IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 14, 2022, accepted January 27, 2022, date of publication January 31, 2022, date of current version February 15, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3148298

A Neural Network Ensemble With Feature Engineering for Improved Credit Card Fraud Detection

EBENEZER ESENOGHO[1], (Member, IEEE), IBOMOIYE DOMOR MIENYE[2], (Member, IEEE), THEO G. SWART[1], (Senior Member, IEEE), KEHINDE ARULEBA[3], AND GEORGE OBAIDO[4]

[1]Center for Telecommunications, Department of Electrical and Electronic Engineering Science, University of Johannesburg, Johannesburg 2006, South Africa
[2]Department of Electrical and Electronic Engineering Science, University of Johannesburg, Johannesburg 2006, South Africa
[3]School of Informatics, University of Leicester, Leicester LE1 7RH, U.K.
[4]Department of Computer Science and Engineering, University of California at San Diego, San Diego, CA 92093, USA

Corresponding author: Ebenezer Esenogho (ebenezere@uj.ac.za)

ABSTRACT Recent advancements in electronic commerce and communication systems have significantly increased the use of credit cards for both online and regular transactions. However, there has been a steady rise in fraudulent credit card transactions, costing financial companies huge losses every year. The development of effective fraud detection algorithms is vital in minimizing these losses, but it is challenging because most credit card datasets are highly imbalanced. Also, using conventional machine learning algorithms for credit card fraud detection is inefficient due to their design, which involves a static mapping of the input vector to output vectors. Therefore, they cannot adapt to the dynamic shopping behavior of credit card clients. This paper proposes an efficient approach to detect credit card fraud using a neural network ensemble classifier and a hybrid data resampling method. The ensemble classifier is obtained using a long short-term memory (LSTM) neural network as the base learner in the adaptive boosting (AdaBoost) technique. Meanwhile, the hybrid resampling is achieved using the synthetic minority oversampling technique and edited nearest neighbor (SMOTE-ENN) method. The effectiveness of the proposed method is demonstrated using publicly available real-world credit card transaction datasets. The performance of the proposed approach is benchmarked against the following algorithms: support vector machine (SVM), multilayer perceptron (MLP), decision tree, traditional AdaBoost, and LSTM. The experimental results show that the classifiers performed better when trained with the resampled data, and the proposed LSTM ensemble outperformed the other algorithms by obtaining a sensitivity and specificity of 0.996 and 0.998, respectively.

INDEX TERMS AdaBoost, credit card, data resampling, fraud detection, LSTM, machine learning.

I. INTRODUCTION

In the past decade, there has been a rise in e-commerce, which has increased credit card utilization significantly. The increasing credit card usage has brought about a constant increase in fraudulent transactions [1]. Fraudulent credit card transactions have severely impacted the financial industry. A recent report showed that about 27.85 billion dollars were lost to credit card fraud in 2018, a 16.2% increase compared to the 23.97 billion dollars lost in 2017, and it is estimated to reach 35 billion dollars by 2023 [2]. These losses can be reduced through efficient fraud monitoring and prevention. Meanwhile, machine learning (ML) has been applied to develop several credit card fraud detection systems [3]–[7]. However, credit card fraud detection remains a challenge from a learning perspective due to the class imbalance that exists in the datasets [4]. Though the class imbalance is not the only problem that has hindered credit card fraud detection, it is the most critical challenge [6]. The class imbalance is a problem that occurs in several real-world ML applications, where datasets have an uneven class distribution. For example, samples belonging to one class (the majority class) are higher than those of the other class (the minority class). Most credit card transaction datasets are imbalanced because the legitimate transactions significantly outnumber the fraudulent transactions [8]. Most traditional ML algorithms perform well when they are trained with balanced data. The skewed class distribution makes conventional ML

The associate editor coordinating the review of this manuscript and approving it for publication was Joey Tianyi Zhou.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

algorithms have biased performance towards the majority class because the algorithms are not designed to consider the class distribution but the error rate [9]. Therefore, more minority class examples are misclassified than the majority class samples [10].

The methods used in the literature to classify imbalanced data can be grouped into three categories, including data-level, algorithm-level, and hybrid techniques. Data-level techniques tend to create a balanced dataset by undersampling the majority class or oversampling the minority class, sometimes the combination of both [11]. Algorithm-level methods aim to solve the class imbalance problem by modifying the classifier to give more attention to the minority class examples. Examples of algorithm-level techniques include ensemble learning and cost-sensitive learning methods [12]. Meanwhile, the hybrid methods combine both data-level and algorithm-level techniques.

Several research works have proposed different methods to handle the imbalanced class problem in credit card fraud detection. For example, Padmaja *et al.* [13] proposed a fraud detection method using k-reverse nearest neighbor (KRNN) to eliminate extreme outliers from the minority class samples. Secondly, hybrid resampling was performed on the dataset, i.e., undersampling of the majority class and the oversampling of the minority class. The resampled data was used to train several classifiers, including the naïve Bayes, C4.5 decision tree, and k-nearest neighbor (KNN) classifiers. Compared to traditional resampling methods, the proposed approach obtained superior performance.

Taha and Malebary [14] proposed a credit card fraud detection method using a light gradient boosting machine (LightGBM). The hyperparameters of the LightGBM were tuned using a Bayesian-based optimization algorithm. The technique achieved an accuracy of 98.40% and a precision of 97.34%. Furthermore, Randhawa *et al.* [1] studied the performance of some standard machine learning algorithms and hybrid classifiers, including ensemble classifiers based on majority voting. The experimental results show that the majority voting technique yields excellent performance in detecting fraudulent transactions.

