1109/ieem.2018.8607671.

[11] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study1," *IDA*, vol. 6, no. 5, pp. 429–449, Nov. 2002, doi: 10.3233/ida-2002-6504.

[12] J. Mathew, C. K. Pang, M. Luo, and W. H. Leong, "Classification of imbalanced data by oversampling in kernel space of support vector machines," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 9, pp. 4065–4076, Sep. 2018, doi: 10.1109/tnnls.2017.2751612.

[13] P. Vuttipittayamongkol, E. Elyan, A. Petrovski, and C. Jayne, "Overlap-based undersampling for improving imbalanced data classification," in *Proc. Intell. Data Eng. Automated Learn. (IDEAL)*, 2018, pp. 689–697, doi: 10.1007/978-3-030-03493-1_72.

[14] N. Chawla, N. Japkowicz, and A. Kotcz, "Editorial," *ACM SIGKDD Explor. Newslett.*, vol. 6, no. 1, p. 1, 2004, doi: 10.1145/1007730.1007733.

[15] A. Fernández, S. Del Río, N. V. Chawla, and F. Herrera, "An insight into imbalanced Big Data classification: Outcomes and challenges," *Complex Intell. Syst.*, vol. 3, no. 2, pp. 105–120, Jun. 2017, doi: 10.1007/s40747-017-0037-9.

[16] V. H. Barella, L. P. F. Garcia, M. P. De Souto, A. C. Lorena, and A. De Carvalho, "Data complexity measures for imbalanced classification tasks," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2018, pp. 1–8, doi: 10.1109/ijcnn.2018.8489661.

[17] G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue, and G. Bing, "Learning from class-imbalanced data: Review of methods and applications," *Expert Syst. Appl.*, vol. 73, pp. 220–239, May 2017, doi: 10.1016/j.eswa.2016.12.035.

[18] A. G. De Sá, A. C. Pereira, and G. L. Pappa, "A customized classification algorithm for credit card fraud detection," *Eng. Appl. Artif. Intell.*, vol. 72, pp. 21–29, Jun. 2018, doi: 10.1016/j.engappai.2018.03.011.

[19] Y. Yong, "The research of imbalanced data set of sample sampling method based on K-means cluster and genetic algorithm," *Energy Procedia*, vol. 17, pp. 164–170, 2012, doi: 10.1016/j.egypro.2012.02.078.

[20] J. Błaszczyński and J. Stefanowski, "Neighbourhood sampling in bagging for imbalanced data," *Neurocomputing*, vol. 150, pp. 529–542, Feb. 2015, doi: 10.1016/j.neucom.2014.07.064.

[21] K. Napierała, J. Stefanowski, and S. Wilk, "Learning from imbalanced data in presence of noisy and borderline examples," in *Rough Sets and Current Trends in Computing*. Brookline, MA, USA: Microtome Publishing, 2010, pp. 158–167, doi: 10.1007/978-3-642-13529-3_18.

[22] G. Lemaitre, F. Nogueira, and C. Aridas, "Imbalanced-learn: A Python toolbox to tackle the curse of imbalanced datasets in machine learning," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 1–5, 2017.

[23] A. Engelbrecht and H. Viktor, *Rule Improvement Through Decision Boundary Detection Using Sensitivity Analysis* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 1999, pp. 78–84, doi: 10.1007/bfb0100474.

[24] Y. Qi. *A Brief Literature Review of Class Imbalanced Prob-LEM*. Accessed: Mar. 2019. [Online]. Available: http://nyc.lti.cs.cmu.edu/IRLab/11-743f04/qyj/IR-pages/ImbalancedSummary.html

[25] M. Goldstein and S. Uchida, "A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data," *PLoS ONE*, vol. 11, no. 4, Apr. 2016, Art. no. e0152173, doi: 10.1371/journal.pone.0152173.

[26] B. Krawczyk, "Learning from imbalanced data: Open challenges and future directions," *Prog. Artif. Intell.*, vol. 5, no. 4, pp. 221–232, Nov. 2016, doi: 10.1007/s13748-016-0094-0.

[27] S. Boughorbel, F. Jarray, and M. El-Anbari, "Optimal classifier for imbalanced data using Matthews correlation coefficient metric," *PLoS ONE*, vol. 12, no. 6, Jun. 2017, Art. no. e0177678, doi: 10.1371/journal.pone.0177678.

