Neural Computing and Applications (2022) 34:5839–5866
https://doi.org/10.1007/s00521-021-06695-z

ORIGINAL ARTICLE

A deep learning model for behavioural credit scoring in banks

Maher Ala'raj[1] · Maysam F. Abbod[2] · Munir Majdalawieh[1] · Luay Jum'a[3]

Received: 8 March 2021 / Accepted: 27 October 2021 / Published online: 14 January 2022

Abstract

The main aim of this paper is to help bank management in scoring credit card clients using machine learning by modelling and predicting the consumer behaviour concerning three aspects: the probability of single and consecutive missed payments for credit card customers, the purchasing behaviour of customers, and grouping customers based on a mathematical expectation of loss. Two models are developed: the first provides the probability of a missed payment during the next month for each customer, which is described as Missed payment prediction Long Short Term Memory model (MP-LSTM), whilst the second estimates the total monthly amount of purchases, which is defined as Purchase Estimation Prediction Long Short Term Memory model (PE-LSTM). Based on both models, a customer behavioural grouping is provided, which can be helpful for the bank's decision-making. Both models are trained on real credit card transactional datasets. Customer behavioural scores are analysed using classical performance evaluation measures. Calibration analysis of MP-LSTM scores showed that they could be considered as probabilities of missed payments. Obtained purchase estimations were analysed using mean square error and absolute error. The MP-LSTM model was compared to four traditional well-known machine learning algorithms. Experimental results show that, compared with conventional methods based on feature extraction, the consumer credit scoring method based on the MP-LSTM neural network has significantly improved consumer credit scoring.

Keywords LSTM · Neural networks · Behavioural scoring · Machine learning · Classification

1 Introduction

For banks and other financial institutions, credit lending products, such as credit cards, personal loans, mortgages, and corporate loans, are the core of their business, and good lending practices lead to high profits. Consequently, it is critical to banks that they effectively acquire new customers and maintain creditworthy ones. Over time, banks build an extensive customer database that can be analysed to evaluate the bank's performance and make strategic decisions. Not all customers behave similarly regarding financial behaviour; hence different treatments should be given to those who meet specific profitable measures based on their repayment or purchasing behaviour [30]. Customers exhibiting such behaviour can be offered more significant incentives and rewards. For banks to identify their good or bad customers, credit scoring and behavioural scoring are used. Hand et al. [38] defined credit scoring as the process of assessing the likelihood of applicants to default in their repayments or not. In addition, [9] defined it by splitting the term into two components: the first,

✉ Maher Ala'raj
maher.alaraj@zu.ac.ae

Maysam F. Abbod
maysam.abbod@brunel.ac.uk

Munir Majdalawieh
munir.majdalawieh@zu.ac.ae

Luay Jum'a
luay.juma@gju.edu.jo

[1] Department of Information Systems, College of Technological Innovation, Zayed University, Dubai 19282, UAE

[2] Department of Electronic and Computer Engineering, College of Engineering, Design and Physical Sciences, Brunel University London, Kingston Lane, Uxbridge UB8 3PH, UK

[3] Logistic Sciences Department, School of Management and Logistic Science, German Jordanian University, Madaba Street, 35247, Amman 11180, Jordan

"credit", meaning "buy now, pay later", and the second, "scoring", which is similar to the mechanism used for credit cards.

Credit scoring consists of two main types: application credit scoring, where a score is used to give a decision on a new credit application; and behavioural scoring, where the score is used to deal with existing customers after they were granted a loan [54]. Behavioural scoring is used by banks in guiding lending decisions in credit limit management strategies; managing debt collection and recovery; retaining future profitable customers; predicting accounts likely to close or settle early; offering new financial products and interest rates; managing dormant accounts; optimising telemarketing operations; and predicting fraudulent activity [19, 38, 46, 58, 59], the number of missed payments, and the future risk of late payment [67].

In addition, [12] have stressed the advantages of having dynamic models that estimate when customers will default or fail to repay as follows: (1) computing the profitability over a customer's lifetime and performing profit scoring; (2) providing the bank with an estimate of the default levels over time, which is helpful for debt provisioning; (3) helping to decide the term of the loan; (4) more easily incorporating changing economic conditions. As banks always try to predict borrowers' credibility as early as possible [52] and predict when the customer could miss a payment in general and a consecutive payment in particular, these types of models give the bank the ability to take early actions against any risk that results in undesirable behaviour by borrowers [63].

The focus of this paper is on behavioural scoring. According to [42], behavioural scoring is used to assess the behaviour of existing customers given their behavioural variables and predict their future purchasing behaviour or credit status. Behavioural scoring allows lenders to monitor the changing behaviour or characteristics of customers regularly and help coordinate customer-level decision-making.

1.1 Motivations and contributions

The primary source of credit card-related risk for banks is customer default, failure to repay debt. A default occurs when a borrower has missed payments, fails to make timely payments, or avoids or stops making credit card payments. Credit cards lack financial asset security for their debts; however, the lender has legal recourse if customers default paying the debt. Most credit card companies offer some months after a customer receives the debt before declaring the account defaulted. Although, if the account owner takes six months before making the payments, the account is fed off, and the lender counts loss on the account [20]. Consecutive missed payments for credit card debt are an early sign of customer insolvency. According to the Basel II convention, a consumer credit default is regarded as delinquency after 90 days [47]. Therefore, the motivation for this research paper is based on the necessity of automatically scoring the customer behaviour on repayments to make risk decisions. Using such scores, banks can split customers into "risk groups", which could help to detect potential bankruptcy early and block the customer's card in time to limit losses. Hence, the task of estimating the missed payment probability for clients who already have one or more missed payments turns out to be essential for bank management.

Apart from behaviour related to missed payments, it is crucial to predict purchase behaviour. A high estimated monthly purchase amount for persons who have multiple consecutive missed payments can be used as an alert indicator for making immediate necessary actions, so the client grouping procedure needs to consider customer behaviour related to their purchase amount. Customers that have a high risk of bankruptcy but do not spend a lot present less of a threat to the bank than customers of similar behaviour who make massive purchases.

The general process in behavioural credit scoring is to use the transaction purchase and payment history of former clients to compute and predict the risk of default for customers [58, 70]. The collected transaction data are used to build a behavioural scoring model that maps the attributes or features of each customer to the probability of missed payment or default. Transaction data for each customer has a temporal structure. Hence, to build a statistical or classical machine learning model, one has to extract a fixed number of features from the data [74]. However, such preprocessing methods are associated with information loss.

Moreover, features are usually selected manually based on expert knowledge of their subjective importance. The number of available features makes up the feature space. The high dimensionality of the feature space has advantages and some severe deficiencies [77].

Transaction datasets for the research might vary in size, nature, and the characteristics or the information they contain. The datasets might include missing values, noisy data, and transactions not relevant to the study. The data set may potentially cause issues in classifier training leading to the inability to capture the various correlation between the identified features and prediction labels (default, missed payment, fraud, etc.). More features require more computation time to train the model. There are inefficient scoring interpretation results and low model accuracy [53].

On the other hand, a small number of extracted features can lead to poor performance of the model. One solution is to perform a feature selection on the extracted features. To deal with erroneous or irrelevant entries in the data, [6, 7] the use of filtering techniques, while in [61], the authors

propose a framework for data transformation to extract useful features from raw transaction data.

Therefore, the main motivations of this paper are:

- The urgency for bank owners and management to have an accurate and timely prediction of consumer credit card default.
- The use of credit card scores in combination with predicted monthly purchases to make necessary financial security decisions.
- The challenging task of utilising all available transactional data to build a universal model that could solve both of those tasks without time-consuming and subjective pre-processing and feature extraction steps.
- The creation of a novel grouping technique based on the mathematical expectations of the loss for each customer.

The main aim of this paper is to help bank management in scoring credit card clients using machine learning. The main contributions and objectives of this paper, based on the above motivations, are to:

- Introduce a deep learning neural network architecture based on Long-Short Term Memory (LSTM) neural networks as a method of customer behaviour score estimation. The architecture is designed such that it can accept temporal multidimensional transaction data as input and incorporate it into the training process not only information related to transactions (amount, date, type, vendor code, etc.) but also demographic and financial information about the client (age, salary amount, country of origin) that allows the model to increase its efficiency.
- Prove the feasibility of LSTM mode and test it on the non-transactional dataset.
- Design two models based on the proposed architecture: the first (MP-LSTM) estimates the customer probability of missed payment validate on the transactional and non-transactional dataset. For customers with several missed payments in the recent past, this score is equivalent to the probability of default, and the model can trigger an alert to that effect. The second model (PE-LSTM) estimates customers' monthly purchase rate using the transactional dataset.

The MP-LSTM models are compared to four classical machine learning algorithms: Support Vector Machine (SVM), Random Forest (RF), Multi-Layer Perceptron Neural Network (MLP), and Logistic Regression (LogR). The paper emphasises the importance of conducting a detailed comparison procedure while proving high accuracy using the LSTM model that best satisfies the users' interests.

The rest of the paper is structured as follows: Sect. 2 provides a snapshot of the relevant literature on credit and behavioural scoring models. Section 3 explains the proposed methodology that is adopted in this paper. Section 4 describes the experimental setup, whilst Sect. 5 presents the experimental results and analysis. Finally, in Sect. 6, conclusions are drawn, and future work possibilities are discussed.

2 Literature review

2.1 Behavioural scoring

Credit scoring is an extensively researched topic in the financial industry by scholars [48]. Scholars have developed and proposed several models using statistical approaches, like Linear Discriminant Analysis (LDA) [13, 60] and LogR [46]. The financial crisis resulted in the Basel Committee on Banking Supervision requesting all compliant banking institutions to implement effective and efficient credit evaluation models in their credit management and tracking systems to support the issuance of loans to individual and corporate clients. Studies prove that that Artificial Intelligence (AI) procedures, such as neural networks, support vector machines, and random forest are appropriate replacements for empirical techniques in credit valuation modelling [11, 17, 71].

[70] presented a review of popular mathematical methods of behavioural scoring analysis. Among these methods, one can outline linear and logistic regression, decision trees, etc. Also, the author analysed the importance of incorporating economic conditions into the scoring systems. The other problem investigated in that paper is converting the probability of consumer defaults into an estimate of the profit or loss of a consumer organisation. In his later research, the same author investigated the usage of Markov chain stochastic processes for modelling the dynamics of consumer delinquency status and behavioural scores [69]. The paper also surveys behavioural scoring, customer scoring, and profit scoring approaches and objectives. A similar review was performed by [55], the authors have investigated the application of binary classification techniques for credit scoring financial analysis. The general results show the usability and importance of the main techniques for credit rating, as well as some of the scientific paradigm changes throughout the years.

Behavioural scoring models use recent customer's credit characteristics to predict their credit payment capability at a specific period. In most cases, the fixed payment period and the outcome period are arbitrarily selected, leading to instability in the prediction-making process. Kennedy et al. [46] evaluated the contrasting effects of adjusting the

outcome and performance periods [46]. The results from their work show that a 12-month performance period results in an easier prediction task when matched with other previous payment periods of varying timelines and that the performance of a logistic regression classifier reduces significantly after the resulting period is adjusted beyond six months.

2.2 Credit cards holders behavioural scoring

As seen above, most papers were focussed on behavioural scoring concerning customer loans. However, behavioural scoring of client's credit card payments has not been sufficiently investigated. Behavioural scoring models help to analyse purchasing behaviour of existing customers [64]. Only a few works have studied the mining of bank databases from the viewpoint of customer behavioural scoring [65]. To remedy this, [41] have used a Taiwanese bank credit card dataset to demonstrate the effectiveness of behavioural scoring. The authors use three commonly discussed data mining techniques: LDA, SVM, and Back Propagation Neural Networks (BPNN).

Sarlija et al. [63] have focussed on combining characteristics from behaviour and socioeconomic data. In a thesis [34], the author aims to apply multiple machine learning algorithms to analyse the default payment of credit cards. Bellotti and Crook [17] offered a valuable contribution in the studies exploring the utilisation of survival analysis to predict credit card defaults. The investigators utilised raw facts from a substantial credit card accounts database to model time to debt default, based on the foundations of survival analysis. The investigators adapted Cox proportional hazards survival modelling to formulate a seemingly superior model on time to default for a relatively voluminous credit card database. The model incorporated macroeconomic values (MV) as the sole time-varying covariates (TVC). The investigators also compared the use of logistic regression alongside the Cox model in the quantitative investigation of credit card defaults. The findings from the study indicated that the Cox proportional hazards survival model performed relatively well in predicting defaults compared to the conventional static logistic regression. The survey by Tony Bellotti and Crook [18] improved the initial model's performance and supported the integration of discrete survival analysis modelling. The resultant model comprised behavioural values (BV), application values, and MVs, with MVs and BVs acting as the TVCs. The research relied on the foundations of Monte Carlo simulation to support a stress test capability for assessing credit card defaults in the event of extreme economic conditions. Similarly, the stress test capability supported the determination of various influencing parameters and their effects on credit-card defaults. It ensured effective forecasting of credit card default rates in line with the specified model.

Agarwal et al. [4] utilised the Cox model to investigate cardholders' social capital's influence on potential credit card defaults. They incorporated the card holder's marital status, age, birthplace, family situation, and the distance between the birthplace and current address to define cardholders' social capital. According to Bellotti and Crook [18], customer's macro-environment variables, behaviour, and demographics are three critical variables that may enhance the performance of the Cox model in predicting credit card defaults. On the other hand Wang et al. [72] identified characteristics of the credit card, customers' attitude, demographics, and personality as the main determinants of credit card debt. Li et al. [51] utilised a standard LR model. The Cox survival analysis technique was used alongside the three NN algorithms to classify clients into good and bad categories based on their default likelihood. The investigators capitalised on findings from previous studies to investigate the factors influencing potential credit card defaults in China, focussing on formulating a dependable prediction framework. They used the Cox proportional hazards model to study how Chinese credit cardholder's behaviour, diversity, social capital, and independence influenced defaults. They also studied the impacts of macroeconomic environments on defaults. The investigators clustered consumer behaviour data into offline and online segments to evaluate the effect of increased online transactions on defaulted credit card payments.

Bellotti and Crook [17] have developed an application of survival analysis to model defaults on an extensive data set of credit card accounts was reported. The hypothesis that the probability of default is affected by general conditions in the economy over time was explored. The study of [76] has proposed using Weibo social, behavioural data as an additional source for the customer scoring model in the case of incomplete customer financial information. A unique credit evaluation index system was proposed, and it was proved that behavioural data was reasonable for this application. Hence, scholars should pay more attention to behavioural data in credit scoring studies.