Despite the numerous studies proposed to handle imbalanced data, this problem remains a challenge, especially in credit card fraud detection [6]. Since the advent of deep learning, recurrent neural networks (RNN), such as long short-term memory (LSTM) and gated recurrent units (GRU), have shown enormous potential in modelling sequential data [15]–[17]. Conventional machine learning algorithms have not been successful in credit card fraud detection because they do not adapt to the dynamic shopping trends of credit card clients, which results in misclassifications when used for fraud detection systems [18]. To address this problem and proffer a robust solution that models the time series in credit card transactions, this study employs the LSTM neural network. The rationale behind this study is that it can be more beneficial to consider the entire sequence of transactions rather than only individual transactions because a method capable of modelling time in credit card data will be more powerful in identifying small shifts in legitimate customer shopping behavior.

The contribution of this study is the development of a robust credit card fraud detection method using an LSTM neural network ensemble. In the process, we implement an effective feature engineering method via resampling of the imbalanced data using the SMOTE-ENN technique. The proposed ensemble technique uses the LSTM neural network as the base learner in the adaptive boosting (AdaBoost) algorithm. This method is significant for two reasons: the LSTM is a robust algorithm for modelling sequential data. Secondly, the AdaBoost technique builds strong classifiers that are less likely to overfit, with lesser false-positive predictions [19]. Hence, integrating the LSTM neural network and AdaBoost algorithm could be an excellent method for effective credit card fraud detection.

The rest of this paper is structured as follows: Section II discusses the credit card fraud detection dataset, together with the conventional AdaBoost and LSTM techniques. Section III presents the proposed credit card fraud detection system, including the feature engineering and the LSTM ensemble. Section IV presents the results and discussions, while Section V concludes the paper and provides future research direction.

II. BACKGROUND

A. DATASET

This research utilizes the well-known credit card fraud detection dataset [20]. The dataset was prepared by the Université Libre de Bruxelles (ULB) Machine Learning Group on big data mining and fraud detection [9]. The dataset contains credit card transactions performed within two days in September 2013 by European credit card clients. The dataset is imbalanced, with only 492 fraudulent transactions out of 284 807. Meanwhile, all the attributes except ''Time'' and ''Amount'' are numerical due to the transformation carried out on the dataset, and they are coded as $V_1, V_2, \ldots, V_{28}$ for confidentiality reasons. The ''Amount'' attribute is the cost of the transaction and the ''Time'' attribute is the seconds that elapsed between a transaction and the first transaction in the dataset. Lastly, the attribute ''Class'' is the dependent variable, and it has a value of 1 for fraudulent transactions and 0 for legitimate transactions.

B. ADAPTIVE BOOSTING

The AdaBoost algorithm [21] is an ensemble technique used to build strong classifiers by voting the weighted predictions of the weak learners [22]. It has achieved excellent performance in several applications, including credit card fraud detection [1] and intrusion detection systems [23]. Overfitting is common in machine learning applications [12], leading to poor classification performance. However, classifiers trained using the AdaBoost technique are less likely to overfit and also, the risk of high false-positive predictions

is reduced [19]. In the AdaBoost implementation, an algorithm is selected to train the base classifier using the initial input data. Secondly, the weights of the samples are adjusted, and more weight is given to the misclassified samples. Furthermore, the adjusted instances are employed to train the subsequent base learner, which attempts to correct the misclassifications from the previous models. The iteration continues until the specified number of models is built, or there are no misclassified samples in the data.

C. LONG SHORT-TERM MEMORY NEURAL NETWORK

Long Short-Term Memory neural network is a special type of recurrent neural network (RNN) that has achieved excellent performance in learning long-term dependencies and avoids the gradient disappearance problem [24]. LSTM consists of a memory cell c_t to remember the previous information and three types of gates that controls how the historical information is used and processed. The three gates are forget gate f_t, input gate i_t, and output gate o_t. The LSTM layers are updated using the following equations:

$$i_t = \sigma(V_i x_t + W_i h_{t-1} + b_i) \quad (1)$$
$$f_t = \sigma(V_f x_t + W_f h_{t-1} + b_f) \quad (2)$$
$$\tilde{c}_t = tanh(V_c x_t + W_c h_{t-1} + b_c) \quad (3)$$
$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tilde{c}_t \quad (4)$$
$$o_t = \sigma(V_o x_t + W_o h_{t-1} + b_o) \quad (5)$$
$$h_t = o_t \otimes tanh(c_t) \quad (6)$$

Meanwhile, $*$ can represent f, i, or o to denote the specific gate or c for the memory cell. Therefore, V_* and W_* are the weight matrices, h_* represent the hidden state, b_* is the bias, h_t is the output vector at time instant t. Furthermore, σ and tanh are the sigmoid and *tanh* activation functions [15]. The operator $\otimes$ represents the Hadamard or element-wise product. The first step in the LSTM algorithm is the identification of unrequired information which would be removed from the cell. An LSTM cell serves as a memory to write, read, and delete information depending on the decisions given by the input, output, and forget gates, respectively [25].