[28] I. Brown and C. Mues, "An experimental comparison of classification algorithms for imbalanced credit scoring data sets," *Expert Syst. Appl.*, vol. 39, no. 3, pp. 3446–3453, Feb. 2012, doi: 10.1016/j.eswa.2011.09.033.

[29] T. Fitzpatrick and C. Mues, "An empirical comparison of classification algorithms for mortgage default prediction: Evidence from a distressed mortgage market," *Eur. J. Oper. Res.*, vol. 249, no. 2, pp. 427–439, Mar. 2016, doi: 10.1016/j.ejor.2015.09.014.

[30] T. Li, J. Li, Z. Liu, P. Li, and C. Jia, "Differentially private naive Bayes learning over multiple data sources," *Inf. Sci.*, vol. 444, pp. 89–104, May 2018, doi: 10.1016/j.ins.2018.02.056.

[31] G. Nanda, K. Vallmuur, and M. Lehto, "Improving autocoding performance of rare categories in injury classification: Is more training data or filtering the solution?" *Accident Anal. Prevention*, vol. 110, pp. 115–127, Jan. 2018, doi: 10.1016/j.aap.2017.10.020.

[32] S. Salzberg, "C4.5: Programs for machine learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993," *Mach. Learn.*, vol. 16, no. 3, pp. 235–240, 1994, doi: 10.1023/a:1022645310020.

[33] S. Sharma, J. Agrawal, and S. Sharma, "Classification through machine learning technique: C4. 5 algorithm based on various entropies," *Int. J. Comput. Appl.*, vol. 82, no. 16, pp. 28–32, Nov. 2013, doi: 10.5120/14249-2444.

[34] H. Sani, C. Lei, and D. Neagu, *Computational Complexity Analysis of Decision Tree Algorithms* (Lecture Notes in Computer Science). Cham, Switzerland: Springer, 2018, pp. 191–197, doi: 10.1007/978-3-030-04191-5_17.

[35] J. Crowley. (2017). *Intelligent Systems: Reasoning and Recognition*. Accessed: Mar. 2018. [Online]. Available: http://www-prima.imag.fr/Prima/jlc/Courses/2016/ENSI2.SIRR/ENSI2.SIRR.S1.pdf

[36] S. Selvi, S. Sowmiya, and R. Sangeetha. (2018). *C5 Causal Decision Tree*. Accessed: Mar. 2018. [Online]. Available: http://ijsrcseit.com/paper/CSEIT1833100.pdf

[37] J. Lever, M. Krzywinski, and N. Altman, "Model selection and overfitting," *Nature Methods*, vol. 13, no. 9, pp. 703–704, Sep. 2016, doi: 10.1038/nmeth.3968.

[38] R. A. Watson and E. Szathmáry, "How can evolution learn?" *Trends Ecol. Evol.*, vol. 31, no. 2, pp. 147–157, Feb. 2016, doi: 10.1016/j.tree.2015.11.009.

[39] W. Liu, S. Chawla, D. A. Cieslak, and N. V. Chawla, "A robust decision tree algorithm for imbalanced data sets," in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2010, pp. 766–777, doi: 10.1137/1.9781611972801.67.

[40] A. Dal Pozzolo and G. Bontempi, "Adaptive machine learning for credit card fraud detection," Ph.D. dissertation, Université Libre de Bruxelles, Bruxelles, Belgium, 2015.

[41] F. Westerlund and P. Wijayatunga, "Credit card fraud detectioN (machine learning algorithms)," M.S. thesis, Ume Univ., Krong Battambang, Cambodia, 2017.

[42] O. Maimon and L. Rokach, *Data Mining and Knowledge Discovery Handbook*. Boston, MA, USA: Springer, 2010.

[43] R. Taghizadeh-Mehrjardi, K. Nabiollahi, B. Minasny, and J. Triantafilis, "Comparing data mining classifiers to predict spatial distribution of USDA-family soil groups in Baneh region, Iran," *Geoderma*, vols. 253–254, pp. 67–77, Sep. 2015, doi: 10.1016/j.geoderma.2015.04.008.

[44] D. Tien Bui, T. A. Tuan, H. Klempe, B. Pradhan, and I. Revhaug, "Spatial prediction models for shallow landslide hazards: A comparative assessment of the efficacy of support vector machines, artificial neural networks, kernel logistic regression, and logistic model tree," *Landslides*, vol. 13, no. 2, pp. 361–378, Apr. 2016, doi: 10.1007/s10346-015-0557-6.