2.3 Machine learning approaches in behavioural scoring

In recent years, loan and credit card transaction information has become significantly larger. Therefore, it is often impossible to use traditional mathematical and statistical models for such types of problems. To build behaviour scoring models, practitioners must consider several important issues, such as the extensiveness of the dataset to model, the planning horizon, and drivers of undesirable behaviour [46]. The literature does not contain strong

recommendations on how to answer these questions. Therefore, this paper investigates some of the issues affecting the building of a behavioural scoring model using machine learning by examining the performance of a large pool of credit card transactions datasets.

Pereira [62] aimed to understand the behaviour of credit card consumers depending on whether they do payment transactions involving a considerable amount of money. The study by [8] yielded an enhanced hybrid model of behavioural and credit valuation based on neural networks and data mining techniques for banking and marketing use. In the new hybrid model, an improved analysis method relying on frequency, monetary value, weighted recency, and credit scoring is developed. The model uses classification (MLP neural network) and clustering (*K*-means algorithm) techniques. A strategic clustering technique capable of identifying appropriate clusters through a combination of lower "average within-cluster distance" and relatively higher "CV for each cluster" indices was designed regarding the applicability of technical dimensions. The two-stage scoring mechanism with wide and deep learning application proposed in the study by Bastani et al. [14] is a combination of profit scoring and credit scoring. Stage 1 was developed to find non-default loans, which were then shifted to stage 2 for probability estimation. The scholars used wide and deep learning to build the predictive models in all the stages to accomplish both generalisation and memorisation. Akkoç [5] suggested a "three-stage hybrid Adaptive Neuro-Fuzzy Inference System credit scoring model". The model was founded on neuro-fuzzy and empirical methodologies [5]. The proposed model's performance was equated with traditional and frequently used models. A ten-fold cross-validation method relying on credit card data of a non-domestic bank in Turkey was employed to assess the effectiveness of the credit scoring models. In another study, Addo et al. [2] developed binary classifiers centred on the machine and deep learning models developed on real data to forecast loan default probability. The authors focussed on seven models: gradient boosting, random forest, elastic net (logistic regression with regularisation), and a neural network methodology with four diverse complexities. The top ten essential features from the above models were selected and used in the modelling process to examine the binary classifiers' stability by paralleling their performance on different data.

Wang et al. [71] proposed a consumer credit scoring technique centred on the attention mechanism LSTM about the user credit operation behaviour data from the peer-to-peer lending sector. This case suggested that deep learning methods could be applied differently. Each event type was treated as a word based on the idea foundations of the Word2vec model. Wang et al. formulated the Event2vec model with a focus on changing each type of event transformation to unique vectors. An attention mechanism LSTM network was later used to forecast a user's credit default probability.

Based on the relevant literature and to the best of our knowledge, there are no studies that apply LSTM neural networks to the task of predicting consecutive missed payments and defaults for customers' credit cards. For example, in Wang et al. [71], an LSTM neural network was used, but the application differs from this research field. This paper discovery is new since the application of LSTM neural networks to missed payment analysis with concurrent use of customer information, and macroeconomic factors have not been studied previously.

3 Methodology

Classical machine learning approaches cannot efficiently use all available data, which leads to bad scoring quality. To make use of classical algorithms, feature extraction needs to be performed [71]. For classification and regression problems of high-dimensional datasets, classical classification algorithms like Decision Trees, SVM, or Logistic Regression are time-consuming to train, and even trained models can show poor performance. That is why to adopt such classical methods; we need to extract features from the initial dataset. Such a two-step approach can lead to the loss of information at the stage of feature extraction. Moreover, feature extraction methods are expert-based, and that is why they are subjective. The way to avoid feature extraction as a first step is to use an appropriate neural network that can efficiently use without preprocessing.

3.1 Recurrent and LSTM neural networks

Recurrent neural networks (RNNs) are a particular class of supervised machine learning models. They are made of a sequence of cells with hidden states which have nonlinear dynamics. RNNs are used mainly with time-series data, for example, speech recognition [33], unsupervised anomaly detection [56], and automated translation [68]. LSTM is also used in economics to forecast time-series data as an alternative to the ARIMA model [66]. Transactional data in credit cards has a temporal nature; it is advisable to use RNNs instead of other types such as fully connected or convolutional neural networks.

In a recurrent neural network, connections between cells form directed cycles. Each cell contains a hidden state, which is updated on each iteration using its previous values. Such structure creates an internal network state and works as a memory.

The RNN equations are:

$$\begin{cases} S_t = f(U \cdot x_t + W \cdot s_{t-1}) \\ h_t = g(V \cdot s_t) \end{cases}, \quad (1)$$

where x is an input vector, s is a hidden vector of RNN layer values, h is an output vector of RNN layer values, U is a weight matrix of the input layer to the hidden layer, V is a weight matrix of the hidden layer to the output layer, W is a weight matrix for the previous time point to the current time point of the hidden layer, and g and f are activation functions for output and hidden layers, respectively. The structure of the usual RNN model is shown in Fig. 1.

In Fig. 1, work of one RNN cell is illustrated. We feed time-series signal X to the cell element by element. The vector X can be an input vector or output from other RNN cells from the previous layer. The RNN cell holds its state s. At each iteration t, the state s_t and output h_t are calculated by Eq. (1). Because of their architecture, RNNs can:

- Recognise patterns, characteristics, and dependencies in sequential and time-series data.
- Store, remember and process past complex signals for long periods.
- Map an input sequence to the output sequence at the current timestamp and predict the sequence in the next timestamp; and
- Reproduce any target dynamics after the training process, even with adjusted accuracy.

However, there are issues with learning long-term dependencies. Because RNN is prone to exploding or vanishing gradients during training, it is challenging to learn long-term dependencies. To solve this problem, [40] have proposed an LSTM based on RNN. As with RNNs, LSTM predictions are always conditioned by the experience of the network's inputs. Its distinguishing feature is special units called memory blocks in the recurrent hidden layer, which perform like accumulators of the state information. Every memory block has memory cells with self-connections, which store the temporal network state, and special multiplicative units called gates, which can control the stream of information. These cells and gates allow the LSTM to trap the gradient in the cell (also known as constant error carousels) and prevent it from vanishing. The gate activation functions are sigmoid; thus output value ranges from 0 to 1 and denotes how much information can be allowed to pass outside. The structure of a single LSTM cell is shown in Fig. 2.

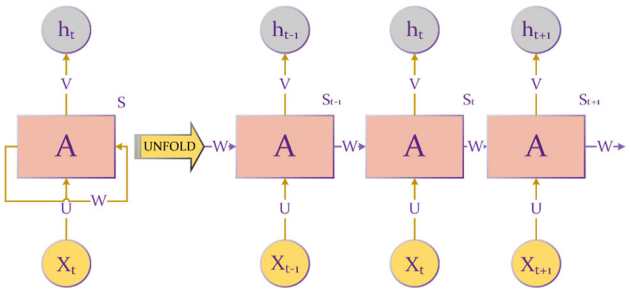

Fig. 1 RNN model structure [71]

As is seen in Fig. 2, an LSTM cell consists of three gates, namely an input gate, an output gate, and a forget gate. Two of these gates contain internal states. On each iteration t, the LSTM cell is using the previous values of the candidate vector C_{t-1} and output vector h_{t-1} to calculate their following values. The output of each gate is post-processed using activation functions. The shape of the activation function is essential and can significantly affect the efficiency of the neural network [33].

By default, the activation function of the recurrent gates is a sigmoid function, which is a nonlinear activation function used primarily on feedforward neural networks. It is a bounded monotonically increasing differentiable real function, defined for all real input values, as given by the following sigmoid function equation:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

The sigmoid function is applied to the output layers of the deep learning architectures in binary classification problems, modelling logistic regression tasks, as well as other neural network domains. However, the sigmoid activation function suffers significant drawbacks, which include sharp damp gradients during back propagation from deeper hidden layers to the input layers, gradient saturation, slow convergence, and nonzero-centred output, thereby causing the gradient updates to propagate in different directions.

The hyperbolic tangent function is the default activation function for an LSTM cell's output gate. The hyperbolic tangent function, tanh, is a smooth antisymmetric function with a range of values $[-1, 1]$. The output of the tanh function is given by.

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \quad (3)$$

The main advantage provided by tanh is that it produces zero-centred output, thereby aiding the backpropagation process. The detailed procedure of an LSTM cell is explained as follows:

On the first step, LSTM should decide which information to forget. For this purpose, the information of the previous memory state is processed through the forget gate f_t:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (4)$$

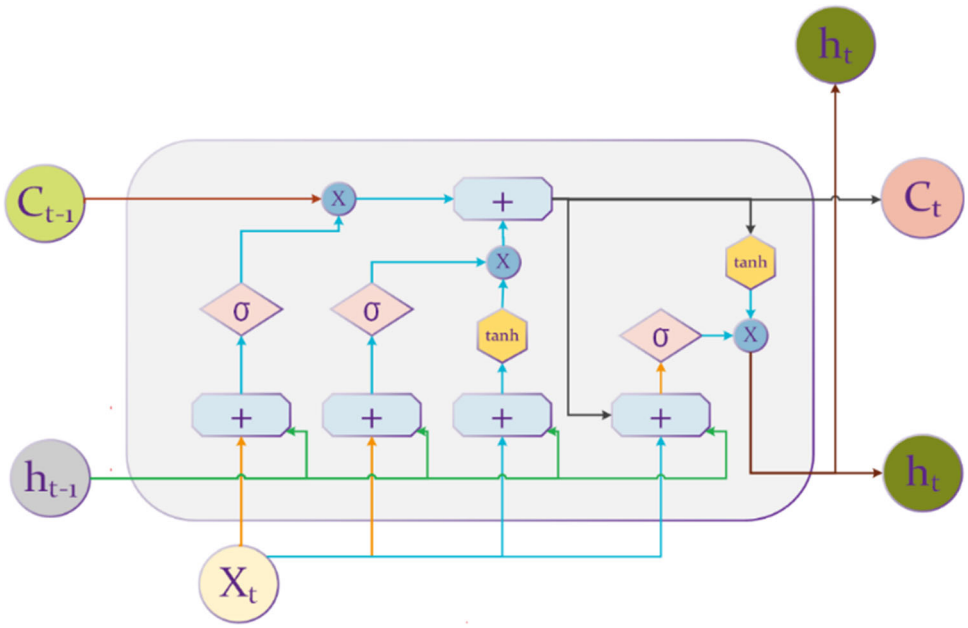

Fig. 2 LSTM model structure [71]

On the second step, input gates i_t decide which information should be updated, and the *tanh* layer updates the candidate vector $\tilde{C}_t$:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{5}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{6}$$

On the next step, memory states C_t are updated as a combination of the two parts above:

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{7}$$

Finally, output gates o_t are used for controlling the output h_t:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{8}$$

$$h_t = o_t \times \tanh(C_t) \tag{9}$$

Therefore, each LSTM layer is characterised by:

- Matrix W_f and vector b_f, which are parameters of the forget gate.
- Matrix W_C and vector b_C, which are parameters of the input gate; and
- Matrix W_o and vector b_o, which are parameters of the output gate.

However, LSTM networks have a limitation: they can work only with temporal data of the same length. That is why the LSTM architecture was modified to incorporate non-temporal customer data, like age, salary, nationality, or place of card issue, into the model.

3.2 Proposed model

Even though the LSTM neural network principles are already well studied, choosing the architecture is often up to the researcher. This includes selecting the number of LSTM layers, number of cells in each layer, activation functions, etc.

To use the LSTM architecture in the behavioural scoring task, it must be modified to make it possible to use transactional data and other customer data (age, salary, country of origin, etc.) The importance of demographic data lies in the inherent difference in payment behaviour for different age or country groups. For example, clients from a specific country might delay payments more often than clients from another country. As a result, two models are presented:

- Missed payment prediction LSTM model (MP-LSTM).
- Purchase estimation LSTM model (PE-LSTM).

The MP-LSTM model aims to automate credit card behaviour scoring for customers and trigger an early alert for credit card default. The PE-LSTM model seeks to estimate the monthly purchase amount for customers, which (combined with the score from the MP-LSTM model) can select potentially insolvent customers who will spend a lot during the next month.

The framework of both models is presented in Fig. 3. This framework consists of data pre-processing, training, and testing data generation, models training and testing, and models evaluation.

The architecture of the two models that are mentioned in Fig. 3 is described in Fig. 4.

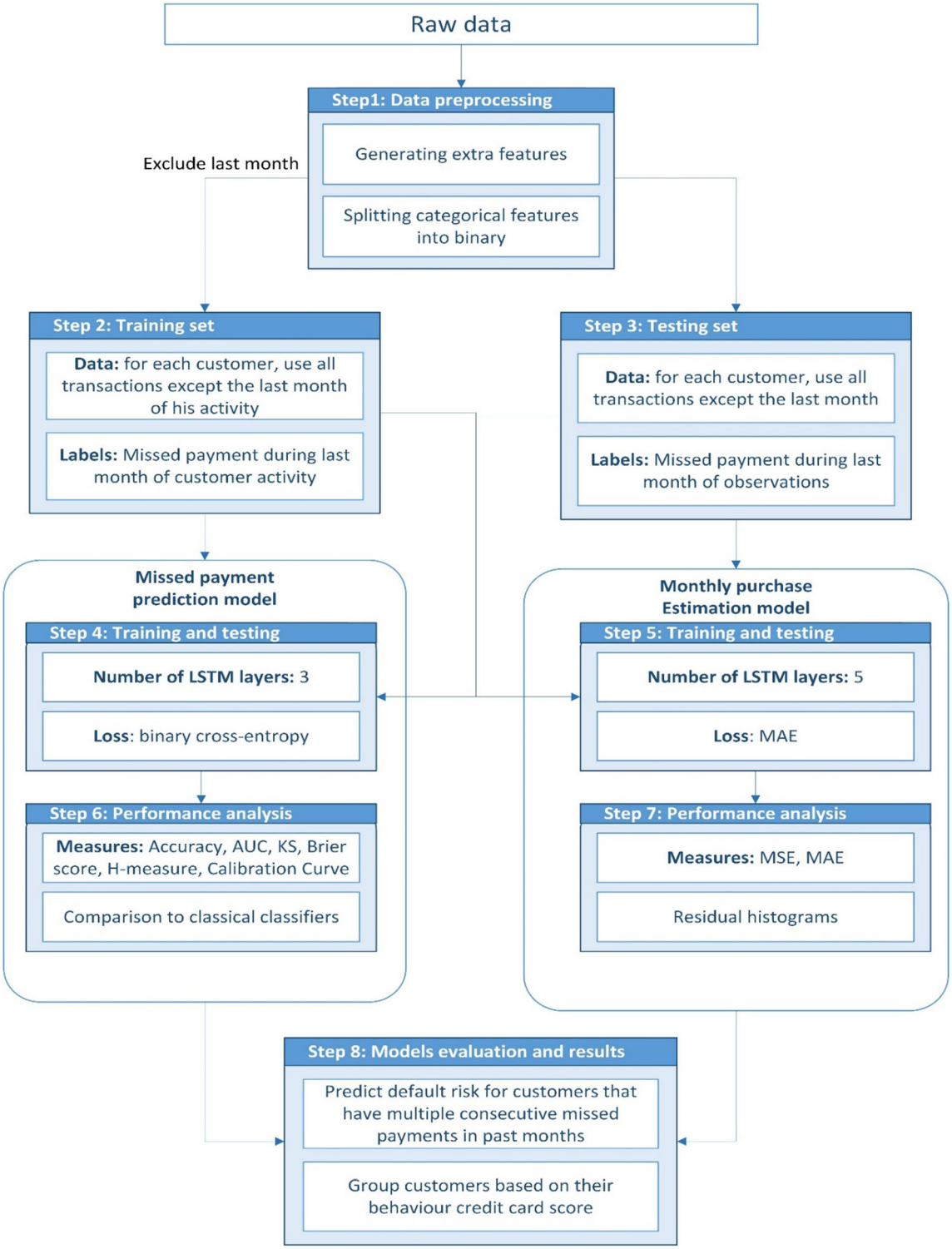

Fig. 3 The framework of the proposed LSTM models

As it is seen from Fig. 4, the first layers are LSTM, while the last layers are fully connected (dense layers). It means that all neurons from the previous layer are connected to the neurons with the current layer. The output of the last LSTM layer is concatenated to the vector of additional information.