III. PROPOSED CREDIT CARD FRAUD DETECTION METHOD

A. SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE AND EDITED NEAREST NEIGHBOR (SMOTE-ENN)

The credit card dataset used in the study is highly imbalanced, leading to poor performance when used to build ML models. The synthetic minority oversampling technique (SMOTE) is widely used in solving the imbalanced class problem [26]–[28]. It is an oversampling technique that balances the class distribution in the dataset by adding synthetic samples to the minority class. In contrast, undersampling methods such as edited nearest neighbor (ENN) creates a balanced dataset by deleting some majority class samples. Meanwhile, undersampling techniques can delete potentially useful examples that might be vital in the learning process. Also, undersampling methods become ineffective when the samples in the majority class significantly outnumber those in the minority class, such as the credit card dataset used in this research. Furthermore, oversampling could lead to overfitting since it makes copies of existing data samples.

Therefore, the proposed credit card fraud detection model employs the synthetic minority oversampling technique and edited nearest neighbor (SMOTE-ENN) method to obtain a balanced dataset. The SMOTE-ENN is a hybrid resampling technique that performs both oversampling and undersampling of the data. It uses SMOTE to oversample the minority class samples and ENN to remove overlapping instances [29]. This algorithm employs the neighborhood cleaning rule from the ENN to remove examples that differ from two in the three nearest neighbors [30]. Algorithm 1 presents the pseudocode for the SMOTE-ENN technique.

Algorithm 1 SMOTE-ENN Technique

Input: Input data

Step 1: Oversampling:

1: Choose a random sample x_i from the minority class
2: Search for the K nearest neighbors of x_i
3: Generate a synthetic sample p by randomly selecting one of the K nearest neighbors q, and connect p and q to create a line segment in the feature space
4: Give the minority class label to the newly created synthetic sample
5: Generate successive synthetic samples as a convex combination of the two selected samples.

Step 2: Undersampling:

6: Select a sample $x_i \in S$, where S denotes the total number of samples x_i from the minority class
7: Search for the K nearest neighbors of x_i
8: If x_i have more neighbors from the other class, then discard x_i.
9: Repeat 6—8 for all the examples in the dataset.

Output: Balanced credit card dataset

B. LSTM ENSEMBLE

This study employs the AdaBoost algorithm to build a robust ensemble model where the base model is an LSTM network. Asuming the credit card dataset contains U training instances, $U = \{(x_1, y_1), \ldots, (x_n, y_i)\}$, where x_* is the independent variable and y_* is the dependent variable (i.e., fraud or legitimate transaction). Let D_m represents the weight distribution of the training samples at the mth boosting iteration, which was assigned a similar value $1/n$ at the first iteration, then the total classification error of the current base model can be computed using:

$$\varepsilon_m = \sum_{i=1}^{n} D_m(i), \quad L_m(x_i) \neq y_i \quad (7)$$

where x_i denotes the input sample and y_i is the corresponding label, L_m represents the trained LSTM model at the mth iteration. Furthermore, the weight distribution of the input data

is updated depending on the prediction performance of the previous classifier in order to assign higher weights to the incorrectly classified instances and lesser weights to the correctly predicted cases. The weight update is achieved using:

$$D_{m+1}(i) = \frac{D_m(i)}{Z_m} e^{-\partial_m y_i L_m(x_i)} \tag{8}$$

where Z_m denotes a normalization parameter and ∂_m represents the voting weight of the base learner L_m. The normalization parameter ensures the weight $D_{m+1}(i)$ have a suitable distribution. Meanwhile, Z_m and ∂_m can be mathematically represented as:

$$Z_m = \sum_1^n D_m(i) \times e^{-\partial_m y_i L_m(x_i)} \tag{9}$$

$$\partial_m = \frac{1}{2} \ln \left(\frac{1 - \varepsilon_m}{\varepsilon_m} \right) \tag{10}$$

After M iterations, the ensemble classifier consists of M base learners. Therefore, the final AdaBoost prediction is the combined predictions weighted by ∂_m:

$$F(x) = sgn(\sum_1^M \partial_m \times L_m(x)) \tag{11}$$

where the sign function $sgn(x)$ is computed using:

$$sgn(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{if } x < 0 \end{cases} \tag{12}$$

The proposed method is represented algorithmically in Algorithm 2. The LSTM models are trained using the resampled data from Algorithm 1 and integrated with the AdaBoost technique to create a powerful ensemble. Lastly, the classification results from the LSTM networks are combined via the weighted voting technique to obtain the final prediction results.

Algorithm 2 LSTM Based Ensemble for Credit Card Fraud Detection

Input: training data, $U=\{(x_1, y_1), \ldots, (x_n, y_i)\}$
LSTM network as base learner L
the number of time steps T and learning iterations M
Output: Ensemble prediction
Procedure:
Step 1: Learn base classifier
initialize the input data weight distribution using $D_m(i)=\frac{1}{n}$, *for all* $i = 1, 2, \ldots, n$
for $m=1, 2, \ldots, M$ **do**
  train an LSTM base learner using U
  for $t=1, 2, \ldots, T$ **do**
    compute the output of the LSTM input gate using (1)
    compute the output of the LSTM forget gate using (2)
    update the LSTM memory cell using (3) and (4)
    update the LSTM output gate using (5)
    compute the LSTM output vector using (6)
  end for
  return $L_m = h_T$
  Step 2: Construct the ensemble prediction
  compute the training error of L_m using (7)
  set the voting weight of L_m using (10)
  update the weights of the training samples using (8)
end for
obtain the final ensemble prediction using (11)

IV. RESULTS AND DISCUSSION

This section presents the experimental results. The proposed LSTM ensemble is benchmarked against some classifiers, including the SVM, MLP, decision tree, LSTM, and the traditional AdaBoost. We performed experiments using the original and resampled datasets to demonstrate the impact of the SMOTE-ENN resampling technique on the performance of the various classifiers. Meanwhile, we used the Python programming language and its associated machine learning libraries for all the experiments. Furthermore, we utilized the stratified 10-fold cross-validation technique to evaluate the performance of the models. The stratified 10-fold cross-validation technique is well suited for imbalanced classification problems. It ensures that the proportion of fraud and non-fraud samples found in the dataset is preserved in all the folds.

