


RESEARCH

Open Access



Enhanced credit card fraud detection based on attention mechanism and LSTM deep model

Ibtissam Benchaji^{1*} , Samira Douzi^{2*}, Bouabid El Ouahidi^{1*} and Jaafar Jaafari^{3*}

*Correspondence:

ibtissam_benchaji@um5.ac.ma; s.douzi@um5r.ac.ma; b.elouahidi@um5r.ac.ma; jaafari.jaafar@gmail.com

¹ L.R.I, Faculty of Sciences, Mohammed V University, Rabat, Morocco

² FMPR, Mohammed V University, Rabat, Morocco

³ FSTM, Hassan II University, Casablanca, Morocco

Abstract

As credit card becomes the most popular payment mode particularly in the online sector, the fraudulent activities using credit card payment technologies are rapidly increasing as a result. For this end, it is obligatory for financial institutions to continuously improve their fraud detection systems to reduce huge losses. The purpose of this paper is to develop a novel system for credit card fraud detection based on sequential modeling of data, using attention mechanism and LSTM deep recurrent neural networks. The proposed model, compared to previous studies, considers the sequential nature of transactional data and allows the classifier to identify the most important transactions in the input sequence that predict at higher accuracy fraudulent transactions. Precisely, the robustness of our model is built by combining the strength of three sub-methods; the uniform manifold approximation and projection (UMAP) for selecting the most useful predictive features, the Long Short Term Memory (LSTM) networks for incorporating transaction sequences and the attention mechanism to enhance LSTM performances. The experimentations of our model give strong results in terms of efficiency and effectiveness.

Keywords: Deep learning, Attention mechanism, Fraud detection, Sequence learning, Recurrent neural networks, LSTM, Dimensionality reduction

Introduction

Recently, with the improved availability of technology and the emergence of new e-service payment solutions, such as e-commerce and mobile payments, credit card transactions have become omnipresent. Such extensive adoption of cashless transactions lead fraudsters to commit frequent fraud attacks and constantly change their strategies to avoid being detected [1, 2].

In payment industry, credit card fraud detection aims to decide whether a transaction is fraudulent or not based on historical data [3]. The decision is extremely challenging because of the following reasons:

1. Fraudsters continuously invent novel fraud patterns, especially those that they use to adapt to fraud detection techniques.

2. Machine learning models that are never updated are inadequate as they do not take account of changes and trends in customer spending behaviors, for example during holiday periods and geographical regions.

In such situations, financial institutions should establish continuously an increasingly sophisticated fraud detection system (FDS) to mitigate the prevailing menace of fraud and detect it immediately, with an objective to prevent fraud before it occurs, protect consumers' interests and reduce the heavy annual financial losses caused by fraud around the world [4–8].

In this paper, we propose a novel credit card fraud detection system based on Long Short Term Memory (LSTM) networks and attention mechanism. The attention mechanism allows a sequence based neural network to automatically focus on the data items that are the most important to the classification task by a data-driven weighted average of local information contained in each term of the sequence which results in an improved detection performance. The main contributions of our proposed fraud detection method are:

1. Optimizing the process of learning classifiers by using feature selection and dimension reduction algorithms such as PCA, t-SNE and UMAP.
2. Overcoming the issue of the imbalanced dataset and incrementing the learning rate by using the Synthetic Minority Oversampling Technique (SMOTE).
3. Constructing the context of consumer's spending behavior by using the sequence learner LSTM recurrent neural networks, as a dynamic pattern recognition classifier to model long term dependencies within transaction sequences.
4. Applying the attention mechanism upon LSTM recurrent networks, that efficiently allows the classifier to learn where to pay selective attention in the input sequence for the global fraud decision, which deliver good performances.
5. Performing experiments on two different datasets from which we conclude that our method is competitive and alternative to existing LSTM works.

This work opens perspectives for dealing with sequential data in fraud detection area. In order to ensure reproducibility, the source code and results of the proposed model can be found at <https://github.com/bibtissam/LSTM-Attention-FraudDetection>.

The rest of the paper is organized as follows; “[Related works](#)” section presents the related works describing prior works in credit card fraud detection domain, “[Background](#)” section presents the structure of our proposed model, “[Methods and materials](#)” section describes the datasets used in this study and discusses the results obtained. Finally, the paper is concluded in “[Conclusion](#)” section and suggested ideas for future research.

Related works

A wide range of machine learning approaches based on supervised learning, unsupervised learning, anomaly detection and ensemble learning have been used in payment card fraud detection [9]. In particular, supervised classification techniques demonstrated to be extremely effective for facing this challenge, where pre-classified datasets

containing labeled historical transactions are used for training a classifier that builds a detection model capable to predict whether a new transaction is fraudulent or not. Some of these algorithms are support vector machines [10, 11], hidden Markov models [11, 12], logistic regression algorithms [10, 13], decision trees [14, 15], random forests [10, 16–19], and k-nearest neighbors [20, 21].

Unsupervised classification methods are used to detect unusual behavior of a system and to identify transactions that do not conform to the model as potential fraudulent cases [22–24]. It can help to detect some new patterns of fraud that have not been detected before.

However, credit card fraud detection presents several challenges that attract the attention of artificial intelligence communities for several reasons. One of them is the fact that credit card fraud data sets are highly imbalanced since the number of genuine transactions is much higher than the fraudulent ones. Thus, many of traditional classifiers fail to detect minority class objects for these skewed data sets [25, 26]. On the other hand, these traditional classifiers aim to identify transactions with a high probability of being fraud, based only on individual transaction information such as amount, time and transaction location [27–29] but ignore detailed sequential information that defines consumers' profile. Such models are inadequate for credit card fraud detection, since they do not consider the consumer spending behavior, which is useful to discover relevant fraud patterns that evolves over time due to seasonality and new attack strategies [30, 31].

Recently, deep learning methods based on recurrent neural networks (RNN) and specially its variant Long Short Term Memory Networks (LSTM), have been used in fraud detection field given their reputation as one of the most accurate learning algorithms in sequence analysis work [32–36]. RNN is a dynamic machine learning approach capable of analyzing the dynamic temporal behaviors of various bank accounts by modeling the sequential dependency between consecutive transactions of credit card holders.

The attention mechanism has also been recently proposed [37] as a way to find context-dependent representations. This method takes into account dependencies between items in a sequence despite of their distance. It has been used to define context in machine translation [37] and image captioning [38] with great success. The idea behind the attention mechanism is to take a weighted average of a set of vectors to construct a context vector that contains the most relevant information, which is then used as input in the next layer.

In this paper, we use LSTM based sequence models and attention models to discover temporal correlations between events that are possibly far away from each other in the input sequence which improve the effectiveness of the classification task and allow for an increase in the detection of fraudulent transactions when compared to traditional models.

Background

In this section, we will introduce the related literature that formed the basis of our work.

Dimensionality reduction algorithms

Feature selection and feature extraction are fundamental preprocessing steps in fraud detection systems [16, 39], to select the optimal subset of relevant features by removing

redundant, noisy and irrelevant features from the original dataset, and decrease the computational cost without a negative effect on the classification accuracy.

Feature selection

The basis of credit card fraud detection lies in the analysis of cardholder's spending behavior. This spending profile is analyzed using optimal selection of variables that capture the unique behavior of a credit card and detect very dissimilar transactions within the purchases of a customer. Also, since the profile of both a legitimate and fraudulent transaction tends to be constantly changing, optimal selection of variables that greatly differentiates both profiles is needed to achieve efficient classification of credit card transaction [40–42]. Therefore, Swarm intelligence based feature selection approach [43] will be used to explore our datasets and study the influence of each feature in the prediction of the target class.