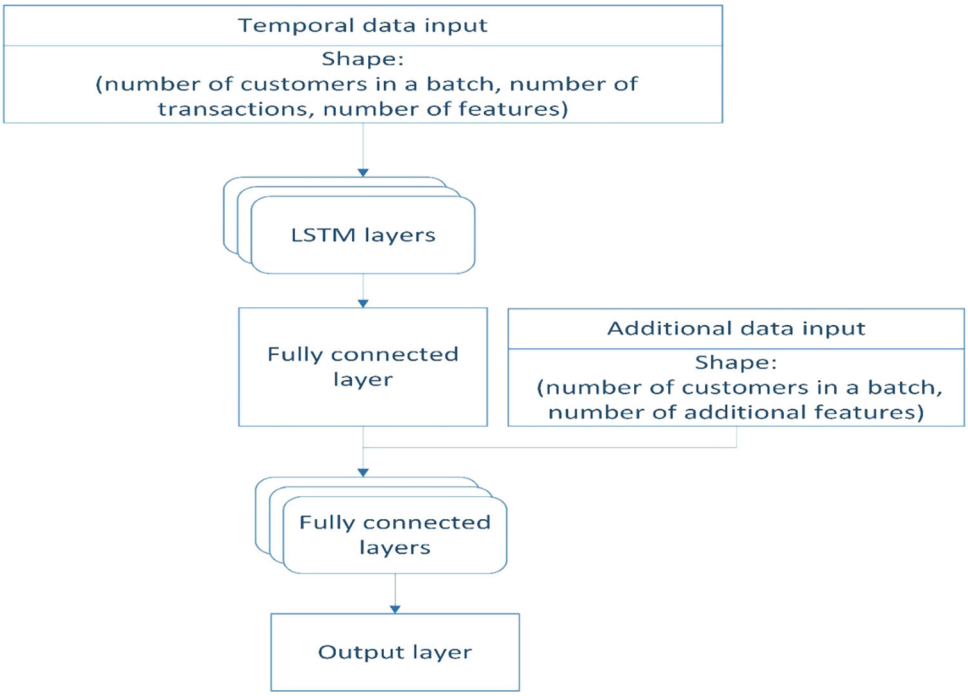

Fig. 4 Neural network architecture for MP-LSTM and PE-LSTM models

Table 1 shows the hyper parameters for the developed models. The model for monthly purchase estimation is more complex than the one for missed payment prediction.

The number of neurons in each layer was selected using grid search, and activation functions were determined by adopting the most commonly used from similar researches [27, 71]. In general, it is hard to find hyper parameters of complex neural networks, so for the number of layers, we used a small set with values {3, 5, 10, 15}, for the number of cells we used {4, 8, 16, 32}. Such a small number of LSTM layers can be explained by the low dimensionality of input data, which is on average equals 2500, much less than the dimensionality of visual or audial data, for which LSTM neural networks are usually used.

4 Experimental design

4.1 Datasets

To verify the practicality and effectiveness of the proposed LSTM models, two datasets are applied. The first dataset is confidential and an anonymous transactional dataset containing 55 months of real customer credit card transactions from a private bank. The second dataset is a public[1] non-transactional credit card dataset used in Yeh and Lien [75]. It was not possible to use the algorithm on other datasets because the vast majority of credit card datasets have already features extracted (they are in pre-processed form). For such datasets using the LSTM model makes no sense because of the low dimensionality and non-temporal structure of data.

4.1.1 Transactional dataset description

The dataset contains information of 25,964 customers that made around 1.9 million transactions; five variables are related to the customers' transactions; the target variable is the probability that the customer will miss the due amount payment on the credit card in the next month or not. The transactional dataset contains information about every transaction, namely:

- Customer unique key.
- Transaction amount.
- Transaction date.
- Transaction post-date.
- Transaction code.
- Vendor code.

A summary of the dataset is illustrated in Table 2, along with some analytics about the dataset.

4.1.2 Non-transactional dataset description

The second dataset is a public non-transactional credit cards dataset that reflects customer's default payments in Taiwan [75]. The size of the data set is 30,000 records, which is large enough to test the efficiency of the proposed

[1] The dataset is available at https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients.

Table 1 Hyper parameters for developed models

Model task	MP-LSM (transactional dataset)	MP-LSTM (non-transactional dataset)	PE-LSTM (transactional dataset)
Number of transactional features	25	3	25
Number of additional features	41	8	41
Number of LSTM layers	3	2	5
Number of cells in each LSTM layer	16	4	16
Activation function of hidden LSTM layers	Hyperbolic tangent	Hyperbolic tangent	Hyperbolic tangent
Number of Fully connected layers (Dense)	3	2	3
Activation function for all Dense layers except the output one	Sigmoid	Sigmoid	Sigmoid
Activation function of the output layer	Sigmoid	Sigmoid	Linear
Loss function	Binary cross-entropy	Binary cross-entropy	Mean average error
Optimizer	Adam	Adam	Adam

Table 2 Private bank transactional dataset properties

Attribute	Value
Transaction dates	January 2014–July 2018
Number of transactions	1,931,162
Number of purchase transactions	1,784,018
Number of clients	25,964
Number of active clients	23,833
Number of clients with at least one missed payment	11,701
Number of clients with two or more consecutive missed payments	7,088
Number of clients with three or more consecutive missed payments	5499
Number of clients with three or more consecutive missed payments during the last month (potential to default)	886
Number of clients with four or more consecutive missed payments	4613

model. The number of non-default payments is 23,364, while the number of default payments is 6636 (the proportion of default payments in the dataset is 22%).

In the dataset the following 23 variables are used as explanatory:

- X1: Amount of the given credit, which includes both the individual consumer credit and his/her family (supplementary) credit
- X2: Gender (1 = male; 2 = female)
- X3: Education (1 = graduate school; 2 = university; 3 = high school; 4 = others)
- X4: Marital status (1 = married; 2 = single; 3 = others)
- X5: Age (year)
- X6–X11: History of past payment. Tracked payment records are denoted from September to April 2005 by X6–X11, respectively. The measurement scale for the repayment status is: − 1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months;...; 8 = payment delay for eight months; 9 = payment delay for nine months and above.
- X12–X17: Amount of bill statement. The amount of the billing statement is denoted from September to April 2005 by X12–X17, respectively.
- Amount of previous payment (NT dollar). X18 = amount paid in September 2005; X19 = amount paid in August 2005;…; X23 = amount paid in April 2005.

The variables can be divided into two groups: numerical and categorical. The examples of the first are X1 (amount of given credits), X5 (age), X6–X11 (history of past payment), etc. The second group contains such variables: X2 (gender), X3 (education), X4 (marital status).

As it can be seen, this dataset is obtained from raw transactional data like the one we have in the first transactional dataset. Columns X6–X23 were obtained by querying the dataset and grouping payment records for each month from April to September 2005.

4.2 Data pre-processing

It is important to prepare data before building the models. Data pre-processing supports the imputation or deletion of missing entries that may hinder effective knowledge discovery processes. While deletion is considered the easiest way of handling missing entries, other procedures may provide more realistic results. For instance, imputation or the replacement of missing entries with newly estimated values may enhance the reliability, validity, and specificity of the model. The investigators utilised the following strategies from Acuna and Rodriguez [1] to impute missing entries in the project dataset.

- Replace missing categorical or nominal data with the most frequent category within the remaining entries, in other words, the mode.
- Replace missing quantitative data with the mean value of the feature that holds that missing value.

As an LSTM neural network accepts only numerical data, it is necessary to extract usable information from the transactional data. Also, categorical variables should be split into binary columns before feeding into the neural network [36]. To treat missing values in the transaction postdate column, a new binary column, which represents that the postdate value is missing, is created.

In Fig. 5, one can see the detailed framework for data pre-processing, which is organised as follows:

- Initial data cleaning.
- Generating new features.
- Splitting categorical features into the set of binary columns.

As a first step, the data needs to be cleaned and transformed into a correct form. All input tables were concatenated into one transaction data frame. Based on the "Transaction date" value, the following additional columns were created:

- Days after the client's previous transaction, or zero if it is the first one.
- Day of the week when the transaction was performed.
- Day of the month when the transaction was performed.
- Month when the transaction was performed.
- Year when the transaction was performed.
- The difference in days between the transaction postdates and the transaction date.
- A binary column that represents whether the transaction postdates is empty.

The "Vendor code" column was replaced by the vendor group. The "Transaction code" column was replaced by the type of each transaction and its sign (Debit or Credit). The reason for this replacement lies in a large number of levels in each of these categorical variables. Finally, the "Transaction sign" and "Vendor group" columns were split into indicator columns. Typically, an indicator column has a value of one when a categorical variable is equal to a specific value and zero otherwise.

4.3 Data partitioning

The main idea behind data partitioning is to break the data into two parts: one for learning and the other for testing. Different splitting methods have been used in the credit scoring literature: the most common is to partition the dataset into training (learning) and testing (evaluation) sets using a cross-validation approach [32].

For the Taiwanese public dataset, we used a five-fold cross-validation approach. All datasets were split into five disjoint subsets, and for each *i-th* fold, we use *i-th* subset

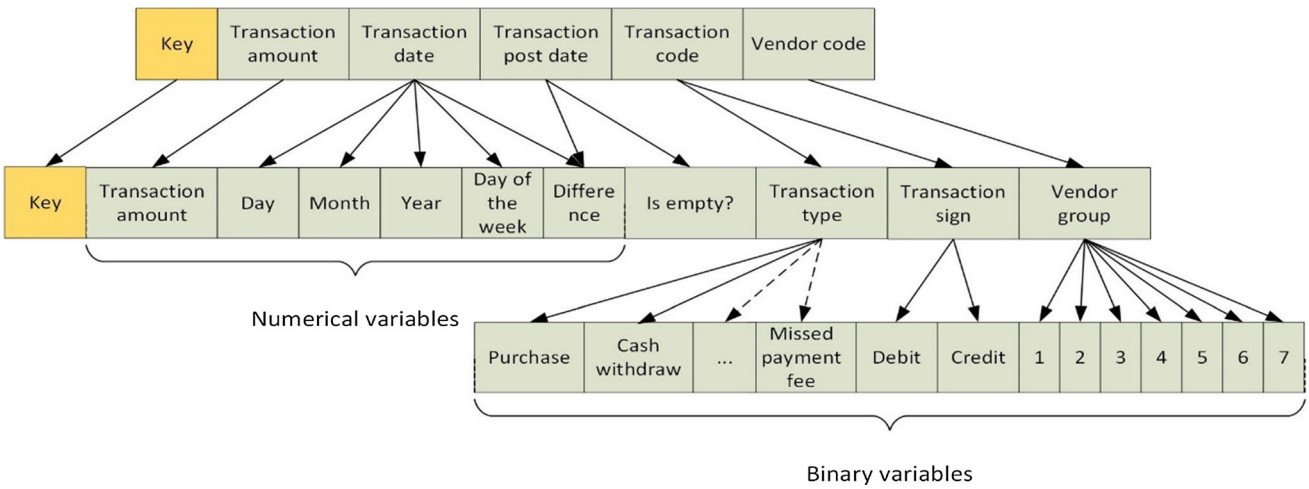

Fig. 5 Data pre-processing module (step 1 in the framework)

for testing and all other subsets for training. Naturally, each client belongs to exactly one-fold, and all clients are assigned to some fold.

However, for time-series data, such an approach makes no sense because newer data depends on older data and not vice versa [15]. Therefore, it cannot be used to train the model on recent data and evaluate its performance on the older data. Hence, it is worth mentioning the process applied for splitting the consecutive transactional data into training and testing sets. This process is not as straightforward as in non-temporal classification problems. The splitting procedure is shown in Fig. 6.

The last month of available transaction data is partitioned as the testing labels. The testing labels are evaluated for missed payment fees during the last month for the MP-LSTM model, while the PE-LSTM model is assessed for the total purchases during the last month. For the non-transactional dataset, five-fold cross-validation and average performance measures over all five folds are used. To train the model and extract training labels, the training data is partitioned into two types, training data and training labels. For the MP-LSTM model, the training labels are stored as a binary vector which denotes whether each client had a missed payment fee during their last month of activity. While the PE-LSTM model, the training labels are the total amount of purchases during the last month of activity of each client.

To feed training and validation transactions into the LSTM network, clients for training and validation data should be divided into groups (buckets). The data for each group is converted into a 3-dimensional array with the dimensions (number of clients in the group; max transaction size for the group; the number of features for every transaction). Information about every client is added as a subarray with zero rows; the group is formed by similarity in transaction count. To begin the groups, their upper and lower transaction count limits must be defined beforehand. To prevent creating huge groups with a lot of dominant zero rows, it was decided to combine clients with a more significant number of transactions into smaller groups. This was achieved by choosing quantiles following the power-law distribution with exponent 0.2. According to this rule, the whole data frame was split by transaction count for every client. Based on the chosen quantiles, it is evident that the clients with smaller transaction counts will be grouped into larger groups. The number of transactions for all clients in each group should be equal. Therefore, "dummy" empty transactions are prepended to each client's data to reach the target size for their group.