The performance of the models is evaluated using the following performance evaluation metrics: sensitivity, specificity, and area under the receiver operating characteristic curve (AUC). Sensitivity, also called recall, indicates the proportion of fraud samples correctly predicted by the classifier [31]. In contrast, specificity (true negative rate) is the proportion of legitimate transactions predicted correctly by the classifier. Meanwhile, the AUC is a measure of the classifier's ability to distinguish between legitimate and fraudulent transactions. An AUC value of 1 implies a perfect model, and the closer the AUC value is to 1, the better the classifier [32]. The sensitivity and specificity can be represented mathematically as:

$$Sensitivity = \frac{TP}{TP + FN} \tag{13}$$

$$Specificity = \frac{TN}{TN + FP} \tag{14}$$

where

- True positive (TP) represents an instance where a transaction is fraudulent, and the classifiers correctly classify it as fraudulent.
- True negative (TN) denotes an instance where a transaction is legitimate, and the classifiers correctly predict it as legitimate.
- False-positive (FP) represents a case where a transaction is legitimate, and the classifier classifies it as fraudulent.
- False-negative (FN) is an instance where a fraudulent transaction is wrongly classified as legitimate.

A. CLASSIFIERS PERFORMANCE WITHOUT DATA RESAMPLING

Firstly, we trained the proposed LSTM ensemble and the benchmark classifiers using the original data, which has not been resampled; this is necessary to demonstrate the impact of the data resampling on the performance of the classifiers. The results obtained are shown in Table 1. The results show that the proposed method achieved superior performance than the other algorithms, having obtained a sensitivity of 0.839, a specificity of 0.982, and an AUC of 0.890. It is observed that all the classifiers obtained poor sensitivity values. The sensitivity or true-positive rate measures the proportion of actual fraud transactions that are correctly identified. The poor sensitivity observed in the classifiers, including the proposed ensemble, can be attributed to the class imbalance inherent in the data, hence, the need for efficient resampling.

TABLE 1. Experimental results without SMOTE-ENN data resampling.

Algorithm	Sensitivity	Specificity	AUC
SVM	0.583	0.954	0.640
MLP	0.755	0.961	0.810
Decision tree	0.588	0.943	0.690
AdaBoost	0.746	0.975	0.830
LSTM	0.761	0.969	0.780
Proposed LSTM Ensemble	0.839	0.982	0.890

B. CLASSIFIERS PERFORMANCE AFTER DATA RESAMPLING

In the second set of experiments conducted in this research, we used the balanced data to train the proposed LSTM ensemble and the other classifiers. The results obtained are shown in Table 2. From the experimental results, the proposed method obtained a sensitivity of 0.996, specificity of 0.998, and AUC of 0.990. Secondly, the classification performance of the various classifiers has been improved compared to Table 1, which can be attributed to the data resampling. Particularly, from Table 2, we can see that the sensitivity values of the classifiers are higher than those in Table 1. Sensitivity is a crucial metric in fraud detection, and the enhanced sensitivity values are significant because it is vital that our models correctly detect fraudulent transactions.

TABLE 2. Experimental results without SMOTE-ENN data resampling.

Algorithm	Sensitivity	Specificity	AUC
SVM	0.912	0.970	0.940
MLP	0.938	0.982	0.930
Decision tree	0.907	0.951	0.920
AdaBoost	0.968	0.994	0.970
LSTM	0.962	0.978	0.950
Proposed LSTM Ensemble	0.996	0.998	0.990

Meanwhile, Fig 1 shows the various classifiers' receiver operating characteristic (ROC) curves. The ROC curve is used to visualize the trade-off between the true-positive rate and false-positive rate, and it is a measure of the prediction ability of the classifier [33]. From Fig 1, the ROC curve of the proposed LSTM ensemble is closer to the upper-left corner, which implies it has a better predictive ability than the other classifiers. Also, the proposed method obtained an AUC value of 0.99, which is superior to the other classifiers. These results imply that the proposed technique achieved high performance in detecting fraudulent and legitimate transactions. Furthermore, Fig 2 and Fig 3 compare the sensitivity and specificity values obtained before and after the SMOTE-ENN data resampling. The figures show that the data resampling significantly enhanced the performance of the various classifiers, including the proposed ensemble.

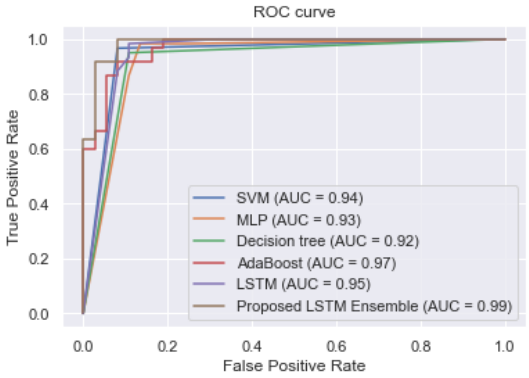

FIGURE 1. ROC curve of the various models.