Feature extraction

In this research, we employ three algorithms to reduce the dataset dimensionality, namely Principal Component Analysis (PCA) [44], t-distributed stochastic neighbor embedding (t-SNE) [45, 46], and uniform manifold approximation and projection (UMAP) [47, 49]. These aforementioned algorithms have been seen as one of the best dimension reduction algorithms used for feature extraction in many applications such as bioinformatics, and visualization [47].

- **Principal Component Analysis (PCA)**

PCA is a popular method used in dimension reduction, aiming to transform the original set of n correlated variables to a new subset of m uncorrelated variables called principal components (PCs) that successively maximize variance. These new variables are linear combinations of the original variables and are derived in decreasing order of importance, such that the first principal component accounts for as much as possible of the variation in the original data.

Given a set of n correlated variables f_1, f_2, \dots, f_n , the objective of PCA consists in replacing these n measured variables by m derived variables z_1, z_2, \dots, z_m , which are uncorrelated and whose variances decrease from first to last, without minimizing any information loss. This transformation is done with respect to the following properties:

- z_1 has maximum possible variance among all possible linear functions of f_1, f_2, \dots, f_n . The correspondent equation is given by:

$$z_1 = \alpha_1 f_1 + \alpha_2 f_2 + \dots + \alpha_n f_n. \quad (1)$$

- z_2 has maximum possible variance among all possible linear functions of f_1, f_2, \dots, f_n , subject to z_2 being uncorrelated with z_1 .

- c. In general, z_k has maximum possible variance among all possible linear functions of f_1, f_2, \dots, f_n , subject to z_k being uncorrelated with z_1, z_2, \dots, z_{k-1} , for $2 \leq k \leq n$.

Although PCA is able to cover the maximum variance among features, but as a linear algorithm, it may performs poorly on the features with nonlinear relationship. Therefore, in order to present high dimensional data on low dimensional and nonlinear basis, some nonlinear dimensional reduction algorithms such as t-SNE and UMAP are employed.

- **t-distributed Stochastic Neighbor Embedding (t-SNE)**

The t-distributed stochastic neighbor embedding (t-SNE) is a machine learning algorithm that is well suited for reducing high nonlinear dimensional data into the two or three dimensional space. It tries to place a point from high dimensional space in a low dimensional one so as to preserve neighborhood identity; closer data points mean high similarity.

There are two main stages in t-SNE. First, it finds a probability distribution over pairs of data such that a pair of similar data points is given a high probability, while a pair of farther away points is given a low probability. Second, it defines a probability distribution in the lower dimension space that is similar to that in the original high dimensional space, and aims to minimize the Kullback–Leibler (KL) divergence between the two distributions [45].

Given a high dimensional input dataset x_1, x_2, \dots, x_n in R^m , our goal is to find an optimal low dimensional representation y_1, y_2, \dots, y_n in R^k , such that $k \leq m$. The similarity of data point x_j to data point x_i is represented by the conditional probability p_{ji} . For the low dimensional counterparts y_i and y_j for the high dimensional data points x_i and x_j , it is computed a similar conditional probability denoted by q_{ji} .

Once p_{ji} and q_{ji} are calculated, the goal of the t-SNE algorithm is to minimize the mismatch between the high and low dimensional representations. The cost function (Eq. 2) that minimizes the Kullback–Leibler (KL) divergences over all points is given as:

$$KL(P|Q) = \sum_i \sum_j p_{ji} \log \frac{p_{ji}}{q_{ji}}, \tag{2}$$

where P and Q represent respectively the probability distributions for p_{ji} and q_{ji} .

Although the t-SNE algorithm is a good technique to visualize data in a low dimensional space, it computes pairwise conditional probabilities for each pair of samples and involves hyperparameters that are not always simple to tune, which comes with a high computational cost.

- **Uniform Manifold Approximation and Projection (UMAP)**

Uniform Manifold Approximation and Projection (UMAP) is an emerging dimensionality reduction technique that has been recently published by McInnes and Healy [49]. It is based on the theory of Riemannian geometry and algebraic topology that uses local manifold approximations and patches together their local fuzzy simplicial set representations to construct a topological representation of the high

dimensional data, then a similar process can be used to search for a low dimensional projection of the data that has the closest possible equivalent fuzzy topological structure of the original space.

Unlike t-SNE which utilizes probabilistic model, UMAP is a graph-based algorithm. The first phase of UMAP is to construct a weighted k-neighbour graph representation of each of the original high-dimensional data point such that the edge-wise cross-entropy between the weighted graph and the original data is minimized. Then, the k-dimensional eigenvectors of the UMAP graph are used to represent each of the original data point.

UMAP considers the input data $X = \{x_1, x_2, \dots, x_n\}$ in R^m , with a metric (or dissimilarity measure) $d: X \times X \rightarrow R^+$ and look for an optimal low dimensional representation $\{y_1, y_2, \dots, y_n\}$ in R^k , such that $k < m$. Given an input hyperparameter k , for each x_i we compute the set $\{x_{i1}, x_{i2}, \dots, x_{ik}\}$ of the k nearest neighbors of x_i under the metric d . For each x_i , we will define ρ_i and σ_i . Let:

$$\rho_i = \min\{d(x_i, x_{ij}) | 1 \leq j \leq k, d(x_i, x_{ij}) > 0\}, \tag{3}$$

where σ_i is defined such that:

$$\sum_{j=1}^k \exp\left(\frac{-\max(0, d(x_i, x_{ij}) - \rho_i)}{\sigma_i}\right) = \log_2(k). \tag{4}$$

One chooses ρ_i to ensure at least one data point is connected to x_i with an edge weight of 1 which is equivalent to the resulting fuzzy simplicial set being locally connected at x_i .

The σ_i is set as a length scale parameter, defining a weighted directed graph $\bar{G} = (V, E, \omega)$, where V is the set of vertices (in this case, the data X), E is the set of directed edges $E = \{(x_i, x_{ij}) | 1 \leq j \leq k, 1 \leq i \leq n\}$, and ω is the weight function for edges defined by setting:

$$\omega(x_i, x_{ij}) = \exp\left(\frac{-\max(0, d(x_i, x_{ij}) - \rho_i)}{\sigma_i}\right). \tag{5}$$

UMAP tries to define an undirected weighted graph G from directed graph \bar{G} via symmetrization. Let A be the adjacency matrix of the graph \bar{G} . A symmetric matrix can be obtained by:

$$B = A + A^T - A \otimes A^T, \tag{6}$$

where T is the transpose and \otimes denotes the Hadamard (or pointwise) product. Then, the undirected weighted Laplacian G (the UMAP graph) is defined by its adjacency matrix B . The goal is to find the optimal low-dimensional coordinates $\{y_i\}_{i=1}^n, y_i \in R^k$, that minimizes the edgewise cross entropy with the original data at each point. The evolution of the UMAP graph Laplacian G can be regarded as a discrete approximation of the Laplace–Beltrami operator on a manifold defined by the data [48]. Implementation and further detail of UMAP can be found in [49].