The final training and testing data consist of:

- A list of 3-dimensional arrays for each group containing the transactional data for each client.
- A list of additional information about customers in each group (client's age, country of origin, salary, city where he obtained his credit card, etc.).
- MP-LSTM model label (binary column which denotes missed payment presence); and

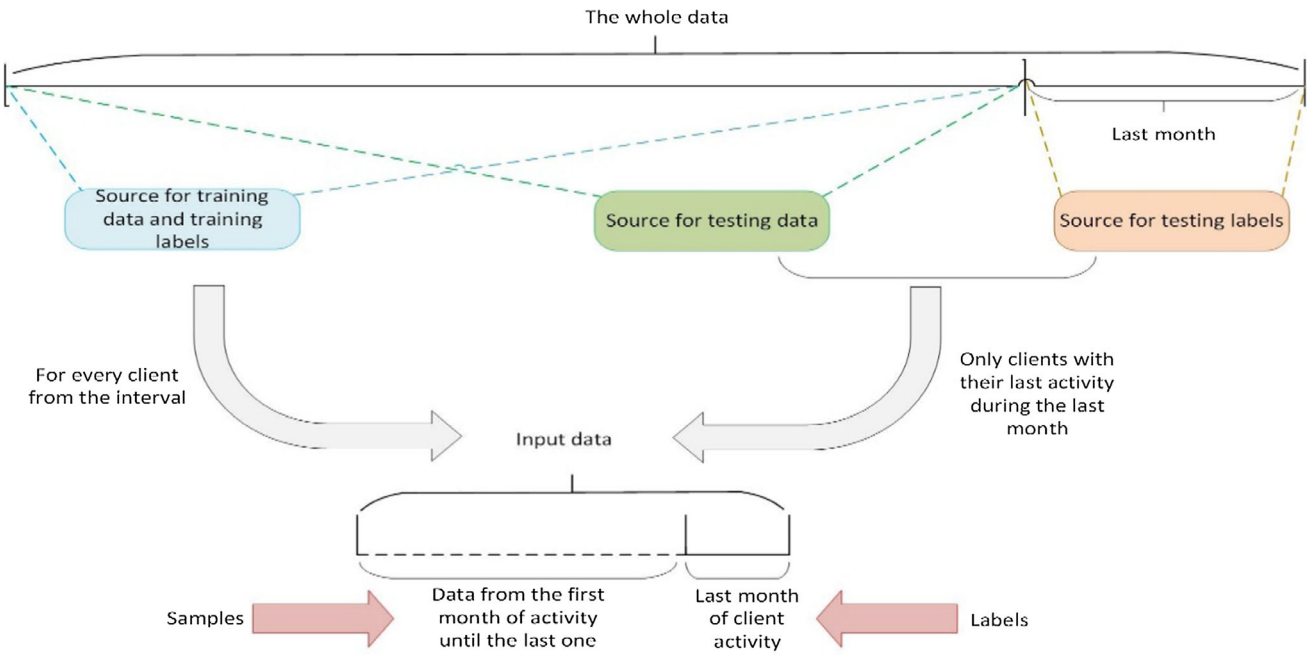

Fig. 6 Splitting data into training and testing set (step 3 and 4 in the framework)

- PE-LSTM model label (numerical column which denotes monthly purchase amount).

4.4 Benchmark model development

The proposed model's results are compared to four benchmark models, including SVM, LogR, MLP, and RF, to assess its accuracy levels. The LogR model is considered an industry standard for developing optimal credit scoring models ([16, 50]. However, Lessmann et al. [50] emphasised the need to compare newly formulated modelling strategies with the standard and other existing models. Several studies utilised MLP, RF, and SVM as benchmark models [21, 35]. The following sections provide comprehensive discussions on the theoretical background of the models.

4.4.1 Multi-layer perceptron neural network

According to Haykin [39], Neural Networks (NN) are machine learning systems conceptualised from the physical appearance of the biological neuron. NN models are formulated to allow them to imitate the functions of the human brain, explicitly capturing multifaceted relations between multiple inputs, signals, and outputs [22]. The multi-layer perceptron is a critical architecture of NN. The layer comprises a single input layer, hidden layer(s), and an output layer. The structure, topology, and learning algorithms are the essential factors to consider when formulating NN models [10]. The three-layer feedforward network backpropagation is the most applied MLP topology. For instance, to develop a credit scoring training subset characterised with the input $x = \{x_1, x_2, \ldots, x_n\}$; the propagation of the MLP model will follow one direction. The model will start after entering data subset x to the relevant layer (in this scenario, x represents the customer's characteristics or attributes). After entering data, the NN utilises links or synapses to send the inputs to hidden layers linked with every input's initial random weight. The hidden layer processes the received signals and applies activation functions accordingly. The result is presented as the output layer's weighted input. The output layer processes the consecutive inputs and establishes a final decision after applying the activation function [57].

4.4.2 Support vector machine

A support vector machine is an effective and efficient machine learning method utilised to classify credit data for scoring purposes. The strategy is also applied in binary classification undertakings to support fine separation surfaces splitting input data into dominant classes. For an instant, SVM may result in good and bad credit score classes. Cortes and Vapnik [26] initiated the utilisation of SVM modelling after their investigations produced an improved version of the pre-existing linear classifiers. A function capable of mapping data into higher dimensional space makes SVM superior to linear classifiers. Linear, polynomial, sigmoid, and radial basis kernel functions were proposed to achieve the specificity of SVM. As a result, an SVM utilises a linear model to map two-class linear data to a high-dimensional space and then implements the nonlinear classes. The nonlinear decision margin existing in the initial space is denoted by the linear model positioned in the novel feature space. As a result, the SVM constructs a hyperplane or an optimal line that separates the two classes perfectly in the space. The relatively higher accuracy levels of SVMs have influenced their utilisation in generalised predictions and credit scoring [43, 49].

4.4.3 Random forest

According to Breiman [23], a random forest model is considered an advanced decision tree (DT) technique consisting of many trees. The generation of n subsets from the primary datasets supports the creation of the trees. Each subset is considered a tree that originates from the random selection of variables, hence the name "random forest". After the generation and training of all the DTs, a voting procedure is utilised to develop the final decision class. The trees determine the most popular class, which s considered as the ultimate output class by the random forest model.

4.4.4 Logistic regression

LogR is considered the industry standard for developing efficient and effective credit scoring models [50]. The method is used extensively and is famous for providing optimal solutions for regression and classification problems. LogR is mainly employed to model binary outcome variables. The variables are commonly signified by 0 or 1 to imply good and bad or no and yes. Atiya and Parlos [11] expressed LogR formula as follows;

$$\log\left[\frac{p}{1-p}\right] = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n \tag{10}$$

In the LogR formula, p represents the dependent variable's probability, while β_0 represents the intercept term, the coefficients associated with the predictor variables $X_i,\ i \in 1, 2, \ldots, n$, is represented by β_i. On the other hand, $\log\left[\frac{p}{1-p}\right]$ represents the response (predicted) variable. This metric is computed by determining the ratio of the two probability outcomes and, after that calculating the logarithm. LogR is a critical statistic in credit scoring because it

helps determine the specific input's conditional probability (customer's characteristics) associated with a particular scoring class.

4.5 Performance measure metrics

The primary investigator implemented five performance indicator measures to evaluate the proposed model to support a robust and reliable conclusion on its predictive specificity. The metrics included (1) Accuracy, (2) Area Under the Curve (AUC), (3) H-measure, (4) Kolmogorov–Smirnov (KS) chart and (5) Brier's score. The performance measures' popularity in credit scoring and their capacity to give comprehensive views on the overall model performance supported their selection. Accuracy was explicitly utilised to assess the model's predictive power based on the proportion of correctly classified bad and good credit products. Lessmann et al. [50] referred to accuracy as a criterion that measured the model's discriminating ability. On the other hand, binary classification techniques utilise AUC procedures to determine the models with optimal class prediction.

AUC are effective when estimating the performance of models without pre-existing error cost information [37]. However, AUC presumes different cost distributions between other classification techniques regarding the distribution of their actual score, thus presenting effective performance comparisons. Hand [37] formulated and proposed the H-measure to measure the performance of classifiers instead of the AUC technique. The H-measure assumes a varied distribution of costs between classifies and does not rely on their scores. Overall, the H-measure computes a single threshold distribution that fits all classifiers.

Attempts to perform an adherence hypothesis test on distribution data lead to the conceptualisation of the KS distribution. The strategy has been utilised in binary classification as a dissimilarity metric for the classifier's discriminant power assessment. The KS distribution measures the distance produced by the classifier's score. The distance separates the two data classes' cumulative distribution functions [3].

According to Brier [24], the Brier score is computed as the mean-squared error. It helps measure the accuracy of the classifier's probability predictions based on the mean squared error. The Brier score attempts to compute a mistake's average quadratic possibility. Whereas the Brier score directly incorporates the probabilities in the computations, accuracy utilises a predetermined cut-off score or threshold to transform the probabilities into binary outcomes. As a result, a lower Brier score is associated with superior classifier performance.

To check whether a model's behavioural score can be considered as the probability of missed payment, calibration curves are considered. Also known as reliability diagrams, they can be applied to classifiers that predict not only the class label but also obtain a probability of the respective label. Reliability diagrams provide a diagnostic to check whether the scores are reliable. Thus, a prediction is considered secure if the event happens with an observed relative frequency consistent with the forecast value [25].

A calibration curve works by sorting the output scores of the classifier. Specifically, the forecasts are divided into a fixed number of buckets along the x-axis. The number of events (class or label = 1) is then counted for each bin (e.g. the relative observed frequency). Finally, the counts are normalised. The results are then plotted as a line plot. If the classifier is forecasting accurately, then it is expected that the percentage of dominant class classifications and the mean probabilities assigned to the dominant classes in each bin to be close to one another. If it is not doing so accurately, then these two values must diverge. The point positions on the curve relative to the diagonal help to interpret the forecasts, for example:

- Below the diagonal: The model has an over-forecast; the probabilities are too large.
- Above the diagonal: The model has an under-forecast; the probabilities are too small.

4.6 Statistical significance tests

Models have unique splitting techniques and performance measures that make it impractical to prove whether a model performs better than another [73]. Implementing innovative hypothesis testing strategies to highlight the existence or inexistence of experimental differences may be appropriate for the comprehensive model performance evaluation. Hypothesis testing enables one to assess whether statistically significant differences in the performance of different models exist and ensures that accuracy is not only influenced by random splitting effects. It is crucial to select the right performance assessment test for other prediction techniques based on the number of classifiers and the nature of inputs.

Researchers may employ parametric statistical tests, such as the paired *t*-test and the nonparametric tests like Wilcoxon and McNemar test to assess model performance [28]. However, the non-parametric tests are more effective because parametric tests have proved to be statistically unsafe and conceptually unsafe in most cases. Nonparametric tests are considered safer and appropriate because they neither assume homogeneity of variance and normality of data [28]. The McNemar test was utilised to help

evaluate the performance of models fitted using a unique dataset [29].

The McNemar test is an effective strategy for examining the existence or inexistence of statistically significant differences in classifier performances [45]. The model performance metric is based on the Chi-square ($\chi 2$) test. The goodness of fit test compares the distribution of the null hypothesis' expected counts to the observed counts. This test is practically applicable to a 2×2 contingency table. The table's cells include the correctly and incorrectly classified number of cases by both models. Besides, the cells show the number of correctly classified samples from only one model.

The objective of McNemar test is to check the null hypothesis, which says that neither of the two models performs better than the other. The alternative hypothesis states that the performance of the two models is not equal.

The McNemar statistic is as follows:

$$\chi^2 = \frac{\left(\left|n_{ij} - n_{ji}\right| - 1\right)^2}{n_{ij} - n_{ji}} \tag{11}$$

where n_{ij} indicates the number of cases misclassified by model i but classified correctly by model j, and n_{ji} Indicates the number of cases misclassified by model j but not by model i.

The computed statistic is considered as a value from the χ^2 distribution with 1 degree of freedom. Based on this assumption, the p-value is calculated. If this p-value is smaller than the predefined significance level α, then we fail to reject the null hypothesis. Otherwise, we reject the null hypothesis and accept the alternative hypothesis. For example, if the value of the test statistic is more significant than 3.84, then (according to the χ^2 table at 95% confidence interval) it can be stated that the two methods differ in their performances. In other words, the difference in performance between the methods i and j is said to be statistically significant.

5 Experimental results and discussion

In this section, the results of the proposed LSTM models are presented along with comparisons to the benchmark classifiers. The model is validated over the above-described real-world credit card datasets across five performance measure metrics. In addition, several tables and figures regarding the proposed model results and comparison to traditional models are provided and discussed. All the experiments for this study were performed using Python 3.8×64 on a PC with an AMD 8-core Ryzen^{TM} 7 3700X 3.6–4.4 GHz processor and 32 GB RAM, running Microsoft Windows 10 operating system.

To outline the discrimination power of the MP-LSTM model, performance measures are calculated not only for all active customers but for different subsets of them:

- Customers with at least one missed payment during the last two months are the group that generally have a low risk of default, but the recent missed payment is a reason to look at those in this group more closely.
- Customers with a missed payment during the last month is a subset of the first group. Whilst one missed payment can be made by chance, here there is a need to look at this group to distinguish riskier customers from other ones.
- Customers with two consecutive missed payments form a group in which most customers might have financial problems because it is unlikely to forget to pay for more than one month.
- Customers with three consecutive missed payments are those on the verge of default. For this group, a fourth missed payment is equivalent to default, so an MP-LSTM prediction of the fourth missed payment is a prediction of default.

As a next step, the MP-LSTM models are compared with four classical classifiers: MLP neural network, logistic regression, SVM, and random forest. Comparisons were made not only using the performance measures but also using the statistical McNemar test all done on the transactional and non-transactional dataset. As a final step, for the transactional dataset PE-LSTM for monthly purchase estimation and group customers into five groups based on the expected loss in case of default is introduced.

5.1 MP-LSTM model results

To prove that the results obtained on the testing set are worthy and to make the results of MP-LSTM significant, different measures need to be evaluated, each of which reflects various aspects of the model performance:

- Accuracy is the simplest method of evaluating the model preciseness. It does not consider any misclassification loss and simply displays the proportion of correctly classified missed payments for the default score threshold, equal to 0.5.
- Sensitivity and specificity analysis give us true model discrimination power for positive and negative classes. It is beneficial when the dataset is imbalanced, as in our case (The ratio of clients with missed payments is usually low). To determine how well sensitivity and specificity values we will use the F1 score, which is calculated using the following equation:

$$F1 = 2 * \frac{\text{sensitivity} * \text{specificity}}{\text{sensitivity} + \text{specificity}} \quad (12)$$

- AUC tells us how the model will perform for different selected thresholds.
- Brier score reflects the model's discriminatory power (i.e. how specific the model is about the customer's predicted missed payment).
- KS reflects the maximum difference between the fraction of correctly classified customers, those who missed a payment and incorrectly classified customers, those who did not miss a payment. The value tells us that the model correctly classifies not only the presence of a missed payment but also the absence of it.
- H-measure is an integral measure over all misclassification costs. A high H-measure value tells us that, regardless of the actual cost of misclassification, the total loss cost of the model is low.