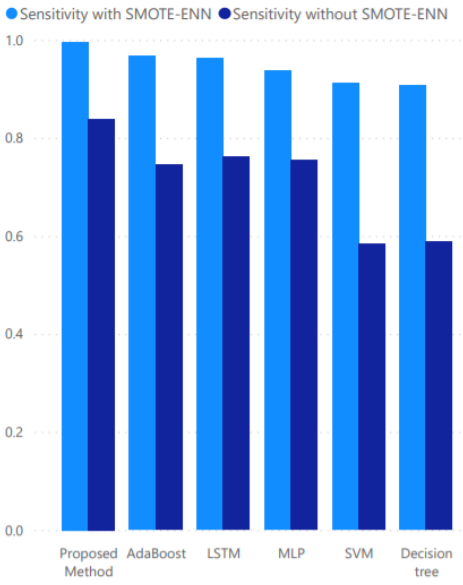

FIGURE 2. Sensitivity comparison.

C. COMPARISON WITH EXISTING METHODS

It is not sufficient to base the superior performance of our proposed method on the comparison with conventional algorithms. However, it is necessary to compare our approach with existing credit card fraud detection methods in the

TABLE 3. Comparison with some existing methods.

Reference	Method	Sensitivity	Specificity	AUC
Kalid et al. [5]	C4.5+NB	0.872	1.000	-
Taha et al. [14]	LightGBM	-	-	0.928
Makki et al. [6]	CS SVM	0.650	-	0.620
Khatri et al. [34]	Optimized Random forest	0.782	-	-
Alkhatib et al. [35]	DNN	0.955	-	0.990
Mrozek et al. [36]	Random forest + SMOTE	0.829	-	0.910
Zhou et al. [37]	AdaBoost + SMOTE + PCA	-	-	0.965
Yotsawat et al. [38]	CS-NNE	-	0.936	0.980
Carta et al. [39]	Stochastic Ensemble Classifier	0.915	-	0.876
Xia et al. [40]	OCHE	-	-	0.937
Feng et al. [41]	DWE-MC	-	-	0.66
Xie et al. [42]	XGBoost + SMOTE	0.988	-	0.970
This paper	Proposed LSTM Ensemble with SMOTE-ENN	0.996	0.998	0.990

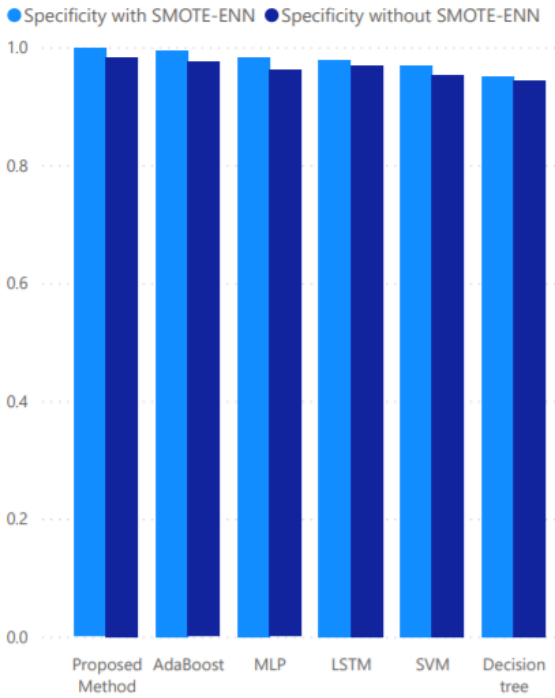

FIGURE 3. Specificity comparison.

literature. The methods include the following: the sequential combination of C4.5 decision tree and naïve Bayes (NB) [5], a light gradient boosting machine (LightGBM) with a Bayesian-based hyperparameter optimization algorithm [14], a light gradient boosting machine (LightGBM) with a Bayesian-based hyperparameter optimization algorithm [14], a cost-sensitive SVM (CS SVM) [6], an optimized random forest (RF) classifier [34], a deep neural network (DNN) [35], a random forest classifier with SMOTE data resampling [36], an improved AdaBoost classifier with principal component analysis (PCA) and SMOTE method [37], a cost-sensitive neural network ensemble (CS-NNE) [38], a stochastic ensemble classifier operating in a discretized feature space [39], a model based on overfitting-cautious heterogeneous ensemble (OCHE) [40], a dynamic weighted ensemble technique using Markov Chain (DWE-MC) [41], and an extreme gradient boosting (XGBoost) ensemble classifier with SMOTE resampling technique [42].

In Table 3, the proposed LSTM ensemble with SMOTE-ENN showed excellent performance compared to the other state-of-the-art methods, indicating the robustness of the proposed approach. Lastly, to further validate the effectiveness of the proposed approach, we carried out more simulations using two more real-world datasets, i.e. the Taiwan default of credit card clients dataset [43] and the German credit dataset [44]. Both datasets have an imbalanced class distribution. The Taiwan dataset contains 30 000 instances, where 6 636 and 23 364 cases are categorized as bad and good clients, respectively. Meanwhile, the German dataset comprises 1 000 instances, where the bad clients are 300, and good clients are 700. The experimental results are tabulated in Tables 4-7.