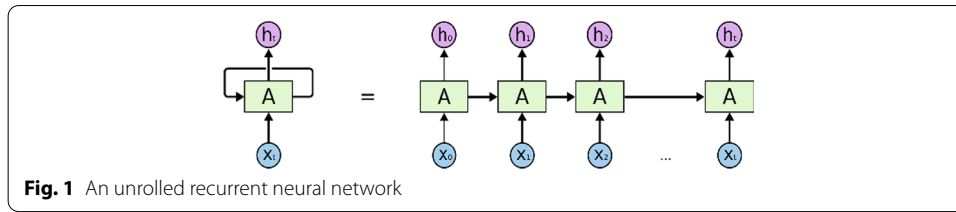


Fig. 1 An unrolled recurrent neural network

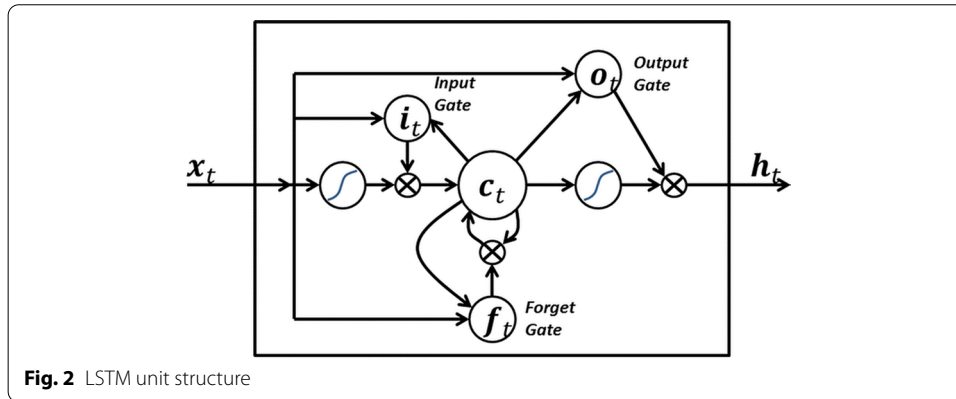


Fig. 2 LSTM unit structure

Compared to t-SNE, UMAP is better able to preserve as much of the local and more of the global data structure, with superior runtime performance [49].

Long Short Term Memory networks

Long Short-Term Memory (LSTM) is a special type of artificial recurrent neural network (RNN) architecture used to model time series information in the field of deep learning (Fig. 1). In contrast to standard feedforward neural networks, LSTM has feedback connections between hidden units that are associated with discrete time steps, which allow long term sequence dependencies to be learned and a transaction label to be predicted given the sequence of past transactions. LSTMs were developed to overcome the problem of vanishing and exploding gradient that can be observed during the training of traditional RNNs [50].

LSTM unit consists of a memory cell that stores information which is updated by three special gates: the input gate, the forget gate and the output gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell. Figure 2 depicts the LSTM unit structure.

At time t , x_t is the input data of the LSTM cell, h_{t-1} is the output of the LSTM cell at the previous moment, c_t is the value of the memory cell, h_t is the output of the LSTM cell.

The LSTM unit calculation method can be divided into the steps below:

1. The first step according to Eq. (3) is to calculate the candidate's memory cell value \tilde{c}_t , W_c is the weight matrix, b_c is the bias.

$$\tilde{c} = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c). \tag{7}$$

- Calculate the value of the input gate i_t , the input gate controls the update of the current input data to the state value of the memory cell, σ is sigmoid function, W_i is the weight matrix, b_i is the bias. The equation for the input gate is given by:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i). \tag{8}$$

- Calculate the value of the forget gate f_t , the forget gate controls the update of the historical data to the state value of the memory cell, W_f is the weight matrix, b_f is the bias. The equation for the forget gate is given by:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f). \tag{9}$$

- Calculate the value of the current moment memory cell c_t , and c_{t-1} is the state value of the last LSTM unit. We use the following equation:

$$c_t = f_t * c_f + i_t * \tilde{c}_t. \tag{10}$$

Here dot product is represented by “*”.

The memory cell update depends on the state value of the last cell and the candidate cell and is controlled by the input gate and the forget gate.

- Calculate the value of the output gate o_t , the output gate controls the output of the memory cell’s state value, W_o is the weight matrix, b_o is the bias. The equation for the output gate is given by:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o). \tag{11}$$

- Finally, calculate the output of the LSTM unit according to the equation:

$$h_t = o_t * \tanh(c_t). \tag{12}$$

Benefit from three control gates and memory cell, LSTM can easily retain, read, reset and update information over long periods of time.

Attention mechanism

In modern deep learning research such as computer vision and language translation [51, 52], attention mechanisms have become an effective way to achieve excellent results by selecting important information. This mechanism aims to focus only on the most relevant piece of information, rather than all of it, which is sufficient for further neural processing [53].

To illustrate the attention mechanism, let consider an RNN Encoder–Decoder architecture: an encoder reads the input sequence of vectors $x=(x_1,x_2,\dots,x_n)$ into a vector c . This approach is often explained in an RNN structure in the following form:

$$S_t = f(x_t, S_{t-1}, c_t), \tag{13}$$

And:

$$c = q(S_1, \dots, S_n), \tag{14}$$

where S_t is the hidden state, c is the output vector of the RNN which is generated by the hidden states. In attention model, the context vector c_t is strongly related to the

sequence of annotations (h_1, \dots, h_n) to which an encoder maps the input sequence. The information about the whole input sequence with a strong focus on the parts surrounding the t -th word of the input sequence is contained in the annotation h_t . Details can be found in the following explanations. Figure 3 shows the attention mechanism in neural network. A weighted sum of those annotations h_t forms the context vector c_t :

$$c_t = \sum_{j=1}^n \alpha_{tj} h_j, \tag{15}$$

where the weight α_{tj} of each annotation h_t is given by:

$$\alpha_{tj} = \frac{\exp(e_{tj})}{\sum_{k=1}^n \exp(e_{tk})}. \tag{16}$$

In which:

$$e_{tj} = a(S_{t-1}, h_j). \tag{17}$$

The function $a(S_{t-1}, h_j)$ is an alignment model that describes the matching ability between the inputs around position j and the outputs at position t . The RNN hidden state S_{t-1} and the j -th annotation h_j of the input sentence are used to calculate the score. The attention mechanism allows a neural network to focus on a subset of its inputs: it always chooses the most relevant inputs. The attention mechanism in Fig. 3 aims to select the most important inputs from the input sequences x_1, x_2, \dots, x_n by using the weight α_{tj} .

Methods and materials

As mentioned above, our proposed model uses first data preprocessing techniques through applying feature selection and dimensionality reduction over credit card fraud datasets, to reduce the number of input features before fed into the model. Then the sequence learner LSTM is employed as the base dynamic pattern recognition classifier to capture the sequential dependency between consecutive credit card transactions. Next, attention mechanism is introduced to give different focus to the information outputted from the hidden layers of LSTM, which allow our model to discover relevant fraud patterns and detect better very dissimilar transactions within the purchases of a consumer. The architecture of the proposed system is shown in Fig. 4.