5.1.1 Transactional dataset results

Table 3 shows the performance indicator measures for the proposed model on different input data. The second row shows the number of clients in every data set, while the next one represents the number of customers with a missed payment during the testing month. The fourth row provides the proportion of customers with a missed payment for all customers in the group. When considering the entire dataset, the general percentage of missed payments isn't high, but it grows proportionally with the number of missed payments in previous months. Thus, it can be concluded that clients who already have problems with paying will, with very high probability, have them in the future too. So, the proposed model should define those customers as unreliable.

The correctness of the MP-LSTM prediction ability is shown in the "Accuracy" column. The tendency of accuracy to grow according to the increase in missed payment numbers should be noted. This means that the model considers consumers with payment problems as those who are more prone to them in the future. Thus, this makes the classification of clients with many missed payments easier. Also, the classifier accuracy is very high in general.

As mentioned earlier, the higher the AUC value, the better the classifier can distinguish between classes. The proposed model shows almost the same prediction ability on all subsets of active customers. In general, it is higher than 85%, which proves good classifier separability. The lower the Brier score is, the better classifier performs. Also, a decrease can be seen in the Brier score along the column from consumers with the lowest number of missed payment fee to those with the highest ones, a tendency like that of the accuracy column. The higher the Kolmogorov–Smirnov statistics, the better the discriminative power of the model. The tendency of its decreasing value along the column can be seen. But for the entire set, this value is sufficiently high to prove good discriminative model ability.

As was mentioned before, the H-measure is an indication of the misclassification loss, and this depends on the relative proportion of objects belonging to each class. The influence of the different numbers of customers with missed payment fees can be seen from the table. But generally, the higher H-measure, the better the classifier is in terms of performance over different misclassification costs. For all subsets of customers that are investigated, this value is good enough.

Based on the results provided in Table 3 we can see that if the customer have several consecutive missed payments, it is very likely that we will miss another one. Thus, for such groups of clients LSTM model almost always correctly predicts the next missed payment (sensitivity

Table 3 Performance measures for LSTM classifier for the transactional dataset

Description	Total	Missed Payments	Proportion	Accuracy (%)	Sensitivity (%)	Specificity (%)	AUC	Brier Score	KS	H-Measure
All customers	10.315	2484	24.08%	90.62	72.87	96.25	0.91	7.62%	0.71	0.63
Customers with at least one missed payment during the last two months	2447	1910	78.05%	84.96	90.58	64.99	0.89	10.85%	0.66	0.44
Customers with missed payment during last month	1925	1621	84.21%	88.78	94.69	57.24	0.9	8.5%	0.65	0.45
Customers with two consecutive missed payments	1325	1257	94.87%	94.79	98.33	29.41	0.86	4.45%	0.56	0.25
Customers with three consecutive missed payments	1026	989	96.39%	96.49	99.09	27.03	0.87	3.12%	0.59	0.26

column). On the other hand, it is hard for it to find among these "bad customers" ones that will eventually pay (specificity column).

As it can be seen, the AUC value for the MP-LSTM model is high for all customers and specific risk groups. However, the proportion of missed payments for all customers and customers with missed payments differs significantly (see Table 3). The shape of the ROC curve is round for all customer groups. The most important for us is the group with three consecutive missed payments, and from Fig. 7, the AUC for this group is 0.87, indicating good discrimination power for most thresholds.

Fig. 8 represents the behaviour score distribution for different customer groups along with the observed behaviour. The splitting process into ten buckets along the x-axis was based on the customer missed payment prediction. Thus, it is expected that the number of customers without a missed payment will decrease along the axis. In contrast, the number of customers with missed payments will increase, and the histogram reflects this tendency. So, whilst there are, of course, misclassified clients in every group, their percentage is significantly less than correctly classified. Thus, the proposed model can be considered as reliable.

Figure 8 compares how well the probabilistic predictions of MP-LSTM for the different client groups are calibrated using ten bins. The calibration curve for all clients shows that it is the best calibrated among the others. It has a small over-forecast for the first seven bins, while the remaining plot represents small under-forecasts. But in general, the curve is very close to the diagonal, especially on the last two bins, where they match nearly completely. It

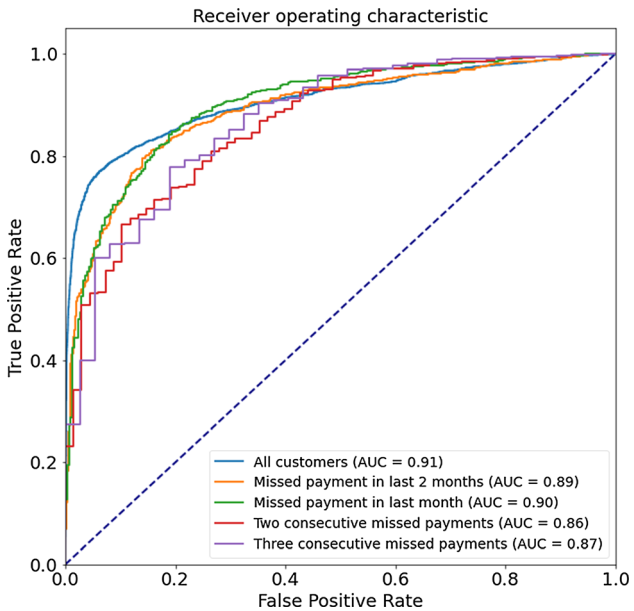

Fig. 7 ROC curves and AUC values for different customer groups

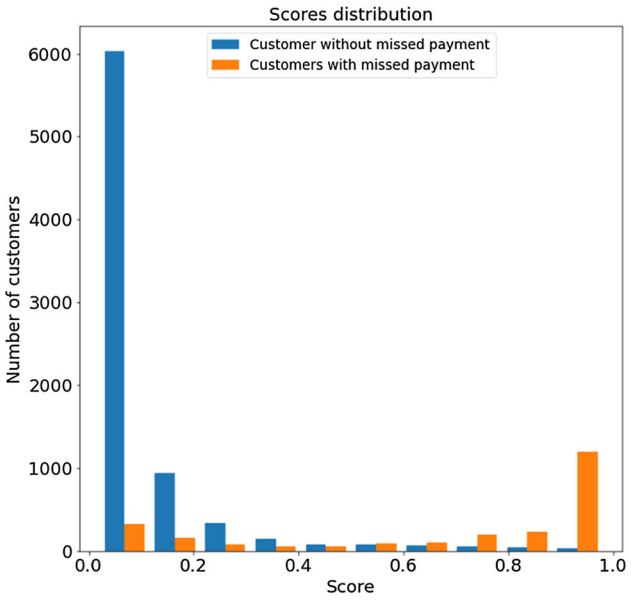

Fig. 8 Distribution of MP-LSTM scores for different customer groups

means that the output of the proposed model for all clients can be used as the customer default probability without additional calibration.

The curves for the customers with one missed payment at different time intervals are very similar. They both have under-forecasts along most of their length, except for one bin with an over-forecast. They are close to the diagonal but not as close as the curve for the entire customer dataset. Both calibration curves for clients with more than one missed payment have a blank first bin. This means that the classifier becomes more discriminative on their data and proves our suggestion that the model considers such clients as those prone to have payment problems and usually gives them a high probability of default. Also, it is noticeable the huge under-forecasts along their length, except one over-forecast for the three consecutive defaults curve (Fig. 9).

5.1.2 Non-transactional dataset results

The same modelling framework has been used to model the non-transactional dataset. Tables 3 and 4 show the performance indicator measures for the proposed model on different input data.

The correctness of the MP-LSTM prediction ability is shown in the "Accuracy" column. Performance measures for the customers with three or more consecutive missed payments are much lower than for other groups. It could be explained by the fact that some proportion of customers drastically change their behaviour in the risk of bankruptcy and trial. So, based on its past behaviour, they should have fourth missed payment, but pressure from the bank forces them to pay. The table shows that the model considers

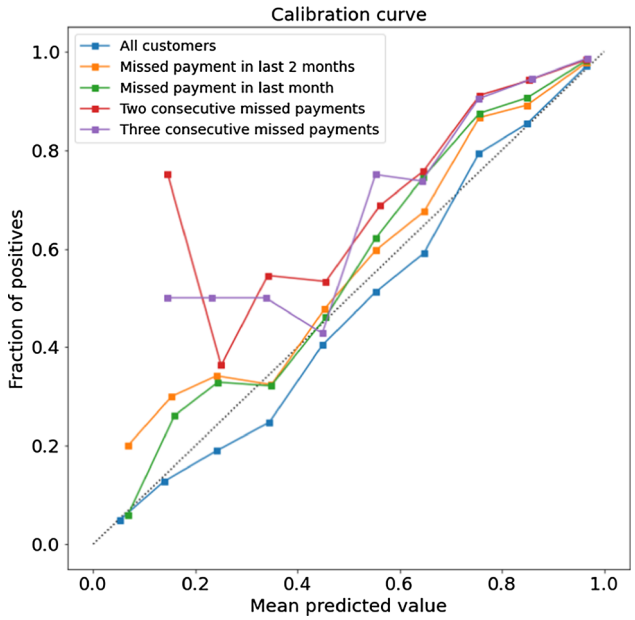

Fig. 9 Calibration curves for different customer groups

consumers with payment problems as those who are more prone to them in the future. The classifier accuracy is lower than for the transactional dataset, which can be explained by initial data pre-processing, which might lead to information loss.

As mentioned earlier, the higher the AUC value, the better the classifier can distinguish between classes. The proposed model shows a similar prediction ability on all subsets of active customers except the last one. For those except the last, it is higher than 77%, which proves good classifier separability. The lower the Brier score is, the better classifier performs. An increase can be seen in the Brier score for the customers with three missed Payments. The higher the Kolmogorov–Smirnov chart statistics, the better the discriminative power of the model. As was mentioned before, for all subsets except the last one, this value is sufficiently high to prove good discriminative model ability.

As was mentioned before, the H-measure is a measure of the misclassification loss, and this depends on the relative proportion of objects belonging to each class. The influence of the different numbers of customers with missed payment fees can be seen from the table. But generally, the higher H-measure, the better the classifier is in terms of performance over different misclassification costs. For all subsets of customers that are investigated, this value is good enough.

As it can be seen, the AUC value for the MP-LSTM model is high for all customers as well as for specific risk groups except the last one (with three consecutive missed payments), even though the proportion of missed payments for all customers and customers with missed payment differs significantly (see Table 4). The shape of the ROC curve is round for all customer groups. The highest AUC is for the customers with two consecutive missed payments. Bank can use this group to early put pressure on such customers and prevent a third missed payment.

Fig. 10 represents the behaviour score distribution for different customer groups along with the observed behaviour. In Fig. 11 the splitting process into ten buckets along the x-axis was based on the customer missed payment prediction. Thus, it is expected that the number of customers without a missed payment will decrease along the axis. In contrast, the number of customers with missed payments will increase, and the histogram reflects this tendency. So, whilst there are, of course, misclassified clients in every group, their percentage is significantly less than correctly classified. Thus, the proposed model can be considered as reliable.

To elaborate more Fig. 11 compares how well the probabilistic predictions of MP-LSTM for the different client groups are calibrated using ten bins. The calibration

Table 4 Performance measures for LSTM classifier for the non-transactional dataset

Description	Total	Missed Payments	Proportion	Accuracy (%)	Sensitivity (%)	Specificity (%)	AUC	Brier Score	KS	H-Measure
All customers	30.000	6636	22.12%	82.03	37.43	94.69	0.78	0.1353	0.42	0.28
Customers with at least one missed payment during the last two months	15265	3997	26.18%	79.27	39.88	93.24	0.77	0.152	0.41	0.27
Customers with missed payment during last month	13714	3567	26.01%	79.5	39.61	93.53	0.77	0.1511	0.41	0.27
Customers with two consecutive missed payments	7974	2592	32.51%	76.81	51.31	89.09	0.79	0.1654	0.44	0.3
Customers with three or more consecutive missed payments	313	210	67.09%	71.88	91.43	32.04	0.66	0.2004	0.28	0.14

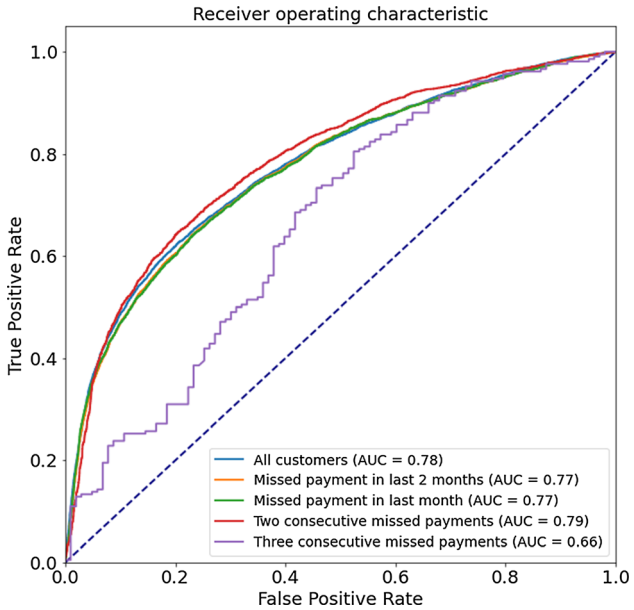

Fig. 10 ROC curves and AUC values for different customer groups

curve for all clients shows that it is the best calibrated among the others. It fits the line almost perfectly, which means that missed payment scores can be considered as probabilities. The only group with the curve far from the central line is customers with three or more consecutive missed payments. This curve has a small over-forecast for low scores, but in general, it also lies close enough to other curves (Fig. 12).