TABLE 4. Experimental results using the Taiwan dataset without SMOTE-ENN data resampling.

Algorithm	Sensitivity	Specificity	AUC
SVM	0.620	0.824	0.650
MLP	0.607	0.790	0.680
Decision tree	0.571	0.772	0.610
AdaBoost	0.666	0.884	0.670
LSTM	0.629	0.859	0.640
Proposed LSTM Ensemble	0.725	0.890	0.700

TABLE 5. Experimental results using the Taiwan dataset with SMOTE-ENN data resampling.

Algorithm	Sensitivity	Specificity	AUC
SVM	0.796	0.887	0.790
MLP	0.820	0.915	0.840
Decision tree	0.709	0.866	0.770
AdaBoost	0.865	0.927	0.890
LSTM	0.838	0.898	0.860
Proposed LSTM Ensemble	0.924	0.951	0.930

From Tables 4-7, the proposed LSTM ensemble obtained the best performance compared to the other classifiers. For the Taiwan credit card dataset, the proposed LSTM ensemble

TABLE 6. Experimental results using the German dataset without SMOTE-ENN data resampling.

Algorithm	Sensitivity	Specificity	AUC
SVM	0.631	0.810	0.630
MLP	0.677	0.841	0.680
Decision tree	0.625	0.790	0.650
AdaBoost	0.708	0.880	0.740
LSTM	0.674	0.849	0.700
Proposed LSTM Ensemble	0.751	0.911	0.810

TABLE 7. Experimental results using the German dataset with SMOTE-ENN data resampling.

Algorithm	Sensitivity	Specificity	AUC
SVM	0.783	0.892	0.810
MLP	0.806	0.915	0.810
Decision tree	0.716	0.870	0.790
AdaBoost	0.822	0.895	0.830
LSTM	0.820	0.916	0.870
Proposed LSTM Ensemble	0.904	0.933	0.910

obtained a sensitivity of 0.924, a specificity of 0.951, and an AUC of 0.930. For the German credit dataset, the proposed classifier achieved a sensitivity of 0.904, a specificity of 0.933, and an AUC of 0.910. Therefore, from the above experimental results, it is fair to conclude that the combination of SMOTE-ENN and the proposed LSTM ensemble is an efficient method to detect credit card fraud.

V. CONCLUSION

Recently, machine learning has been crucial in detecting credit card fraud, though the class imbalance has been a significant challenge. This paper proposed an efficient approach for credit card fraud detection. Firstly, the SMOTE-ENN technique was employed to create a balanced dataset. Secondly, a robust deep learning ensemble was developed using the LSTM neural network as the base learner in the AdaBoost technique. From the experimental results, using the well-known credit card fraud detection dataset, the proposed LSTM ensemble with SMOTE-ENN data resampling achieved a sensitivity of 0.996, a specificity of 0.998, and an AUC of 0.990, which is superior to the other benchmark algorithms and state-of-the-art methods. Therefore, combining the SMOTE-ENN data resampling technique and the boosted LSTM classifier is an efficient method in detecting fraud in credit card transactions. Future research would consider more resampling techniques and improved feature selection techniques for enhanced classification performance.