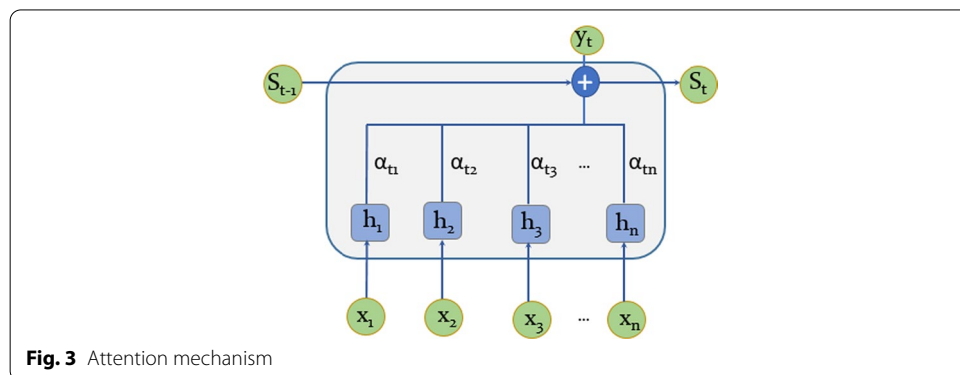


Fig. 3 Attention mechanism

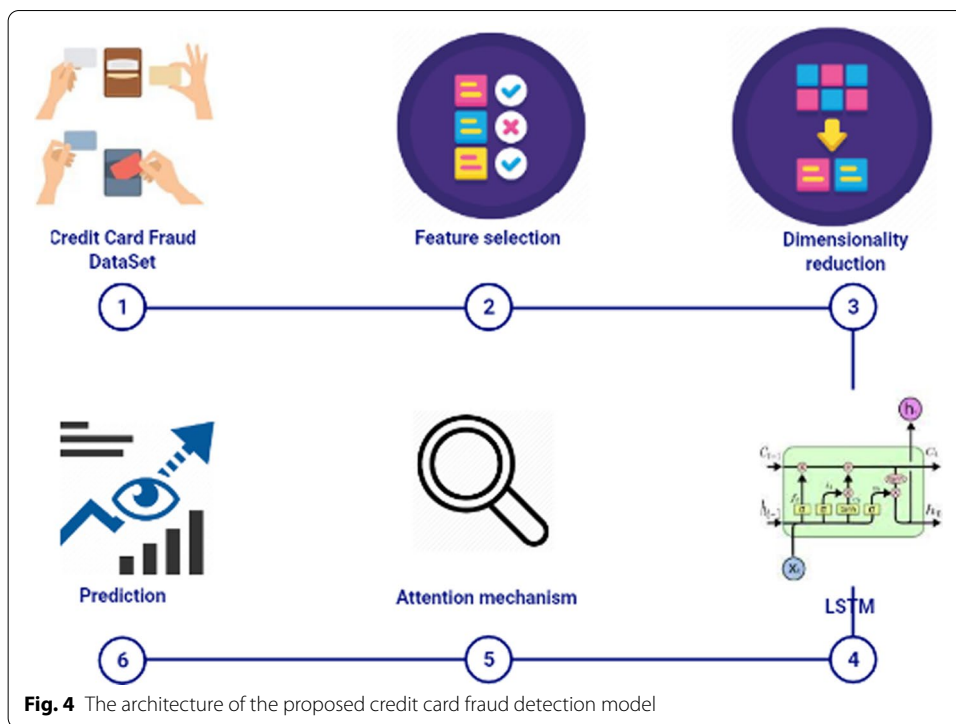


Fig. 4 The architecture of the proposed credit card fraud detection model

Table 1 The credit card datasets description

Name	Instances	Features	Normal	Fraudulent
Dataset-1	284,807	31	284,315	492
Dataset-2	594,643	10	587,443	7,200

The steps of our proposed model for credit card fraud detection are detailed below. We will describe the two datasets we use in our experiments, data preprocessing results and we will provide the detailed implementation and the evaluation metrics used in this work.

Datasets

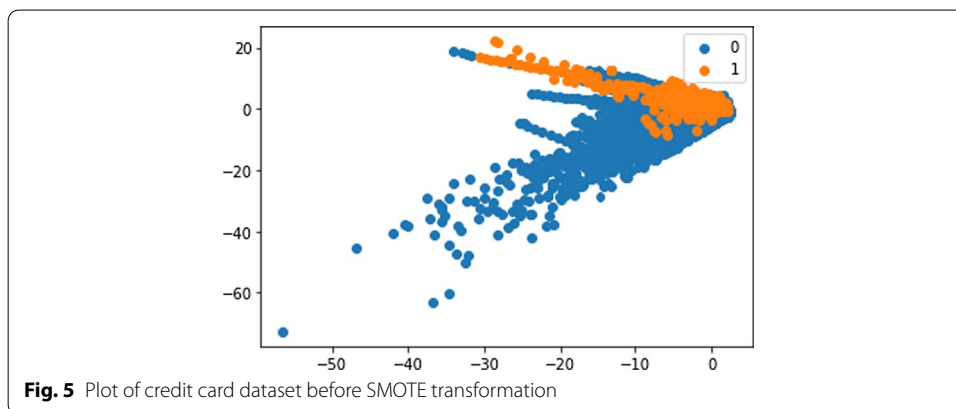
Datasets provide a way to train and validate the efficacy of the proposed methods, hence playing an important role in motivating research. In this subsection, we describe two different datasets used in the experimentations of our proposed approach. A brief summary of the two datasets is presented in Table 1.

Dataset-1

The first dataset downloaded from www.kaggle.com, consists of credit card transactions made by European cardholders occurring within two days in September 2013, where it has 492 frauds out of 284,807 transactions [54]. It consists of 31 features including the time when a transaction took place, the amount of transactions, and 28 other attributes labeled from V1 to V28 and the target label ‘Class’ which decides if a transaction is fraudulent or not by a binary value ‘1’ and ‘0’ respectively.

Table 2 Attributes of Dataset-2

Name	Description
Step	The day the transaction took place from 1 to 180
Customer ID	A number identifying the customer account involved in the transaction
Age category	A categorical value putting the customer into one of 8 different age groups
Gender	A categorical variable indicating the gender of the customer
Zip Code of account	The zip code associated with the customer
Merchant ID	A number identifying the merchant involved in the transaction
Zip Code of Merchant	The zip code of the merchant
Category purchase	A categorical variable indicating what type of good or service was purchased
Amount of purchase	The total amount that the transaction cost
Fraud status	A binary variable indicating if the transaction was fraudulent or not



Dataset-2

The second dataset consists of 594,643 transactions made during 180 simulated days, among which 7200 ($\approx 1.2\%$) are considered fraudulents. This is a synthetic dataset created for financial fraud detection by using BankSim software, which is a simulation tool specifically designed to emulate fraud data [55]. BankSim uses a multi-agent-based simulation methodology based on a sample of aggregated real transaction data that a bank in Spain offers. The original bank data is made up of thousands of transactional data records from November 2012 to April 2013. BankSim uses multiple agents of three different categories to mimic this original bank data: traders, customers, and fraudsters. These agents communicate with each other over a sequence of simulated days, resulting in a purchase transaction log closely resembling the original bank data. All attributes are presented in Table 2.

Dataset processing

We can see that our two datasets are highly imbalanced since the number of negative (majority) instances outnumbers the amount of positive (minority) class instances. In fact, for example in Dataset-1, frauds are typically less than 0.171% of the overall transactions, as shown in Fig. 5. Thus, to enhance the classification performance

of the minority fraud instances which is the class of great interest, we use the advanced oversampling technique called Synthetic Minority Oversampling Technique (SMOTE) [56, 57] to generate synthetic training instances from the minority class. Figure 6 presents the schematic diagram of the transformed Dataset-1 using SMOTE method.

Dimensionality reduction

Feature selection

As mentioned above, we use feature selection as a first step in the exploration of our datasets. The objective is to study the influence of each feature in the prediction of the target class and select the optimal subset of relevant features by removing redundant and noisy attributes. From the presented visual Swarm intelligence algorithm plots made for Dataset-1 in Fig. 7, we can see that the comparative analysis of credit card dataset demonstrates that features labeled Time, V5, V6, V7, V8, V9, V13, V15, V16, V18, V19, V20, V21, V22, V23, V24, V25, V26, V27, V28, amount do not contribute to the fraud prediction. Thus, we decide to consider them as irrelevant attributes and remove them from the original dataset. Table 3 presents the remain features.