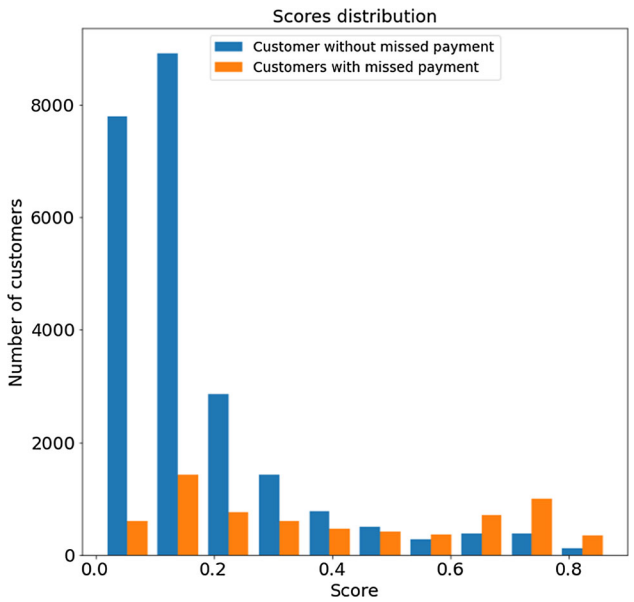

Fig. 11 Distribution of MP-LSTM scores for different customer groups

5.2 Benchmark model results and comparison

To verify the strength and discriminative power of the MP-LSTM model on the transactional and non-transactional data, their performance is compared with four traditional classifiers: MLP, SVM, RF and LogR.

5.2.1 Transactional dataset results vs benchmark classifiers

For all these classifiers, the transaction data needed to be converted to extract a fixed number of features for each customer. Based on an expert opinion, these features were selected:

- The difference in time between the last and first transaction of a client (including all transaction codes)
- The difference in time between the last and first purchase of a client
- The difference in time between the last and foremost activity of a client
- Count of purchases, cash machine withdrawals, and other transactions (3 features).
- Amount of purchases, cash machine withdrawals, and other transactions (3 features).
- Months of customer loyalty.
- The number of loyal months.
- Count of missed payments.
- Sum of missed payment fee transactions.
- Count of missed payment fees that were made during loyal months.
- Presence of bank account.
- Max number of consecutive missed payment fees.

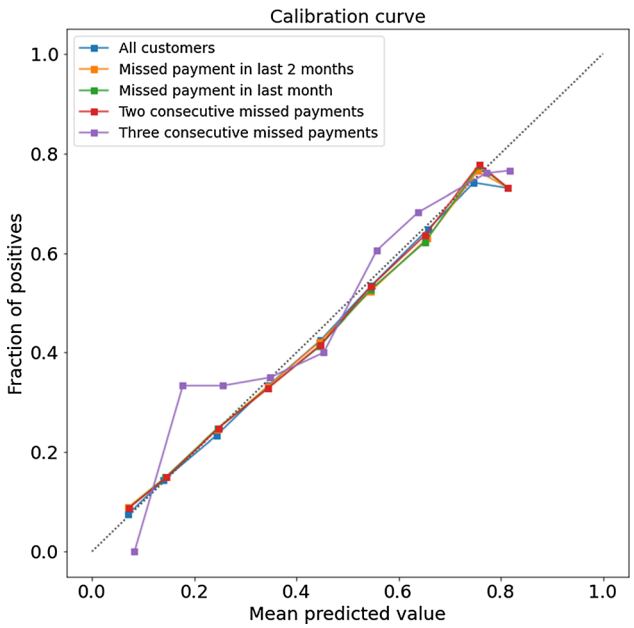

Fig. 12 Calibration curves for different customer groups

- Count of consecutive missed payment fees.
- Count of missed payments that were made between the last and first purchase of a client
- Count of missed payments that were made after the last purchase of a client
- Integral of missed payment fee frequency, calculated by the equation:

$$\text{Missed payment frequency} = \sum_{i=1}^{T} (0.95)^{T-i} l(i) \qquad (13)$$

where T is the final month of training data and

$$l(i) = \begin{cases} 0, & \text{no missed payment in month } i \\ 1, & \text{missed payment in month } i \end{cases} \qquad (14)$$

These features were selected by experts from in banking and financial industry, and they cover almost all aspects of client behaviour. The goal of this research is about developing LSTM models and not about extracting features.

The results show that despite the exhaustive list of extracted features, the traditional classifiers have worse performance than the MP-LSTM model.

Table 5 represents the performance indicator measures for the different classifiers on the same input data. At first sight, the correctness of the predictions is similar and high enough for all the models. However, achieving even a 1% or 2% increase in an already high accuracy can sometimes be a challenging task. As it is evident, MP-LSTM differs from others at least at 1.5%, which can be considered as a good result.

Threshold changes can improve classifier accuracy; maximum accuracy can be achieved by applying the optimal threshold. So, as it can be seen, there is a slight increase for all of them when using the optimal threshold, but the highest value still belongs to the proposed model. As has already been mentioned, the higher the KS value, the better the classifier performs. The last three classifiers have almost the same KS value. We are applying the MLP neural network results in a lower score. The highest score belongs to MP-LSTM again. This proves the advantage of the proposed model over the traditional classification tools.

H-measure is ultimately the same for all classifiers but MP-LSTM, which has a higher value, proving that it performs much better. Thus, the table demonstrates that the proposed model gives more accurate predictions than other classifiers according to all performance indicator measures. So, it can be concluded that it was constructed correctly. As it is clear from Fig. 13, the performance of MP-LSTM is superior compared to that of the other classifiers. The worst AUC value is from the SVM classifier (especially in the second part of the plot), which means it is not acceptable to use it to increase the True Positive Rate value.

MP-LSTM is a superior model in predicting missed payments (high sensitivity). On the other hand, the specificity of the proposed model is also relatively high, and overall sensitivity–specificity values are balanced (the F1 score of the MP-LSTM model is much higher than for other models).

Figure 14 clearly shows how well the probabilistic predictions of the different classifiers are calibrated, using a calibration curve with ten bins. The plot shows that there are two perfectly calibrated classifiers: MP-LSTM and logistic regression. A small over-forecasts and under-forecasts along the MP-LSTM length can be noticed, but the curve is still close to the diagonal, especially for the last two bins, where they match entirely. This means that the output of the proposed model can be considered as the probability of consumer default without additional calibration or improvements in the MP-LSTM structure.

Logistic regression is known to produce well-calibrated probabilities by default as it directly optimizes log-loss, which is just a convenient restatement of class probability. In other words, probability figures directly into the cost function that logistic regression solves for, and hence the algorithm produces unbiased probability estimates. The MLP calibration curve shows that the classifier under-forecasts Consumer missed payment probabilities for most of the bins. Thus, its prediction ability is not good enough.

Table 5 Comparison of performance measures for all classifiers for the transactional dataset

Classifier	Accuracy (%)	Sensitivity (%)	Specificity (%)	F1 score	AUC	Brier score	KS	H-measure	Max accuracy (%)	Optimal threshold
MLP NN	89.11	55.27	95.87	70.12	0.87	0.0836	0.63	0.51	89.17	0.53
SVM	88.04	39.42	97.74	56.18	0.84	0.0939	0.63	0.51	89.46	0.19
Random Forest	88.15	43.24	97.11	59.84	0.88	0.0931	0.64	0.51	88.76	0.26
Logistic Regression	88.01	42.56	97.08	59.18	0.87	0.0902	0.65	0.51	88.68	0.37
MP-LSTM	90.62	72.87	96.25	82.94	0.91	0.0762	0.71	0.63	90.69	0.52

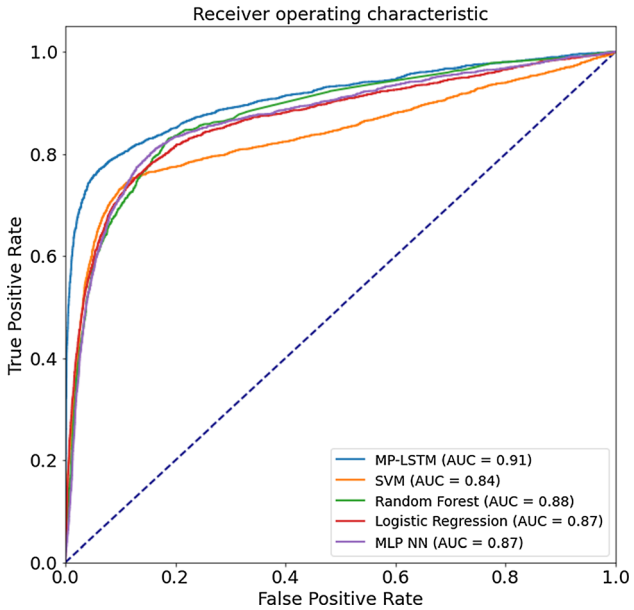

Fig. 13 ROC curves and AUC values for all classifiers in the transactional dataset

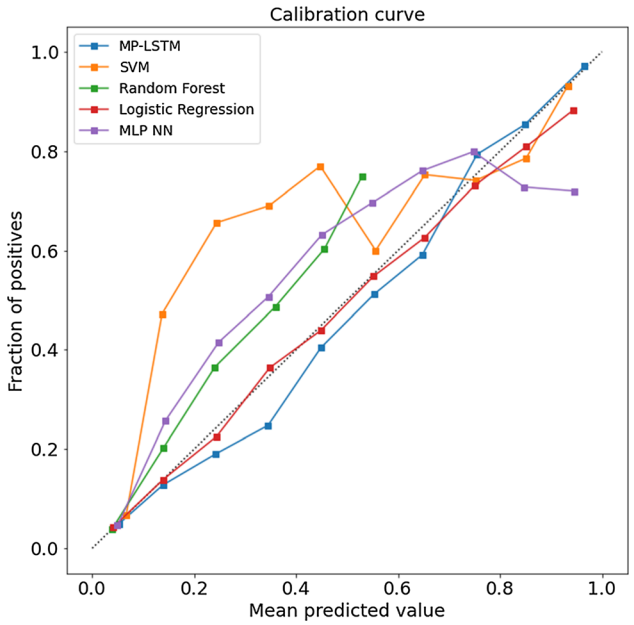

Fig. 14 Calibration curves for all classifiers in the transactional dataset

This may be caused by the small number of neurons in hidden layers (Fig. 14).

When comping SVM, it is worth mentioning that this classifier is among those which do not naturally evaluate class probabilities. Thus, the scores must not be interpreted as such. That is why the usual calibration curve for SVM looks like a sigmoid function. A huge under-forecast is easily seen for the first half of the bins. But for the second half, the calibration curve is closer to the diagonal. Thus, scores obtained from this model cannot be interpreted as probabilities. The random forest calibration curve also shows under-forecast for the first six bins. It is also worth mentioning that this is the only classifier with empty bins, which means that even if the consumer has missed a payment, the random forest classifier will not assign a probability greater than 60%. That is why, by this measure, random forest performs the worst.

To make sure that the difference in performance measures is statistically significant and are not caused by chance, the McNemar test is used. The table represents the results of applying the McNemar test for pairwise comparison of the LSTM model and the other classifiers. To interpret testing results correctly, the significance threshold $\alpha = 0.05$ must be previously defined. According to the results above, every classifier pair shows a statistically significant performance difference. Moreover, the previously mentioned performance indicator measures prove that all the traditional classifiers show worse prediction ability than the LSTM model (Table 6).

5.2.2 Non-transactional dataset results vs benchmark classifiers

The non-transactional dataset comparison results are shown in Table 7, which represents the performance indicator measures for the different classifiers on the same input data.

At first sight, the correctness of the predictions is similar and high enough for all of the models. The closest performance model to MP-LSTM is the MLP model. The performance of all classifiers is so relative to each other lies in the lower dimensionality of the input data. For each customer, there are only 23 features instead of hundreds of transactions in the previous data set. So, simple classifiers have fewer problems in extracting useful information from feature space. Threshold changes can improve classifier accuracy; maximum accuracy can be achieved by applying the optimal threshold. So, as it can be seen, there is a slight

Table 6 McNemar test for MP-LSTM pairwise comparison with other classifiers

	p-value	Statistic
MLP NN	2.7×10^{-26}	112.6
SVM	9.9×10^{-64}	284.0
Random forest	1.6×10^{-47}	209.7
Logistic regression	1.7×10^{-56}	250.8

Table 7 Comparison of performance measures for all classifiers for the non-transactional dataset

Classifier	Accuracy (%)	Sensitivity (%)	Specificity (%)	FF1 score	AUC	Brier score	KS	H-measure	Max accuracy (%)	Optimal threshold
SVM	81.46	28.56	96.49	44.07	0.7	0.1432	0.37	0.23	81.89	0.4
Random Forest	80.18	17.01	98.12	28.99	0.77	0.1438	0.41	0.27	81.56	0.32
MLP NN	81.83	37.07	94.48	53.25	0.77	0.136	0.42	0.28	81.9	0.52
Logistic Regression	80.88	22.56	97.44	36.64	0.72	0.1455	0.37	0.24	81.79	0.41
MP-LSTM	82.03	37.43	94.69	53.65	0.78	0.1353	0.42	0.28	82.09	0.53

increase for all of them when using the optimal threshold, but the highest value still belongs to the proposed model.

MP-LSTM and MLP-NN have the same KS value, slightly higher than the corresponding value for other classifiers. A similar pattern can be observed with H-measure. Brier score for the MP-LSTM model is the lowest, which proves the quality of this model. The MP-LSTM validated on the non-transactional data set has achieved results that outperformed [31], where they used the same dataset in measures such as AUC by 11% and H-measure +0.05. On the other hand, the accuracy is very slightly lower than their study by 0.8%.

MP-LSTM, together with MLP NN, have the most balanced sensitivity–specificity values, and the F1 score of these classifiers confirm this fact. Random Forest and Logistic Regression, on the other hand, quite effectively predict good client behaviour but fail in predicting missed payments.

To provide more rigorous analysis, we also provide standard deviation of the most critical performance metrics of all classifiers overall 5-folds.

As it can be seen from Table 8, the MP-LSTM classifier has the lowest standard deviation for accuracy and brier score and second-lowest value for AUC. It seems that we can rely on this algorithm more than on others.

As it can be seen from Fig. 15, the performance of MP-LSTM is slightly better. The worst AUC value is from the SVM classifier (especially in the second part of the plot), which means that it is acceptable to increase the True Positive Rate value. Similar unsatisfactory behaviour of the SVM classifier can be noted for the transactional dataset.

Figure 16 compares how well the probabilistic predictions of the different classifiers are calibrated, using a calibration curve with ten bins. The plot shows that there are two perfectly calibrated classifiers: MP-LSTM and logistic regression. The curve of the MP-LSTM classifier is even closer to the diagonal than the curve of logistic regression. Moreover, it is closer than the curve of the MP-LSTM classifier for the transactional dataset. That is why scores of this classifier can be used as probabilities. The worst curves have MLP neural network and random forest classifiers (similar behaviour to their performance for the transactional data set). To make sure that the difference in performance measures is statistically significant and are not caused by chance, the McNemar test is used.