REFERENCES

[1] 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.
[2] H. Tingfei, C. Guangquan, and H. Kuihua, "Using variational auto encoding in credit card fraud detection," *IEEE Access*, vol. 8, pp. 149841–149853, 2020.
[3] M. C. M. Oo and T. Thein, "An efficient predictive analytics system for high dimensional big data," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 1, pp. 1521–1532, Jan. 2022.
[4] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, "Credit card fraud detection: A realistic modeling and a novel learning strategy," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 8, pp. 3784–3797, Aug. 2017.
[5] S. N. Kalid, K.-H. Ng, G.-K. Tong, and K.-C. Khor, "A multiple classifiers system for anomaly detection in credit card data with unbalanced and overlapped classes," *IEEE Access*, vol. 8, pp. 28210–28221, 2020.
[6] S. Makki, Z. Assaghir, Y. Taher, R. Haque, M.-S. Hacid, and H. Zeineddine, "An experimental study with imbalanced classification approaches for credit card fraud detection," *IEEE Access*, vol. 7, pp. 93010–93022, 2019.
[7] S. A. Ebiaredoh-Mienye, E. Esenogho, and T. G. Swart, "Artificial neural network technique for improving prediction of credit card default: A stacked sparse autoencoder approach," *Int. J. Electr. Comput. Eng.*, vol. 11, no. 5, p. 4392, Oct. 2021.
[8] 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.
[9] H. Patel, D. S. Rajput, G. T. Reddy, C. Iwendi, A. K. Bashir, and O. Jo, "A review on classification of imbalanced data for wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 16, no. 4, Apr. 2020, Art. no. 1550147720916404.
[10] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *J. Big Data*, vol. 6, no. 1, pp. 1–54, Dec. 2019.
[11] 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.
[12] I. D. Mienye and Y. Sun, "Performance analysis of cost-sensitive learning methods with application to imbalanced medical data," *Informat. Med. Unlocked*, vol. 25, 2021, Art. no. 100690.
[13] T. M. Padmaja, N. Dhulipalla, R. S. Bapi, and P. R. Krishna, "Unbalanced data classification using extreme outlier elimination and sampling techniques for fraud detection," in *Proc. 15th Int. Conf. Adv. Comput. Commun. (ADCOM)*, Dec. 2007, pp. 511–516.
[14] A. A. Taha and S. J. Malebary, "An intelligent approach to credit card fraud detection using an optimized light gradient boosting machine," *IEEE Access*, vol. 8, pp. 25579–25587, 2020.
[15] J. Yang, J. Qu, Q. Mi, and Q. Li, "A CNN-LSTM model for tailings dam risk prediction," *IEEE Access*, vol. 8, pp. 206491–206502, 2020.
[16] F. Shen, X. Zhao, G. Kou, and F. E. Alsaadi, "A new deep learning ensemble credit risk evaluation model with an improved synthetic minority oversampling technique," *Appl. Soft Comput.*, vol. 98, Jan. 2021, Art. no. 106852.
[17] G. Maragatham and S. Devi, "LSTM model for prediction of heart failure in big data," *J. Med. Syst.*, vol. 43, no. 5, pp. 1–13, May 2019.
[18] B. Wiese and C. Omlin, "Credit card transactions, fraud detection, and machine learning: Modelling time with LSTM recurrent neural networks," in *Innovations in Neural Information Paradigms and Applications*. Springer, 2009, pp. 231–268.
[19] S. Subudhi and S. Panigrahi, "Application of OPTICS and ensemble learning for database intrusion detection," *J. King Saud Univ. Comput. Inf. Sci.*, May 2019.
[20] *Credit Card Fraud Detection*. Accessed: Oct. 2021, 26. [Online]. Available: https://kaggle.com/mlg-ulb/creditcardfraud
[21] R. E. Schapire, "A brief introduction to boosting," in *Proc. IJCAI*, vol. 99, 1999, pp. 1401–1406.
[22] F. Wang, Z. Li, F. He, R. Wang, W. Yu, and F. Nie, "Feature learning viewpoint of AdaBoost and a new algorithm," *IEEE Access*, vol. 7, pp. 149890–149899, 2019.
[23] M. Mazini, B. Shirazi, and I. Mahdavi, "Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 31, no. 4, pp. 541–553, Oct. 2019.
[24] W. Wang, M. Tong, and M. Yu, "Blood glucose prediction with VMD and LSTM optimized by improved particle swarm optimization," *IEEE Access*, vol. 8, pp. 217908–217916, 2020.
[25] S. R. Venna, A. Tavanaei, R. N. Gottumukkala, V. V. Raghavan, A. S. Maida, and S. Nichols, "A novel data-driven model for real-time influenza forecasting," *IEEE Access*, vol. 7, pp. 7691–7701, 2019.
[26] S. F. Abdoh, M. A. Rizka, and F. A. Maghraby, "Cervical cancer diagnosis using random forest classifier with SMOTE and feature reduction techniques," *IEEE Access*, vol. 6, pp. 59475–59485, 2018.

[27] A. Ishaq, S. Sadiq, M. Umer, S. Ullah, S. Mirjalili, V. Rupapara, and M. Nappi, "Improving the prediction of heart failure Patients' survival using SMOTE and effective data mining techniques," *IEEE Access*, vol. 9, pp. 39707–39716, 2021.
[28] Asniar, N. U. Maulidevi, and K. Surendro, "SMOTE-LOF for noise identification in imbalanced data classification," *J. King Saud Univ. Comput. Inf. Sci.*, Feb. 2021.
[29] M. S. K. Inan, R. E. Ulfath, F. I. Alam, F. K. Bappee, and R. Hasan, "Improved sampling and feature selection to support extreme gradient boosting for PCOS diagnosis," in *Proc. IEEE 11th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Jan. 2021, pp. 1046–1050.
[30] T. Le, M. T. Vo, B. Vo, M. Y. Lee, and S. W. Baik, "A hybrid approach using oversampling technique and cost-sensitive learning for bankruptcy prediction," *Complexity*, vol. 2019, pp. 1–12, Aug. 2019.
[31] S. Subudhi and S. Panigrahi, "Use of optimized fuzzy C-Means clustering and supervised classifiers for automobile insurance fraud detection," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 32, no. 5, pp. 568–575, Jun. 2020.
[32] S. A. Ebiaredoh-Mienye, E. Esenogho, and T. G. Swart, "Integrating enhanced sparse autoencoder-based artificial neural network technique and softmax regression for medical diagnosis," *Electronics*, vol. 9, no. 11, p. 1963, Nov. 2020.
[33] I. D. Mienye and Y. Sun, "Improved heart disease prediction using particle swarm optimization based stacked sparse autoencoder," *Electronics*, vol. 10, no. 19, p. 2347, Sep. 2021.
[34] S. Khatri, A. Arora, and A. P. Agrawal, "Supervised machine learning algorithms for credit card fraud detection: A comparison," in *Proc. 10th Int. Conf. Cloud Comput., Data Sci. Eng. (Confluence)*, Jan. 2020, pp. 680–683.
[35] K. I. Alkhatib, A. I. Al-Aiad, M. H. Almahmoud, and O. N. Elayan, "Credit card fraud detection based on deep neural network approach," in *Proc. 12th Int. Conf. Inf. Commun. Syst. (ICICS)*, May 2021, pp. 153–156.
[36] P. Mrozek, J. Panneerselvam, and O. Bagdasar, "Efficient resampling for fraud detection during anonymised credit card transactions with unbalanced datasets," in *Proc. IEEE/ACM 13th Int. Conf. Utility Cloud Comput. (UCC)*, Dec. 2020, pp. 426–433.
[37] H. Zhou, L. Wei, G. Chen, P. Lin, and Y. Lin, "Credit card fraud identification based on principal component analysis and improved AdaBoost algorithm," in *Proc. Int. Conf. Intell. Comput., Autom. Syst. (ICICAS)*, Dec. 2019, pp. 507–510.
[38] W. Yotsawat, P. Wattuya, and A. Srivihok, "A novel method for credit scoring based on cost-sensitive neural network ensemble," *IEEE Access*, vol. 9, pp. 78521–78537, 2021.
[39] S. Carta, A. Ferreira, D. R. Recupero, and R. Saia, "Credit scoring by leveraging an ensemble stochastic criterion in a transformed feature space," *Prog. Artif. Intell.*, vol. 10, pp. 1–16, May 2021.
[40] Y. Xia, J. Zhao, L. He, Y. Li, and M. Niu, "A novel tree-based dynamic heterogeneous ensemble method for credit scoring," *Expert Syst. Appl.*, vol. 159, Nov. 2020, Art. no. 113615.
[41] X. Feng, Z. Xiao, B. Zhong, Y. Dong, and J. Qiu, "Dynamic weighted ensemble classification for credit scoring using Markov chain," *Int. J. Speech Technol.*, vol. 49, no. 2, pp. 555–568, Feb. 2019.
[42] Y. Xie, A. Li, L. Gao, and Z. Liu, "A heterogeneous ensemble learning model based on data distribution for credit card fraud detection," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–13, Jul. 2021.
[43] *UCI machine Learning Repository: Default of Credit Card Clients Data Set*. Accessed: Mar. 14, 2020. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients
[44] *UCI Machine Learning Repository: Statlog (German Credit Data) Data Set*. Accessed: Oct. 31, 2020. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)