[45] R. Nazir, E. Momeni, K. Marsono, and H. Maizir, "An artificial neural network approach for prediction of bearing capacity of spread foundations in sand," *J. Teknol.*, vol. 72, no. 3, p. 1, 2015, doi: 10.11113/jt.v72.4004.

[46] Y. L. Murphey, H. Guo, and L. A. Feldkamp, "Neural learning from unbalanced data," *Int. J. Speech Technol.*, vol. 21, no. 2, pp. 117–128, Sep. 2004, doi: 10.1023/b:apin.0000033632.42843.17.
[47] R. Akbani, S. Kwek, and N. Japkowicz, "Applying support vector machines to imbalanced datasets," *Mach. Learn.*, vol. 2004, pp. 39–50, Mar. 2004, doi: 10.1007/978-3-540-30115-8_7.
[48] G. Batista, R. Prati, and M. Monard, *Balancing Strategies and Class Overlapping* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2005, pp. 24–35, doi: 10.1007/11552253_3.
[49] S. Ullman and E. Sali, "Object classification using a fragment-based representation," in *Biologically Motivated Computer Vision*. Berlin, Germany: Springer, 2000, pp. 73–87, doi: 10.1007/3-540-45482-9_8.
[50] Z. Yang and D. Gao, "Classification for imbalanced and overlapping classes using outlier detection and sampling techniques," *Appl. Math. Inf. Sci.*, vol. 7, no. 1L, pp. 375–381, Feb. 2013, doi: 10.12785/amis/071l50.
[51] Y. Sahin and E. Duman, "Detecting credit card fraud by decision trees and support vector machines," in *Proc. Int. MultiConf. Eng. Comput. Scientists*, Hong Kong, 2011, pp. 1–6.
[52] K. R. Seeja and M. Zareapoor, "FraudMiner: A novel credit card fraud detection model based on frequent itemset mining," *Sci. World J.*, vol. 2014, pp. 1–10, May 2014, doi: 10.1155/2014/252797.
[53] M. Fahmi, A. Hamdy, and K. Nagati, "Data mining techniques for credit card fraud detection: Empirical study," in *Proc. Int. Conf. Sustain. Vital Technol. Eng. Inform.*, 2016, pp. 1–9.
[54] A. Pawar, S. Dongare, A. Deokate, H. Sangle, and P. Mokal, "Outlier detection using oversampling PCA for credit card fraud detection," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 5, no. 5, pp. 1–11, 2017.
[55] A. F. R. Rahman and M. C. Fairhurst, "Multiple classifier decision combination strategies for character recognition: A review," *Int. J. Document Anal. Recognit.*, vol. 5, no. 4, pp. 166–194, Jul. 2003, doi: 10.1007/s10032-002-0090-8.
[56] J. Stefanowski. (2008). *Lamsade.Dauphine.fr*. Accessed: Aug. 2018. [Online]. Available: http://www.lamsade.dauphine.fr/~projet_cost/ALGORITHMIC_DECISION_THEORY/pdf/Stefanowski/Stefanowski4.pdf
[57] M. Woźniak, M. Graña, and E. Corchado, "A survey of multiple classifier systems as hybrid systems," *Inf. Fusion*, vol. 16, pp. 3–17, Mar. 2014, doi: 10.1016/j.inffus.2013.04.006.
[58] A. A. Aburomman and M. B. I. Reaz, "A novel SVM-kNN-PSO ensemble method for intrusion detection system," *Appl. Soft Comput.*, vol. 38, pp. 360–372, Jan. 2016, doi: 10.1016/j.asoc.2015.10.011.
[59] H. Sayadi, N. Patel, S. P. D. , A. Sasan, S. Rafatirad, and H. Homayoun, "Ensemble learning for effective run-time hardware-based malware detection: A comprehensive analysis and classification," in *Proc. 55th ACM/ESDA/IEEE Design Automat. Conf. (DAC)*, 2018, Art. no. 1, doi: 10.1109/dac.2018.8465828.
[60] J. Brownlee. (2019). Bagging and random forest ensemble algorithms for machine learning. Machine Learning Mastery. Accessed: Dec. 2018. [Online]. Available: https://machinelearningmastery.com/bagging-and-random-forest-ensemble-algorithms-for-machine-learning/
[61] I. Sohony, R. Pratap, and U. Nambiar, "Ensemble learning for credit card fraud detection," in *Proc. ACM India Joint Int. Conf. Data Sci. Manage. Data (CoDS-COMAD)*, 2018, pp. 289–294. Accessed: Dec. 21, 2019, doi: 10.1145/3152494.3156815.
[62] Y. Xia, C. Liu, Y. Li, and N. Liu, "A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring," *Expert Syst. Appl.*, vol. 78, pp. 225–241, Jul. 2017, doi: 10.1016/j.eswa.2017.02.017.
[63] P. Singh, "Comparative study of individual and ensemble methods of classification for credit scoring," in *Proc. Int. Conf. Inventive Comput. Inform. (ICICI)*, Nov. 2017, pp. 968–972, doi: 10.1109/icici.2017.8365282.
[64] M. Zareapoor and P. Shamsolmoali, "Application of credit card fraud detection: Based on bagging ensemble classifier," *Procedia Comput. Sci.*, vol. 48, pp. 679–685, Dec. 2015, doi: 10.1016/j.procs.2015.04.201.
[65] M. Abedini, F. Ahmadzadeh, and R. Noorossana, "Customer credit scoring using a hybrid data mining approach," *Kybernetes*, vol. 45, no. 10, pp. 1576–1588, Nov. 2016, doi: 10.1108/k-09-2015-0228.