Table 9 represents the results of applying the McNemar test for pairwise comparison of the LSTM model and the other classifiers. During the application of the McNemar test, the same value of significance threshold is used for the transactional data set $\alpha = 0.05$. According to the results above, every classifier pair shows a statistically significant performance difference. The closest MP-LSTM classifier is MLP NN, which has a p-value equal to 4.5%. Current results of the McNemar test combined with the previously

Table 8 Comparison of performance measures standard deviation

Classifier	Accuracy STD (%)	AUC STD (%)	Brier score STD
SVM	0.46	1.03	0.37
Random forest	0.47	0.47	0.38
MLP NN	0.46	0.42	0.21
Logistic regression	0.41	0.71	0.29
MP-LSTM	**0.37**	**0.45**	**0.17**

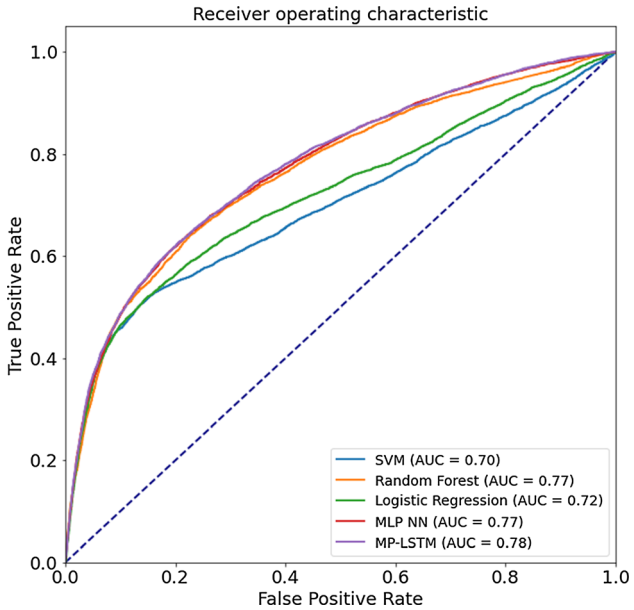

Fig. 15 ROC curves and AUC values for all classifiers in the non-transactional dataset

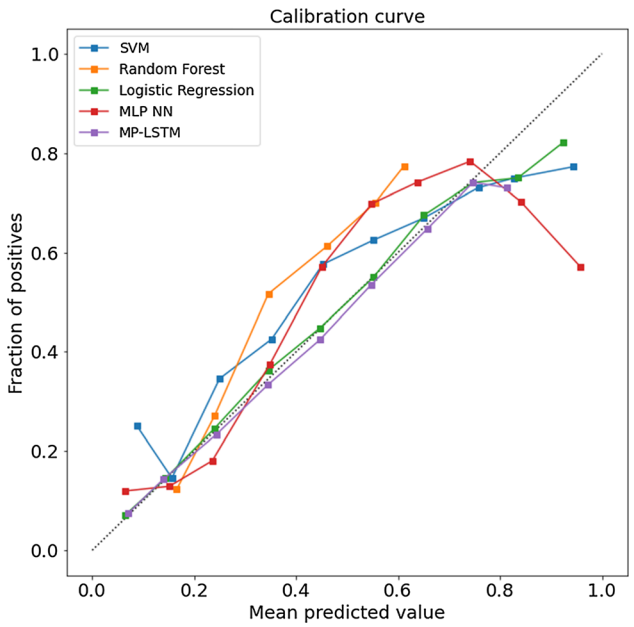

Fig. 16 Calibration curves for all classifiers in the non-transactional dataset

mentioned performance indicator measures prove that all the traditional classifiers show worse prediction ability than the LSTM model.

Several studies in the literature have proposed new approaches for credit and behavioural scoring and validated them on the non-transactional Taiwan credit dataset [7, 31, 44]. Hence, it is feasible to benchmark the reported MP-LSTM results of the proposed model to evaluate its feasibility and efficiency across different performance measures. Table 10 summarises a comparison of MP-LSTM approach results against those mentioned above in recent related studies in the literature.

Table 9 McNemar test for MP-LSTM pairwise comparison with other classifiers

	p-value	Statistic
MLP NN	0.045	4
SVM	2.7×10^{-7}	26.4
Random forest	1.9×10^{-32}	141
Logistic regression	7.5×10^{-17}	69.5

Although it's not the paper's main objective, we also provided model sensitivity analysis to determine which features are the most important. After removing non-temporal features, model accuracy dropped by 1.65% in the transactional dataset and 1.03% for the non-transactional dataset. Therefore, we can conclude that primary information about future missed payments is stored in the past transactional behaviour of the client and not his demographic info. However, after removing demographic features, we can see that some valuable information is missed.

It is evident from the comparison that the proposed MP-LSTM approach has shown a remarkable result on AUC measure when compared with Jadhav et al. [44] and Feng et al. [31], where the difference reached on average 9.5% compared to their best-proposed approaches, this indicates the capability of the proposed model in distinguishing between dataset classes. Also, the H-measure and Brier Score of the MP-LSTM model outperformed that in Feng et al. [31] and Ala'raj et al. [7], respectively. LSTM accuracy achieved 1.47% lower than that in Ala'raj et al. [7]. Overall, the MP-LSTM approach results on Taiwan non-transactional dataset showed better results against studies compared.

5.3 PE-LSTM model and customer grouping

For banks management, it is important not only to have probabilities of missed payments but also to group customers to have a corrective action for each group. Based on the transactional dataset introduce, five groups were introduced: "very low", "low", "medium", "medium/high", "high". A grouping is proposed based not only on the probability of missed payment but also on the estimated monthly purchase amount. The reason for this is to predict the bank's loss amounts. Therefore, customers are grouped

Table 10 Comparison of the proposed approach (MP-LSTM) results with recent approaches against Taiwan non-transactional credit card dataset

Studies	Proposed approach	Taiwan credit dataset			
		Accuracy (%)	AUC	H-measure	Brier Score
Feng et al. [31]	DECSP	83.1	0.67	0.23	–
Jadhav et al. [44]	IGDFS + SVM	82.5733	0.60627	–	–
	GAW + SVM	81.2097	–*	–	–
	IGDFS + KNN	81.1733	0.62703	–	–
	GAW + KNN	80.9833	–	–	–
	IGDF + NB	81.98	0.69861	–	–
	GAW + NB	82.0267	–	–	–
[7]	DES-LA	**0.835**	0.769	–	0.159
This paper	MP-LSTM	82.03	**0.78**	**0.28**	**0.1353**

**Denotes to non-investigated in their study*

based on the mathematical expectation of their loss. For each customer i, the value of expected loss as a product of missed payment probability and desired monthly purchase amount is calculated using Eq. (14).

$$\text{Loss}_i = \text{Missed payment probability}_i * \text{Monthly purchase amount}_i \quad (15)$$

Here, all customers are placed in the hypothetical situation where they have three consecutive missed payment fees, and the next one will be a potential default. Only one additional grouping rule is made: all clients with three consecutive missed payment fees in recent months are categorised into the "high" risk group.

To obtain the monthly purchase amount for consumers, the new PE-LSTM model was designed and trained. The structure of the model is presented in Fig. 4. As it can be seen from Table 2, the architecture of the model is more complex compared to the MP-LSTM model because the regression task is more complex than the classification one. After training the PE-LSTM model, it is tested to estimate the purchase amount during the last month of the available data. The MAE measure of the PE-LSTM classifier is equal to 2623, while the standard deviation of residuals for the PE-LSTM classifier is equal to 9070. The standard deviation number is high because customers spend a lot of money while purchasing.

Figure 17 represents the histogram of actual and predicted monthly purchases for active customers. The higher the actual purchase is, the smaller its group becomes, except for the last one. The model's predictions show the same tendency. Also, the difference between the actual and predicted values for most groups is slight. The closest match is observed for the first group (customers whose purchases are lower than 1000). Thus, it can be assumed that it is easier for the model to estimate the purchase rate of customers who do not spend a lot.

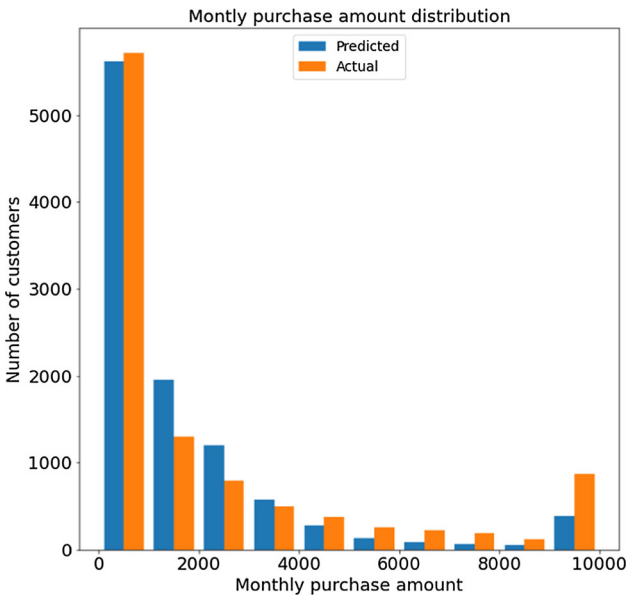

Fig. 17 Distribution of actual and predicted monthly purchases

The distribution of clients according to the residual value, defined as the difference in their actual and predicted purchases, is shown in Fig. 18. A tendency can be seen here that the farther from the 0-difference value, the smaller the group becomes, except for the last and first bins. Moreover, the largest groups are around the 0 value, which means that most customers' purchases are predicted accurately. Also, the number of clients with a significant negative difference is relatively smaller than in the groups with positive residuals. Based on the figure, it can be concluded that, in general, residuals are normally distributed with a small number of outliers. The reason for the right outlier may be due to unplanned spending by customers.

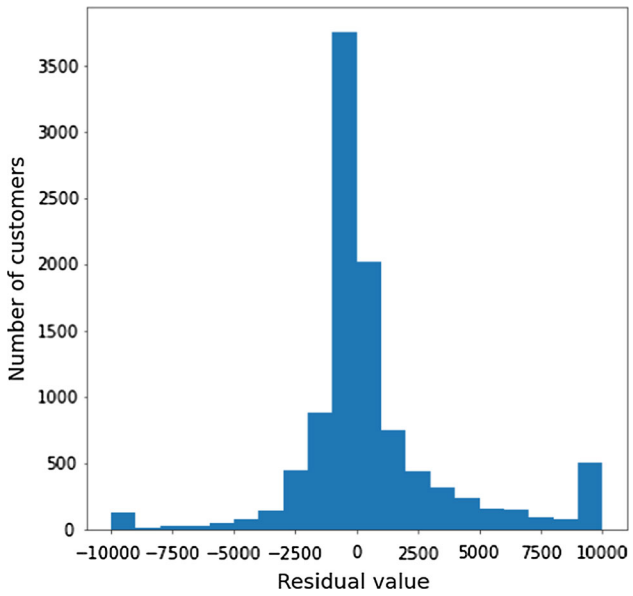

Fig. 18 Residual plot

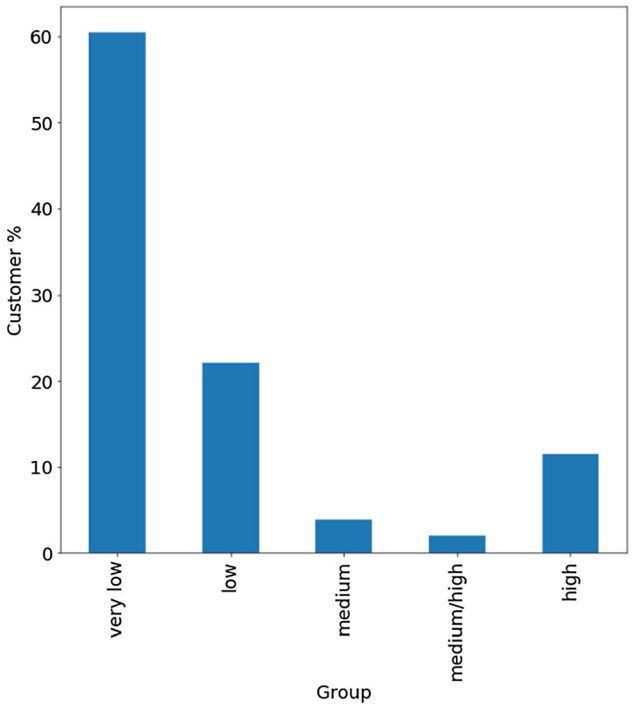

Fig. 19 Grouping clients by risk group

Looking at Fig. 19, it is apparent that most clients are in the "very low" and "low" risk groups. On the other side, about 15% of clients are in the "high" group and will almost certainly be in default during the testing month. The obtained groupings can be considered as the final output of both models. One use for these groupings can be to create different loyalty policies for different customers to encourage them to spend more and pay on time.

6 Conclusion

In this paper, two models were presented; namely, MP-LSTM, which was validated on transactional and non-transactional datasets, and PE-LSTM, validated on the same transactional dataset, which can be beneficial tools for banks. Nowadays, modern technology allows providing a vast diversity of products and services to customers. So, it is imperative to acquire new customers and maintain those who are creditworthy effectively. Without proper credit risk management, banks can face tremendous losses, hence the importance of having a tool that can help classify clients according to their credit card default probability. With the increase in transactional data volumes, classical machine learning approaches cannot efficiently use all available data, which leads to bad scoring quality. To make use of classical algorithms, feature extraction need to be performed. The proposed model was based on the LSTM neural network architecture, which has such benefits:

(1) It allows for avoiding the problem of manually extracting features from the transactions data set.
(2) It allows using not only transactional data but also demographics data as well.
(3) To train the model, transaction sequences of various lengths can be used (some customers made thousands of transactions, while others made only a few)
(4) Training the model on customer credit card behaviour in previous months gives a highly accurate prediction of default probability and purchase amount for every customer.

To prove the effectiveness of using the proposed models, it was compared to other traditional classification models: MLP, SVM, RF and LogR. The following performance measures were used for the comparison, specifically: accuracy, AUC, H-measure, Kolmogorov–Smirnov test, Brier score, calibration curves, and the McNemar test. All measures prove outperformance by the MP-LSTM model. Therefore, it can be concluded that MP-LSTM performs statistically better than other classifiers. Its calibration curve shows that the output of the model can be considered as the probability of default without any additional improvements. This fact also implies that the model can be used to predict missed payment $\boldsymbol{n}$ month ahead using the following equation

$$p_{\mathrm{n}} = 1 - (1 - p_1)^{\mathrm{n}} \tag{16}$$

where p_1 is a probability of missed payment one month ahead; p_{n} is the required value of $\boldsymbol{n}$ month ahead probability of missed payment. The equation is based on the basics of probability theory.

When talking about purchase amount prediction, it has been shown that most customer predictions can be used for

"reliability" analysis. In addition, the combined predictions of both LSTM-based models were used to determine the predicted loss amount due to missed payments and split customers into groups based on that risk.