EBENEZER ESENOGHO (Member, IEEE) received the B.Eng. degree in computer engineering, in 2008, the M.Eng. degree in electronics/telecommunication engineering from the University of Benin, in 2012, and the Ph.D. degree in electronic (5G cognitive networks) from the University of KwaZulu-Natal, in 2017. He previously lectured with the University of Benin, from March 2011 to March 2013, and rose to the rank of Lecturer I. Previously, he was involved in research and teaching with the Centre for Radio Access and Rural Technology, Centre of Excellence, University of KwaZulu-Natal, funded by Alcatel, Ericsson, and Huawei. He has authored/coauthored several peer-reviewed journals and conference papers, chaired sessions in conferences and reviewed some notable ISI/Scopus journals. His research interests include the fifth generation (5G) wireless networks, cognitive radio networks, wireless sensor networks, artificial intelligence/machine learning, big data & mobile/cloud computing, and visible light communication. He is a Registered Engineer in Nigeria and a member of the South Africa Institute of Electrical Engineers (SAIEE) and IEEE Region 8. He was a recipient of several grants/scholarship/award/fellowships, including the CEPS/Eskom's HVDC 2013, CEPS/Eskom's HVDC 2014, J. W Nelson 2015, and GES 2017, 2018, 2019/2020. He was a UJ/DST/NRF research delegate to the H2020-ESASTAP EU-South Africa STI Cooperation on Strengthening Technology, Research and Innovation in Vienna, Austria.

IBOMOIYE DOMOR MIENYE (Member, IEEE) received the B.Eng. degree in electrical and electronic engineering and the M.Sc. degree *(cum laude)* in computer systems engineering from the University of East London, in 2012 and 2014, respectively, and the Ph.D. degree in electrical and electronic engineering from the University of Johannesburg, South Africa. He is currently a Postdoctoral Research Fellow with the Department of Electrical and Electronic Engineering Science, University of Johannesburg. His research interests include machine learning and deep learning for healthcare applications.

THEO G. SWART (Senior Member, IEEE) received the B.Eng. and M.Eng. degrees *(cum laude)* in electrical and electronic engineering from Rand Afrikaans University, South Africa, in 1999 and 2001, respectively, and the D.Eng. degree from the University of Johannesburg, South Africa, in 2006. He was the Chair of the IEEE South Africa Chapter on Information Theory. He is currently an Associate Professor with the Department of Electrical and Electronic Engineering Science, and also the Director of the Center for Telecommunications, University of Johannesburg. To date, he has more than 20 journal articles and more than 50 conference papers to his name. His research interests include digital communications, error-correction coding, constrained coding, and power-line communications. He has been the co-editor and a contributor of two editions of a comprehensive book on power-line communications. He is also a Specialist Editor in Communications and Signal Processing for the *SAIEE Africa Research Journal*. He has served on several technical program committees for IEEE conferences and regularly serves as a reviewer for IEEE journals and conferences.

KEHINDE ARULEBA received the Ph.D. degree in computer science from the University of the Witwatersrand, Johannesburg, South Africa. He is currently a Lecturer with the University of Leicester, U.K. Before that, he was a Postdoctoral Fellow with Walter Sisulu University, South Africa. His expertise extends beyond computer science. He actively collaborates with researchers across disciplines. His research interests include machine learning, data ethics, and ICT4D.

GEORGE OBAIDO received the M.Sc. and Ph.D. degrees in computer science from the University of the Witwatersrand, Johannesburg, South Africa. He is currently a Postdoctoral Scholar with the Department of Computer Science and Engineering, Jacobs School of Engineering, University of California at San Diego, San Diego, USA. His research interests include the intersection of computational techniques (machine learning and deep learning) to finding solutions to many societal problems.