[66] A. Ali, S. Shamsuddin, and A. Ralescu. (2015). *Classification With Class Imbalance Problem: A Review*. Accessed: Aug. 2018. [Online]. Available: http://home.ijasca.com/data/documents/13IJASCA-070301_Pg176-204_Classification-with-class-imbalance-problem_A-Review.pdf
[67] S. Wang, L. L. Minku, and X. Yao, "A systematic study of online class imbalance learning with concept drift," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 4802–4821, Oct. 2018, doi: 10.1109/tnnls.2017.2771290.
[68] M. Bekkar, H. Djemaa, and T. Alitouche, "Evaluation measures for models assessment over imbalanced data sets," *J. Inf. Eng. Appl.*, vol. 3, no. 10, Apr. 2013.
[69] H. Hossin and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," *Int. J. Data Mining Knowl. Manage. Process*, vol. 5, no. 2, pp. 1–11, Mar. 2015, doi: 10.5121/ijdkp.2015.5201.
[70] J. Akosa, "Predictive accuracy: A misleading performance measure for highly imbalanced data," in *Proc. SAS Global Forum*, 2017, pp. 2–5.
[71] A. D. Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Calibrating probability with undersampling for unbalanced classification," in *Proc. IEEE Symp. Ser. Comput. Intell.*, Dec. 2015, pp. 159–166, doi: 10.1109/ssci.2015.33.
[72] S. Maes, K. Tuyls and B. Vanschoenwinkel, "Credit card fraud detection using Bayesian and neural networks," in *Proc. 1st Int. Naiso Congr. Neuro Fuzzy Technol.*, 2002, pp. 261–270.
[73] N. Carneiro, G. Figueira, and M. Costa, "A data mining based system for credit-card fraud detection in e-tail," *Decis. Support Syst.*, vol. 95, pp. 91–101, Mar. 2017, doi: 10.1016/j.dss.2017.01.002.
[74] K. Bache and M. Lichman. (2013). *UCI Machine Learning Reposito-ry*. Accessed: Mar. 2018. [Online]. Available: http://archive.ics.uci.edu/ml/index.php
[75] I. Yeh. (2016). *UCI Machine Learning Repository: Default of Credit Card Clients Data Set*. Accessed: Mar. 2018. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients
[76] L. Irby. (2018). *Learn How Credit Card Default Happens and What You Can Do About It*. Accessed: Mar. 2018. [Online]. Available: https://www.thebalance.com/what-is-credit-card-default-960209
[77] *Kuala Lumpur*, Credit Card, Bank Negara Malaysia, Kuala Lumpur, Malaysia, 2019.
[78] A. Venkatesh, and S. Gracia, "Prediction of credit-card defaulters: A comparative study on performance of classifiers," *Int. J. Comput. Appl.*, vol. 145, no. 7, pp. 36–41, 2016, doi: 10.5120/ijca2016910702.
[79] A. Charleonnan, "Credit card fraud detection using RUS and MRN algorithms," in *Proc. Manage. Innov. Technol. Int. Conf. (MITicon)*, Oct. 2016, pp. 1–73, doi: 10.1109/miticon.2016.8025244.
[80] K. Randhawa, C. K. Loo, M. Seera, C. P. Lim, and A. K. Nandi, "Credit card fraud detection using AdaBoost and majority voting," *IEEE Access*, vol. 6, pp. 14277–14284, 2018, doi: 10.1109/access.2018.2806420.
[81] H. Wang, P. Zhu, X. Zou, and S. Qin, "An ensemble learning framework for credit card fraud detection based on training set partitioning and clustering," in *Proc. IEEE SmartWorld, Ubiquitous Intell. Comput., Adv. Trusted Comput., Scalable Comput. Commun., Cloud Big Data Comput., Internet People Smart City Innov. (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, 2018, pp. 94–98, doi: 10.1109/smartworld.2018.00051.
[82] P. Xenopoulos, "Introducing DeepBalance: Random deep belief network ensembles to address class imbalance," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 3684–3689, doi: 10.1109/bigdata.2017.8258364.
[83] A. Salazar, G. Safont, A. Soriano, and L. Vergara, "Automatic credit card fraud detection based on non-linear signal processing," in *Proc. IEEE Int. Carnahan Conf. Secur. Technol. (ICCST)*, Oct. 2012, pp. 207–212, doi: 10.1109/ccst.2012.6393560.
[84] A. Salazar, G. Safont, and L. Vergara, "Semi-supervised learning for imbalanced classification of credit card transaction," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2018, pp. 1–7, doi: 10.1109/ijcnn.2018.8489755.
[85] A. Soriano, L. Vergara, B. Ahmed, and A. Salazar, "Fusion of scores in a detection context based on alpha integration," *Neural Comput.*, vol. 27, no. 9, pp. 1983–2010, Sep. 2015, doi: 10.1162/neco_a_00766.
[86] A. Roy, J. Sun, R. Mahoney, L. Alonzi, S. Adams, and P. Beling, "Deep learning detecting fraud in credit card transactions," in *Proc. Syst. Inf. Eng. Design Symp. (SIEDS)*, 2018, pp. 129–134.
[87] D. Bermejo-Peláez, S. Ash, G. Washko, R. S. J. Estépar, and M. Ledesma-Carbayo, "Classification of interstitial lung abnormality patterns with an ensemble of deep convolutional neural networks," *Sci. Rep.*, vol. 10, no. 1, pp. 1–5, 2020, doi: 10.1038/s41598-019-56989-5.