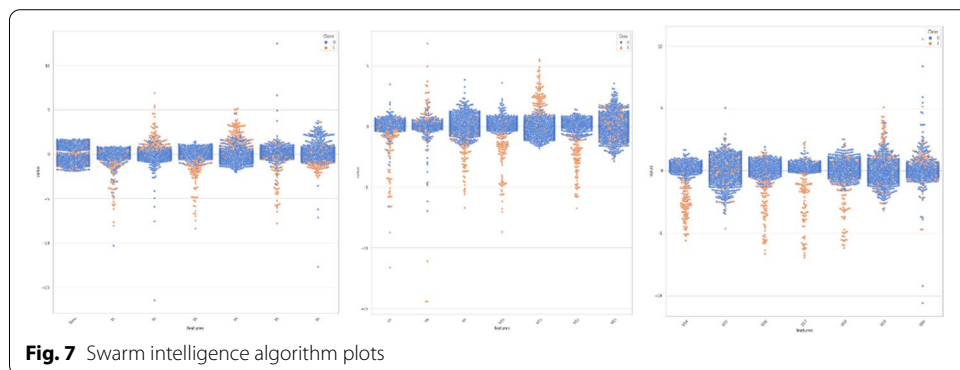
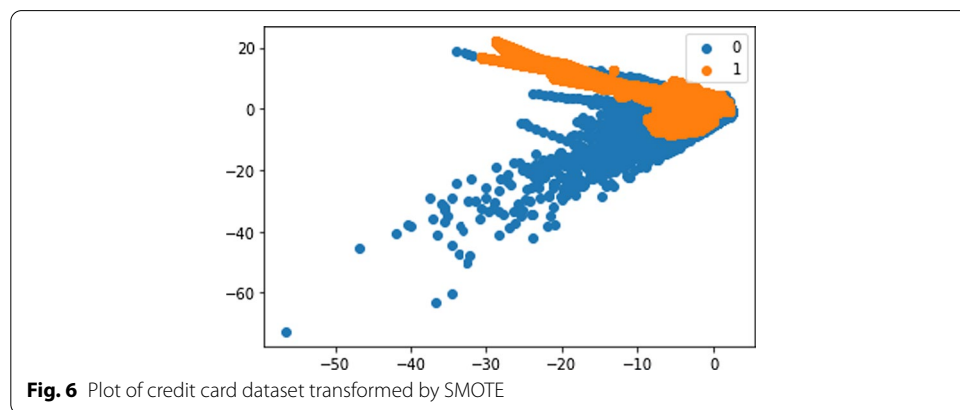


Table 3 The remain features after Swarm algorithm

V1	V2	V3	V4	V10
V11	V12	V14	V17	

Feature extraction

We applied the three dimensionality reduction algorithms PCA, t-SNE and UMAP on our credit card datasets to generate the robust and discriminative features for fraudulent instances, which will aid in detecting effective fraudulent transactions. Figure 8 shows the performance of each algorithm applied on Dataset-1. For each case, the dataset were reduced into the three dimensional space using default parameters, and the plots were colored according to the label of each data point in Dataset-1, namely the purple color is used to represent normal transactions and the orange color represents fraudulent transactions.

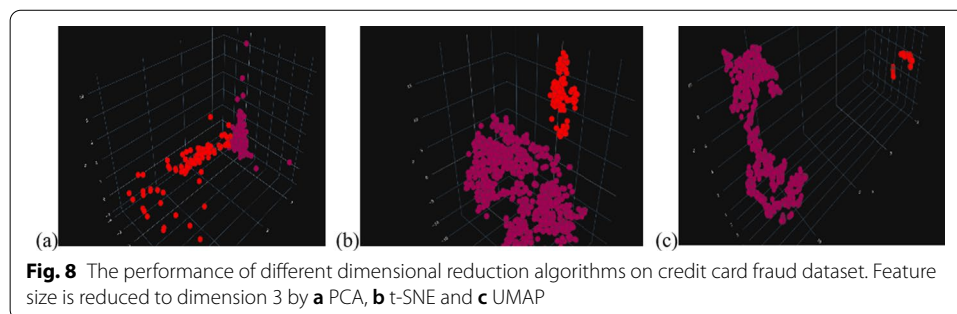
It can be seen that PCA does not present good discrimination, whereas UMAP and t-SNE show very good discrimination. However, Comparing t-SNE to UMAP, the latter is more able to preserve as much the local and the global data structure, with superior runtime performance. Based on this, we choose UMAP as a reduction algorithm to extract the embedding features that will be used during training and testing phases.

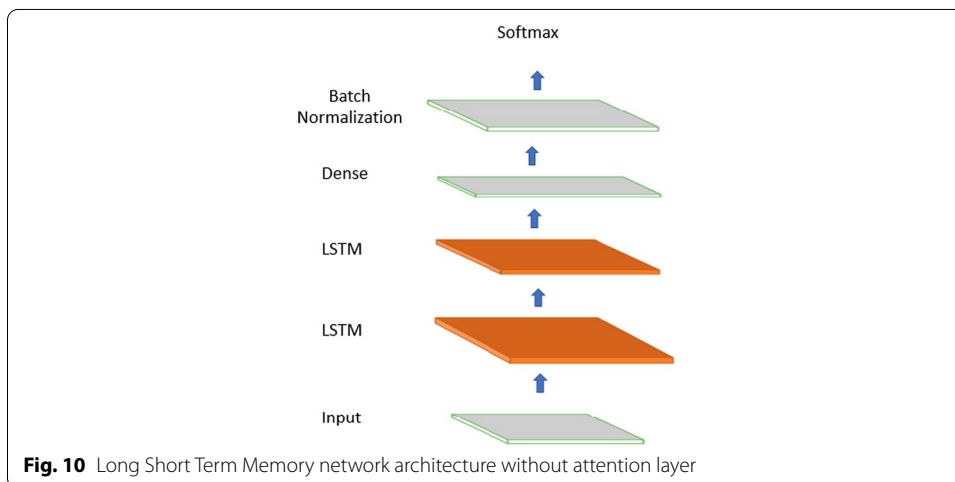
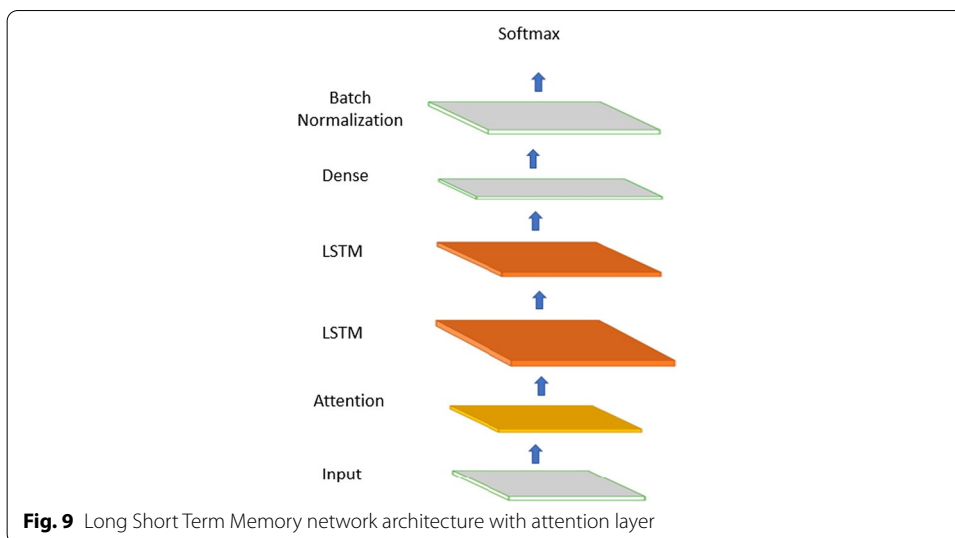
Implementation and evaluation metrics

We employ Long Short Term Memory networks to model the sequential dependency between consecutive transactions of credit card holders. The hidden state architecture of LSTM allows establishing connections between neural networks' nodes across time steps.

Therefore, the model can retain information from past inputs, allowing it to identify temporal associations between events that may be dispersed in the input sequence. LSTM is an adequate model of succession patterns in sequential data points where the occurrence of one event may depend further back in time on the presence of several other events. However, there are still much more aspects to improve:

1. LSTM networks have to represent the entire input sequence x_1, x_2, \dots, x_n as a single vector c , which can cause information loss since all information needs to be compressed into c . Furthermore, it need to decode the passed information from this single vector only, witch is a highly complex task. Thus, the performance of the LSTM networks degrade rapidly as the length of the input sequence increases.