Constructing a behaviour score for customers is novel because it is based on estimated bank losses rather than on missed payments. Thus, all the above-mentioned points prove that the proposed models can efficiently predict the probability of default and estimate bank losses. As a result, they can help banks reduce losses, attract new customers, and provide services and products to worthy clients. In the future work, the model will be tested on other datasets that are transactional and non-transactional in nature to prove its efficiency. Moreover, extending the proposed model to customer credit scoring for consumer loans.

Acknowledgements This work was supported by the Office of Research, Zayed University [grant number R20053].

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Acuna E, Rodriguez C (2004) The treatment of missing values and its effect on classifier accuracy. In Journal of Classification, pp 639–647. https://doi.org/10.1007/978-3-642-17103-1_60
2. Addo P, Guegan D, Hassani B (2018) Credit risk analysis using machine and deep learning models. Risks 6(2):38. https://doi.org/10.3390/risks6020038
3. Adeodato P, Melo S (2016) On the equivalence between Kolmogorov-Smirnov and ROC curve metrics for binary classification
4. Agarwal S, Chomsisengphet S, Liu C, Song C, Souleles NS (2018) Benefits of relationship banking: Evidence from consumer credit markets. J Monetary Econ 96:16–32
5. Akkoç S (2012) An empirical comparison of conventional techniques, neural networks and the three-stage hybrid Adaptive Neuro Fuzzy Inference System (ANFIS) model for credit scoring analysis: the case of Turkish credit card data. Eur J Oper Res 222(1):168–178. https://doi.org/10.1016/j.ejor.2012.04.009
6. Ala'raj, M., & Abbod, M. F. (2016). Classifiers consensus system approach for credit scoring. Knowl Based Syst 104: 89–105. 10.1016/j.knosys.2016.04.013
7. Ala'raj M, Majdalawieh M, Abbod MF (2020) Improving binary classification using filtering based on k-NN proximity graphs. J Big Data 7(1): 1–18. https://doi.org/10.1186/s40537-020-00297-7
8. Alborzi M, Khanbabaei M (2016) Using data mining and neural networks techniques to propose a new hybrid customer behaviour analysis and credit scoring model in banking services based on a developed RFM analysis method. Int J Bus Inf Syst 23(1):1–22. https://doi.org/10.1504/ijbis.2016.078020
9. Anderson R (2007) The credit scoring toolkit: theory and practice for retail credit risk management and decision automation. OUP Oxford, p 731
10. Angelini E, Tollo G, Roli A (2008) A neural network approach for credit risk evaluation. Q Rev Econ Finance 48:733–755. https://doi.org/10.1016/j.qref.2007.04.001
11. Atiya AF, Parlos AG (2000) New results on recurrent network training: unifying the algorithms and accelerating convergence. IEEE Trans Neural Netw 11(3):697–709. https://doi.org/10.1109/72.846741
12. Baesens B, Gestel T, Van Stepanova M, Van Den Poel D, Vanthienen J (2005) Neural network survival analysis for personal loan data. J Oper Res Soc 56(9): 1089–1098. https://doi.org/10.1057/palgrave.jors.2601990
13. Baesens B, Gestel TV, Viaene S, Stepanova M, Suykens J, Vanthienen J (2003) Benchmarking state-of-the-art classification algorithms for credit scoring. J Oper Res Soc 54(6):627–635. https://doi.org/10.1057/palgrave.jors.2601545
14. Bastani K, Asgari E, Namavari H (2019) Wide and deep learning for peer-to-peer lending. Expert Syst Appl 134:209–224. https://doi.org/10.1016/j.eswa.2019.05.042
15. Bauer A, Züfle M, Herbst N, Kounev S (2019) Best practices for time series forecasting (Tutorial Paper). https://doi.org/10.1109/FAS-W.2019.00069
16. Bellotti T, Crook J (2007) Modelling and predicting loss given default for credit cards. In: Credit scoring and credit control XI conference
17. Bellotti T, Crook J (2009) Credit scoring with macroeconomic variables using survival analysis. J Oper Res Soc 60(12):1699–1707. https://doi.org/10.1057/jors.2008.130
18. Bellotti T, Crook J (2013) Forecasting and stress testing credit card default using dynamic models. Int J Forecast 29(4):563–574. https://doi.org/10.1016/j.ijforecast.2013.04.003
19. Bensic M, Sarlija N, Zekic-Susac M (2005) Modelling small-business credit scoring by using logistic regression, neural networks and decision trees. Intell Syst Account Finance Manag 13(3):133–150. https://doi.org/10.1002/isaf.261
20. Bertola G, Disney R, Grant C (2006) The economics of consumer credit
21. Bhatia S, Sharma P, Burman R, Hazari S, Hande R (2017) Credit scoring using machine learning techniques. Int J Comput Appl 161:1–4
22. Bhattacharyya S, Maulik U (2013) Soft computing for image and multimedia data processing. https://doi.org/10.1007/978-3-642-40255-5
23. Breiman L (2001) Random forests. Mach Learn 45(1):5–32. https://doi.org/10.1023/A:1010933404324
24. Brier GW (1950) Verification of forecasts expressed in terms of probability. Mon Weather Rev 78:1–3
25. Bröcker J, Smith L (2007) Increasing the reliability of reliability diagrams. Weather Forecast 22(3):651–661. https://doi.org/10.1175/WAF993.1
26. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297. https://doi.org/10.1007/BF00994018
27. Cui Z, Ke R, Wang Y (2018) Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction. arXiv preprint arXiv:1801.02143
28. Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30
29. Dietterich TG (1998) Approximate statistical tests for comparing supervised classification learning algorithms. Neural Comput 10(7):1895–1923. https://doi.org/10.1162/089976698300017197
30. Dyché J (2001) The CRM handbook: a business guide to customer relationship management. Addison-Wesley Longman Publishing Co., Inc. arXiv preprint arXiv:1801.02143
31. Feng X, Xiao Z, Zhong B, Qiu J, Dong Y (2018) Dynamic ensemble classification for credit scoring using soft probability. Appl Soft Comput 65:139–151

32. Glennon D, Kiefer N, Larson C, Choi H (2008) Development and validation of credit scoring models. J Credit Risk. https://doi.org/10.21314/JCR.2008.075
33. Graves A, Mohamed A, Hinton G (2013) Speech recognition with deep recurrent neural networks. In: ICASSP, IEEE international conference on acoustics, speech and signal processing—Proceedings, vol 38, pp 6645–6649. https://doi.org/10.1109/ICASSP.2013.6638947
34. Gui L (2019) Application of machine learning algorithms in predicting credit card default payment. University of California
35. Haltuf M (2014) Support vector machines for credit scoring. University of Economics in Prague Faculty of Finance, Department of Banking and Inscurance
36. Hancock JT, Khoshgoftaar TM (2020) Survey on categorical data for neural networks. J Big Data 7(1):28. https://doi.org/10.1186/s40537-020-00305-w
37. Hand DJ (2009) Measuring classifier performance: a coherent alternative to the area under the ROC curve. Mach Learn 77(1):103–123. https://doi.org/10.1007/s10994-009-5119-5
38. Hand DJ, Henley WE (1997) Statistical classification methods in consumer credit scoring: a review. J R Stat Soc A Stat Soc 160(3):523–541. https://doi.org/10.1111/j.1467-985x.1997.00078.x
39. Haykin SS (2009) Neural networks and learning machines (Third). Pearson Education.
40. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9:1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
41. Hsieh H-I, Lee T-P, Lee T-S (2010) Data mining in building behavioral scoring models. In: 2010 international conference on computational intelligence and software engineering, pp 1–4. https://doi.org/10.1109/cise.2010.5677005
42. Hsieh N-C (2004) An integrated data mining and behavioral scoring model for analyzing bank customers. Expert Syst Appl 27(4):623–633. https://doi.org/10.1016/j.eswa.2004.06.007
43. Huang C-L, Chen M-C, Wang C-J (2007) Credit scoring with a data mining approach based on support vector machines. Expert Syst Appl 33(4):847–856. https://doi.org/10.1016/j.eswa.2006.07.007
44. Jadhav S, He H, Jenkins K (2018) Information gain directed genetic algorithm wrapper feature selection for credit rating. Appl Soft Comput 69:541–553
45. Kavzoglu T (2017) Chapter 33—Object-oriented random forest for high resolution land cover mapping using quickbird-2 imagery (Samui P, Sekhar S, T-H VEB. of Balas NC (eds)). Academic Press, pp 607–619. https://doi.org/10.1016/B978-0-12-811318-9.00033-8
46. Kennedy K, Namee BM, Delany SJ, O'Sullivan M, Watson N (2013) A window of opportunity: assessing behavioural scoring. Expert Syst Appl 40(4):1372–1380. https://doi.org/10.1016/j.eswa.2012.08.052
47. Kim H, Cho H, Ryu D (2018) An empirical study on credit card loan delinquency. Econ Syst 42:437–449. https://doi.org/10.1016/j.ecosys.2017.11.003
48. Kumar PR, Ravi V (2007) Bankruptcy prediction in banks and firms via statistical and intelligent techniques: a review. Eur J Oper Res 180(1):1–28. https://doi.org/10.1016/j.ejor.2006.08.043
49. Lahsasna A, Ainon R, Wah T (2010) Credit scoring models using soft computing methods: a survey. Int Arab J Inf Technol 7:115–123
50. Lessmann S, Baesens B, Seow H-V, Thomas L (2015) Benchmarking state-of-the-art classification algorithms for credit scoring: an update of research. pp.124–136. Eur J Oper Res. https://doi.org/10.1016/j.ejor.2015.05.030
51. Li Y, Li Y, Li Y (2019) What factors are influencing credit card customer's default behavior in China? A study based on survival analysis. Physica A 526:120861. https://doi.org/10.1016/j.physa.2019.04.097
52. Lim MK, Sohn SY (2007) Cluster-based dynamic scoring model. Expert Syst Appl 32(2):427–431. https://doi.org/10.1016/j.eswa.2005.12.006
53. Liu Y, Schumann M (2005) Data mining feature selection for credit scoring models. J Oper Res Soc 56(9):1099–1108. https://doi.org/10.1057/palgrave.jors.2601976
54. Liu Y (2001) New issues in credit scoring application. Abteilung Wirtschaftsinformatik II, Georg-August-Universität, Institut für Wirtschaftsinformatik
55. Louzada F, Ara A, Fernandes GB (2016) Classification methods applied to credit scoring: systematic review and overall comparison. Surv Oper Res Manag Sci 21(2):117–134. https://doi.org/10.1016/j.sorms.2016.10.001
56. Malhotra P, Vig L, Shroff G, Agarwal P (2015) Long short term memory networks for anomaly detection in time series 89–94
57. Malhotra R, Malhotra DK (2003) Evaluating consumer loans using neural networks. Omega 31:83–96. https://doi.org/10.2139/ssrn.314396
58. Malik M, Thomas LC (2010) Modelling credit risk of portfolio of consumer loans. J Oper Res Soc 61(3):411–420. https://doi.org/10.1057/jors.2009.123
59. McNab H, Wynn A (2000) Principles and practice of consumer credit risk management. CIB Publishing
60. Mylonakis J, Diacogiannis G (2010) Evaluating the likelihood of using linear discriminant analysis as a commercial bank card owners credit scoring model. Int Bus Res 3(2):9. https://doi.org/10.5539/ibr.v3n2p9
61. Neto R, Adeodato P, Salgado AC (2016) A framework for data transformation in credit behavioral scoring applications based on model driven development. Expert Syst Appl 72:293–305. https://doi.org/10.1016/j.eswa.2016.10.059
62. Pereira S (2019) Modelling credit card customer behaviour. Work Project presented as a partial requirement for Degree of Master of Statistics and Information Management, with a specialization in Information Analysis and Management
63. Sarlija N, Bensic M, Zekic-Susac M (2009) Comparison procedure of predicting the time to default in behavioural scoring. Expert Syst Appl 36(5):8778–8788. https://doi.org/10.1016/j.eswa.2008.11.042
64. Setiono R, Thong JYL, Yap C-S (1998) Symbolic rule extraction from neural networks. Inf Manag 34(2):91–101. https://doi.org/10.1016/s0378-7206(98)00048-2
65. Sharda R, Wilson RL (1996) Neural network experiments in business failures predication: a review of predictive performance issues. Int J Comput Intell Organ. https://doi.org/10.1109/hicss.1993.284245
66. Siami Namini S, Siami Namin A (2018) Forecasting economics and financial time series: ARIMA vs. LSTM
67. So MMC, Thomas LC (2011) Modelling the profitability of credit cards by Markov decision processes. Eur J Oper Res 212(1):123–130. https://doi.org/10.1016/j.ejor.2011.01.023
68. Sutskever I, Vinyals O, Le QV (2014) Sequence to sequence learning with neural networks. In: Advances in neural information processing systems, pp 3104–3112
69. Thomas LC, Ho J, Scherer W (2001) Time will tell: behavioural scoring and the dynamics of consumer credit assessment. IMA J Manag Math 12(1):89–103. https://doi.org/10.1093/imaman/12.1.89
70. Thomas LC (2000) A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers. Int J Forecast 16(2):149–172. https://doi.org/10.1016/s0169-2070(00)00034-0
71. Wang C, Han D, Liu Q, Luo S (2018) A Deep learning approach for credit scoring of peer-to-peer lending using attention

mechanism LSTM. IEEE Access 7:2161–2168. https://doi.org/10.1109/access.2018.2887138
72. Wang L, Lu W, Malhotra NK (2011) Demographics, attitude, personality and credit card features correlate with credit card debt: a view from China. J Econ Psychol 32(1):179–193. https://doi.org/10.1016/j.joep.2010.11.006
73. Witten IH, Frank EF, Hall MA (2011) Credibility: evaluating what's been learned. In Witten IH, Frank E, Hall MA (eds) Data mining: practical Machine learning tools and techniques, 3rd edn. Morgan Kaufmann, pp 147–187 https://doi.org/10.1016/B978-0-12-374856-0.00005-5
74. Xie Y, Liu G, Cao R, Li Z, Yan C, Jiang C (2019) A feature extraction method for credit card fraud detection. In: 2019 2nd international conference on intelligent autonomous systems ({ICoIAS}). https://doi.org/10.1109/icoias.2019.00019
75. Yeh IC, Lien CH (2009) The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. Expert Syst Appl 36(2):2473–2480
76. Yu J, Yao J, Chen J (2019) Credit scoring with AHP and fuzzy comprehensive evaluation based on behavioural data from weibo platform. Tehn Vjes Tech Gaz 26(2):462–470. https://doi.org/10.17559/tv-20181217180231
77. Yu L, Liu H (2003) Feature selection for high-dimensional data: a fast correlation-based filter solution. Proc Twentieth Int Conf Mach Learn 2:856–863

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).
Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.
These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.
We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.
While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.
These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.
To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.
Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.
If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com