SURAYA NURAIN KALID was born in Kuala Terengganu, Terengganu, Malaysia, in 1990. She received the Diploma degree in IT and the bachelor's degree in IT majoring in software engineering from Multimedia University, Malaysia, in 2010 and 2014, respectively, where she is currently pursuing the master's degree.

From 2014 to 2016, she was an Assistant Manager Core Network, responsible for assisting the upgrading of IP infrastructure for Telekom Malaysia's backbone. Then, she was an IT Executive assigned to TM's subsidiary Fiberail under the Network Management Centre. Since November 2016, she has been an Assistant Lecturer with Multimedia University, Malaysia. Her research interests include artificial intelligence, data mining, and software development.

KENG-HOONG NG is currently a Lecturer with the Faculty of Computing and Informatics, Multimedia University. He has been with the University, as an Academic Staff, since 2002. He enjoys lecturing, provides training (CCNA professional course), and doing research. For his personal interest, he likes to travel, exercise (badminton), and find undervalued stocks in his spare time. His research interests include data clustering, data classification, bioinformatics, and computer networks.

GEE-KOK TONG currently holds a lecturer position with the Faculty of Computing and Informatics, Multimedia University, Cyberjaya, Malaysia. He has been conducting research work in financial econometric time series, such as vector error correction model, GARCH modeling, extreme value theory, copula, value-at-risk, conditional value-at-risk, risk adjusted performance measures, portfolio management, and Monte Carlo simulation.

KOK-CHIN KHOR received the B.Sc. and M.Sc. degrees from Universiti Putra Malaysia (UPM), and the Ph.D. degree from Multimedia University (MMU), Malaysia. Apart from teaching, he is also a certified Instructor for Cisco professional networking courses. His research interests include data mining and computer networking. Prior to joining the academic sector, he worked as a Web Programmer and System Analyst. He was the Chair of a few special sessions of international research conferences, as well as the reviewer for a numbers of journals and conferences.