2. LSTM networks process the input sequence elements with the same manner, there is no way to give more importance to some of the input elements compared to others while processing the sequence.

To address the problem above, we propose to add the attention mechanism upon LSTM layers which enables the classifier to automatically extract global dependencies from the sequence of transactions and focus on the data items that are most relevant to the classification task.

Our deep learning model is composed of 6 layers namely: Two layered LSTM networks followed by dropout on each layer, an attention layer added before the LSTM layer as depicted in Fig. 9. The LSTM layer takes the output of the attention layer as the input with the activation function assumed to be tanh. At the end of the two LSTM layers, we add a dense layer to obtain two valued outputs which are the prediction classes (normal transaction and fraud transaction). Finally, BatchNormalization layer is applied after the dense layer. The output of the BatchNormalization layer is passed into a softmax classification

layer. For comparison purpose, we also give the LSTM network without the attention layer in Fig. 10.

This model is based on Keras deep learning framework which is an opensource neural network library written in Python. The detailed workflow of the proposed model can be summarized as in Algorithm 1

Algorithm 1: Workflow of our proposed prediction model

Input: Historical credit card transactions collected up to time n :

$$x_1, x_2, \dots, x_n$$

Output: Prediction of fraud based on sequential transactions of Credit Card holder

- 1 Start
 - 2 Divide dataset into training, validation, and testing sets.
 - 3 Credit Card data is taken and reshaped into a three dimensional tensor (N, L, F) where N is the number of training sequences, L is the sequence length and F is the number of features of each sequence.
 - 4 A network structure is built with 6 layers, where there is one input layer and one attention layer followed by two LSTM networks in the next layers, a dense layer in the subsequent layer to obtain two valued outputs which are the prediction classes (normal and fraud), and finally a BatchNormalization layer is applied after the dense layer.
 - 5 Define learning parameters (memory size, learning rate, batch size and epochs) and set tensor variables for weight and bias vectors.
 - 6 Define cross-entropy loss function and add Adam optimization function to minimize the cross-entropy loss function.
 - 7 Train the constructed network on the Credit Card data.
 - 8 Use the output of the last layer as prediction of the next time step.
 - 9 **while** *Optimal convergence is not reached* **do**
 - 10 Compute training error.
 - 11 Compute validation error.
 - 12 Update weights and biases using back propagation.
 - 13 Obtain predictions by providing test data as input to the trained network.
 - 14 Evaluate accuracy by comparing predictions made with actual data.
 - 15 End
-

In credit card fraud domain, fraud detection systems try to reduce the false positive and false negative rate, knowing that the latter (FN) has severe costs on financial institutions as well as a decrease in customer satisfaction.

To assess the performance of our proposed fraud detection system with more accuracy, we use the confusion matrix as shown in Table 4. From this matrix, the following evaluation

Table 4 Classification confusion matrix

	Actual positive $y = 1$	Actual negative $y = 0$
Predicted positive $c = 1$	True positive (TP)	False positive (FP)
Predicted negative $c = 0$	False negative (FN)	True positive (TN)

metrics are extracted, namely: Accuracy, Sensitivity (or Recall), Specificity and Precision. These metrics are calculated as:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}, \quad (18)$$

$$\text{Sensitivity} = \frac{TP}{TP + FN}, \quad (19)$$

$$\text{Specificity} = \frac{TN}{FP + TN}, \quad (20)$$

$$\text{Precision} = \frac{TP}{TP + FP}. \quad (21)$$

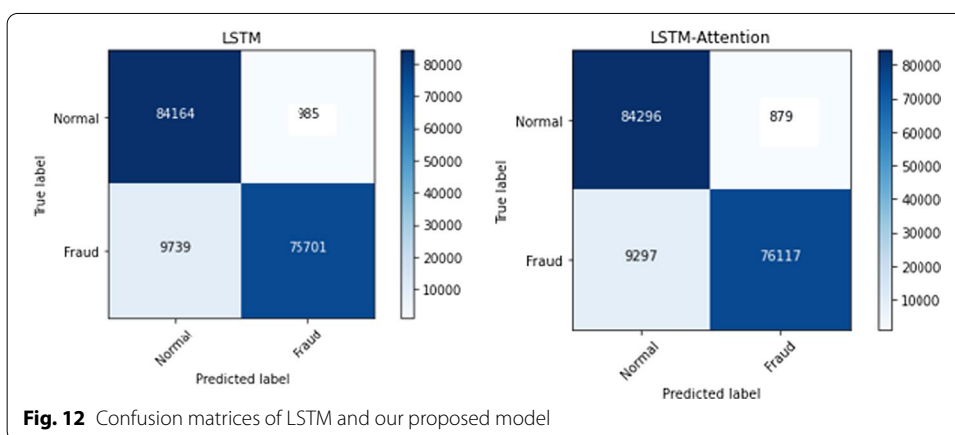
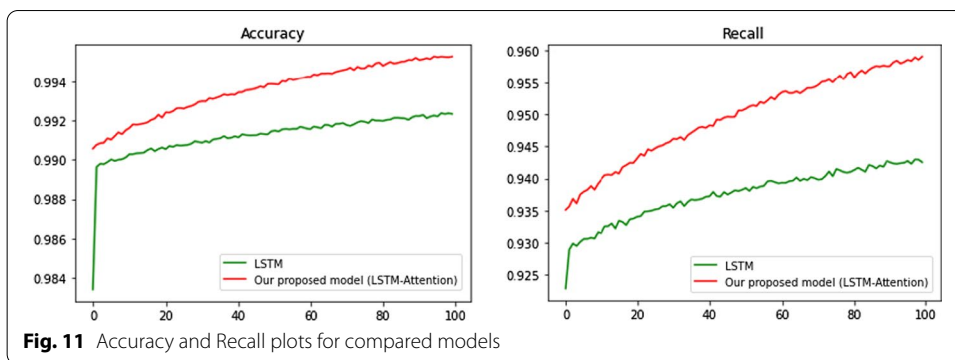
True positives (TP) are cases classified as positive which are actually positive. True negative (TN) are cases classified rightly as negative. False positive (FP) are cases classified as positive but are negative cases. False negative (FN) are cases classified as negative but are truly positive. Specificity gives the accuracy on negative (legitimate) cases classification. Precision gives the accuracy in cases classified as fraud (positive) and sensitivity (Recall) gives the accuracy on positive (fraud) cases classification.

However, when evaluating fraud detection models, financial institutions have to face many challenges, such as false positive rate and false negative rate. False positives (FP) are cases classified by the Fraud Detection System (FDS) as fraudulent transactions but represent in reality normal behaviors. These cases, although their resulted errors during classification, do not cause significant damage to financial institutions. False negatives (FN), on the other side, are cases identified wrongly by the FDS as normal transactions but are truly fraudulent ones which has severe costs on financial institutions as well as a decrease in customer satisfaction. Therefore, we will be interested more to the sensitivity (Recall) metric that gives the accuracy on positive (fraud) cases classification, which is the most appropriate evaluation metric in this domain to check the effectiveness of our proposed model.

Results and discussion

Our study is based on the aforementioned processed credit card fraud datasets characterized by the temporally ordered sequence of transactions which allow our proposed model to predict the label of a transaction after having seen several transactions that precede it. Each dataset is divided into three sets. The first 70% subset of data is the training set used for training the models, the second 15% subset of data is the validation set used for validating the classifiers to avoid overfitting and improve model performance and the last 15% test subset of data is used to test the network generalization. Same training set and testing set of the credit card data are chosen for comparison between our proposed model and the baseline LSTM model.

The accuracy and recall plots for both models applied for example over our dataset named Dataset-1 are presented in Fig. 11, from which we see that our model (LSTM-attention) achieved the more superior accuracy and sensitivity (recall) rates. This



significant improvement is because by using attention mechanism, more relevant patterns can be extracted from sequence transactions which allow the sequence classifier to automatically focus on the data items that are the most important to the classification task by a data-driven weighted average of each transaction contained in the sequence which results in an improved detection performance.

Moreover, to highlight the classification performance of our proposed model, in terms of sensitivity, we present a visualization of the confusion matrix performed on each model applied for example over our dataset Dataset-1 (Fig. 12) from which we illustrate that our proposed model has a good capability to minimize the number of fraudulent transactions classified as normal and catch the rare fraudulent transactions, which is of great importance in real life for financial service providers.

As well, to assess the analysis of our experimental results, we compared our work with state-of-the-art fraud detection models listed in Table 5. The major reason for selecting these models is that they exhibit promising performances and they use the same dataset Dataset-1 described in this work, hence making the comparison more practical and reliable. Table 5 shows the performance values of each used model, in term of accuracy, precision and sensitivity (recall). The latter metric is of high importance in fraud detection domain, since financial institutions are interested more in detecting fraud instances that may occur, to protect consumers’ interests and reduce the heavy annual financial losses caused by fraud.

Table 5 The accuracy, recall and precision metrics

Algorithms	Accuracy	Precision	Recall
GRU (2020) [58]	–	0.8626	0.7208
LSTM (2020) [58]	–	0.8575	0.7408
SVM (2021) [59]	0.9349	0.9743	0.8976
KNN (2021) [59]	0.9982	0.7142	0.0393
ANN (2021) [59]	0.9992	0.8115	0.7619
Our proposed model (LSTM-attention) over Dataset-1	0.9672	0.9885	0.9191
Our proposed model (LSTM-attention) over Dataset-2	0.9748	0.9769	0.9722

The best results on each dataset for our proposed model is in bold

As we can see from these experimental results, our proposed model achieves better results than the compared classification GRU, LSTM, SVM, KNN and ANN methods, which demonstrate the effectiveness of our proposed model in this paper on credit card fraud detection task.

Conclusion

In this paper, we aimed to improve the prediction efficiency during the identification of fraudulent transactions, by combining the strength of different Machine Learning techniques, namely: The Swarm intelligence based approach to select the optimal subset of relevant features, the Uniform Manifold Approximation and Projection (UMAP) method to reduce the dataset dimensionality, the Synthetic Minority Over-sampling Technique (SMOTE) to overcome the problem of imbalanced data, the sequence learner LSTM networks to model long term dependencies within transaction sequences and the attention mechanism to automatically focus on the data items that are the most relevant to the classification task. Thus, our proposed model is capable to catch useful patterns within consumer behavior which helps to distinguish effectively fraudulent transactions from the normal ones.

To validate our results, we performed our model on two different credit card datasets, and it shows its ability to deliver a high sensitivity performance during the detection of fraudulent instances that are of great interest in this domain. Furthermore, in terms of comparison with recent works, our model provides a very good performance.

As a future work, we plan to study a novel credit card fraud detection model that relies solely on attention and transformers architecture without using any recurrent networks to process sequences.

Acknowledgements

Not applicable.

Authors' contributions

All authors read and approved the final manuscript.

Funding

Not applicable. This research received no specific grant from any funding agency.

Availability of data and materials

Not applicable. For any collaboration, please contact the authors.

Declarations

Ethics approval and consent to participate

The author confirms the sole responsibility for this manuscript. The author read and approved the final manuscript

Consent for publication

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Competing interests

The authors declare that they have no competing interests.

Received: 24 June 2021 Accepted: 6 November 2021

Published online: 04 December 2021

References

1. ACFE. Report to the nations 2018 global study on occupational fraud and abuse. 2019. <https://doi.org/10.1002/9781118929773.oth1>.
2. Carcillo F, Le Borgne Y-A, Caelen O, Bontempi G. Streaming active learning strategies for real-life credit card fraud detection: assessment and visualization. *Int J Data Sci Anal*. 2018. <https://doi.org/10.1007/s41060-018-0116-z>.
3. Chandola V, Banerjee A, Kumar V. Anomaly detection for discrete sequences: a survey. *IEEE Trans Knowl Data Eng*. 2012;24:823–39.
4. Popat RR, Chaudhary J. A survey on credit card fraud detection using machine learning. In: Proceedings of the 2nd international conference on trends in electronics and informatics, ICOTI 2018; 2018. <https://doi.org/10.1109/ICOEI.2018.8553963>.
5. Zafar A, Sirshar M. A survey on application of Data Mining techniques; it's proficiency in fraud detection of credit card. *Res Rev J Eng Technol*. 2018;7:15–23.
6. Kültür Y, Çağlayan MU. Hybrid approaches for detecting credit card fraud. *Expert Syst*. 2017. <https://doi.org/10.1111/exsy.12191>.
7. Mohammed E, Far B. Supervised machine learning algorithms for credit card fraudulent transaction detection: a comparative study. In: *IEEE annals of the history of computing*. IEEE; 2018. <https://doi.org/10.1109/IRI.2018.00025>.
8. Carcillo F, Le Borgne Y-A, Caelen O, et al. Combining unsupervised and supervised learning in credit card fraud detection. *Inf Sci*. 2019. <https://doi.org/10.1016/j.ins.2019.05.042>.
9. Abdallah A, Maarof AM, Zainal A. Fraud detection system: a survey. *J Netw Comput Appl*. 2016;68:90–113.
10. Bhattacharyya S, Jha S, Tharakunnel K, Westland JC. Data mining for credit card fraud: a comparative study. *Decis Support Syst*. 2011;50(3):602–13.
11. Dhok SS, Bamnote GR. Credit card fraud detection using hidden Markov model. *Int J Adv Res Comput Sci*. 2012;3(3):816–20.
12. Srivastava A, Kundu A, Sural S, Member S. Credit card fraud detection using hidden Markov model. *IEEE Trans Dependable Secure Comput*. 2008;5(1):37–48.
13. Dal Pozzolo A, Johnson RA, Caelen O, Waterschoot S, Chawla NV, Bontempi G. Using HDDT to avoid instances propagation in unbalanced and evolving data streams. In: Proceedings of international joint conference on neural networks. 2014. p. 588–94.
14. Phua C, Lee V, Smith K, Gayler R. A comprehensive survey of data mining-based fraud detection research. 2010. [arXiv:1009.6119](https://arxiv.org/abs/1009.6119).
15. Sahin Y, Bulkan S, Duman E. A cost-sensitive decision tree approach for fraud detection. *Expert Syst Appl*. 2013;40(15):5916–23.
16. Dal Pozzolo A, Caelen O, Borgne Y-AL, Waterschoot S, Bontempi G. Learned lessons in credit card fraud detection from a practitioner perspective. *Expert Syst Appl*. 2014;41(10):4915–28.
17. Bahnsen AC, Aouada D, Stojanovic A, Ottersten B. Feature engineering strategies for credit card fraud detection. *Expert Syst Appl*. 2016;51(1):134–42.
18. Bahnsen AC, Stojanovic A, Aouada D. Cost sensitive credit card fraud detection using Bayes minimum risk. In: Proceedings of the 12th international conference on machine learning and applications, vol. 1. 2013. p. 333–8.
19. Van Vlasselaer V, Bravo C, Caelen O, Eliassi-Rad T, Akoglu L, Snoeck M, Baesens B. APATE: a novel approach for automated credit card transaction fraud detection using network-based extensions. *Decis Support Syst*. 2015;75:38–48.
20. Ganji VR, Mannem SNR. Credit card fraud detection using anti-k nearest neighbor algorithm. *Int J Comput Sci Eng*. 2012;4(6):1035–9.
21. Pun J, Lawryshyn Y. Improving credit card fraud detection using a meta-classification strategy. *Int J Comput Appl*. 2012;56(10):41–6.
22. Liu FT, Ting KM, Zhou Z-H. Isolation forest. In: Proceedings of the eighth IEEE international conference on data mining. 2008. p. 413–22.

23. Zhao X, Zhang J, Qin X. Loma: a local outlier mining algorithm based on attribute relevance analysis. *Expert Syst Appl.* 2017;84(30):272–80.
24. Hemalatha CS, Vaidehi V, Lakshmi R. Minimal infrequent pattern based approach for mining outliers in data streams. *Expert Syst Appl.* 2015;42(4):1998–2012.
25. Hlosta M, Striž R, Kupčík J, Zendulka J, Hruška T. Constrained classification of large imbalanced data by logistic regression and genetic algorithm. *Int J Mach Learn Comput.* 2013;3(2):214–8.
26. Benchaji I, Douzi S, El Ouahidi B. Novel learning strategy based on genetic programming for credit card fraud detection in Big Data. In: *Proceedings of international conference Big Data analytics, data mining and computational intelligence.* 2019. p. 3–10.
27. Donato JM, Schryver JC, Hinkel GC, Schmoyer RL, Leuze MR, Grandy NW. Mining multi-dimensional data for decision support. *Future Gener Comput Syst.* 1999;15:433–41.
28. Mahmoudi N, Duman E. Detecting credit card fraud by modified fisher discriminant analysis. *Expert Syst Appl.* 2015;42(5):2510–6.
29. Minegishi T, Niimi A. Proposal of credit card fraudulent use detection by online-type decision tree construction and verification of generality. *Int J Inf Secur Res.* 2011;1(4):229–35.
30. Dal Pozzolo A, Boracchi G, Caelen O, Alippi C, Bontempi G. Credit card fraud detection: a realistic modeling and a novel learning strategy. *IEEE Trans Neural Netw Learn Syst.* 2018;29(8):3784–97.
31. Quah JT, Sriganesh M. Real-time credit card fraud detection using computational intelligence. In: *Expert systems with applications*, vol. 35. Amsterdam: Elsevier; 2008. p. 1721–32.
32. Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. *Nature.* 1986;323(6088):533–6.
33. Elman JL. Finding structure in time. *Cogn Sci.* 1990;14(2):179–211.
34. Graves A, Jaitly, N. Towards end-to-end speech recognition with recurrent neural networks. In: *Proceedings of the 31st international conference on machine learning*, vol. 32. 2014. p. 1764–72.
35. Benchaji I, Douzi S, El Ouahidi B. Credit card fraud detection model based on LSTM recurrent neural networks. *J Adv Inf Technol.* 2021;12(2):113–8. <https://doi.org/10.12720/jait.12.2.113-118>.
36. Jurgovsky J, Granitzer M, Ziegler K, Calabretto S, Portier P, He-Guelton L, Caelen O. Sequence classification for credit-card fraud detection. *Appl Expert Syst.* 2018;100:234–45.
37. Bahdanau D, Cho K, Bengio Y. 2015. Neural machine translation by jointly learning to align and translate. In: *3rd international conference on learning representations, ICLR.* 2015.
38. Xu K, Ba J, Kiros R, Cho K, Courville A, Salakhudinov R, Zemel R, Bengio Y. Show, attend and tell: neural image caption generation with visual attention. In: *Proceedings of the 32nd international conference on machine learning.* 2015. p. 2048–57.
39. Gore S, Govindaraju V. Feature selection using cooperative game theory and relief algorithm. In: *Knowledge, information and creativity support systems: recent trends, advances and solutions.* Cham: Springer; 2016. p. 401–12.
40. West J, Bhattacharya M. Intelligent financial fraud detection: a comprehensive review. *Comput Secur.* 2016;57:47–66. <https://doi.org/10.1016/j.cose.2015.09.005>.
41. Kamaruddin SK, Ravi V. Credit card fraud detection using big data analytics: use of PSOANN based one-class classification. In: *ACM international conference proceeding series.* 2016. <https://doi.org/10.1145/2980258.2980319>.
42. Hormozi H, Hormozi E, Akbari MK, Javan MS. Credit cards fraud detection by negative selection algorithm on hadoop. In: *IKT 2013–2013 5th conference on information and knowledge technology.* 2013. <https://doi.org/10.1109/IKT.2013.6620035>.
43. Brezočnik L, Fister I Jr, Podgorelec VV. Swarm intelligence algorithms for feature selection: a review. *Appl Sci.* 2018;8:1521. <https://doi.org/10.3390/app8091521>.
44. Jolliffe IT, Cadima J. Principal component analysis: a review and recent developments. *Philos Trans Math Phys Eng Sci.* 2016;374(20150202):2065.
45. Linderman GC, Rachh M, Hoskins JG, Steinerberger S, Kluger Y. Fast interpolation-based t-SNE for improved visualization of single-cell RNA-seq data. *Nat Methods.* 2019;16(3):243–5.
46. Van der Maaten L, Hinton GG. Visualizing data using t-SNE. *J Mach Learn Res.* 2008;9(Nov):2579–605.
47. Becht E, McInnes L, Healy J, Dutertre C-A, Kwok IW, Ng LG, Ginhoux F, Newell EW. Dimensionality reduction for visualizing single-cell data using UMAP. *Nat Biotechnol.* 2019;37(1):38–44.
48. Chen J, Zhao R, Tong Y, Wei G-W. Evolutionary de Rham–Hodge method. 2019. arXiv preprint [arXiv:1912.12388](https://arxiv.org/abs/1912.12388).
49. McInnes L, Healy J, Melville J. UMAP: uniform manifold approximation and projection for dimension reduction. 2018. arXiv preprint [arXiv:1802.03426](https://arxiv.org/abs/1802.03426).
50. Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput.* 1997;9(8):1735–80.
51. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. *CoRR abs/1409.0473* (2014). [arXiv:1409.0473](https://arxiv.org/abs/1409.0473).
52. Chorowski J, Bahdanau D, Serdyuk D, Cho K, Bengio Y. Attention-based models for speech recognition. In: *Advances in neural information processing systems 28: annual conference on neural information processing systems 2015, December 7–12, 2015, Montreal, Quebec, Canada.* 2015. p. 577–85. <http://papers.nips.cc/paper/5847-attention-based-models-for-speech-recognition>.
53. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I. Attention is all you need. In: *Advances in neural information processing systems 30: annual conference on neural information processing systems 2017, 4–9 December 2017, Long Beach, CA, USA.* p. 6000–10. <http://papers.nips.cc/paper/7181-attention-is-all-you-need>.
54. Pozzolo AD, Caelen O, Johnson RA, Bontempi G. Calibrating probability with undersampling for unbalanced classification. In: *IEEE symposium series on computational intelligence.* 2015.
55. Vaughan G. Efficient big data model selection with applications to fraud detection. *Int J Forecast.* 2018;36(3):1116–27.

56. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: synthetic minority over-sampling technique. *J Artif Intell Res.* 2002;16:321–57.
57. Kumari P, Mishra SP. Analysis of credit card fraud detection using fusion classifiers. In: *Advances in intelligent systems and computing*. Cham: Springer; 2019. https://doi.org/10.1007/978-981-10-8055-5_11.
58. Forough J, Momtazi S. Ensemble of deep sequential models for credit card fraud detection. *Appl Soft Comput J.* 2020. <https://doi.org/10.1016/j.asoc.2020.106883>.
59. Asha RB, Suresh Kumar KR. Credit card fraud detection using artificial neural network. *Glob Transit Proc.* 2021. <https://doi.org/10.1016/j.glt.2021.01.006>.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
