Application of an Improved Optimization Using Learning Strategies and Long Short Term-Memory for Bankruptcy Prediction

Juliana Adeola Adisa, Samuel Ojo, Pius Adewale Owolawi, Agnieta Pretorius, Sunday Olusegun Ojo,

***Abstract*—Over the past century, bankruptcy prediction has dominated economics headlines and continues to do so. An efficient way to deal with bankruptcy is to develop models that integrate multiple econometric measures and predicts a company's financial status. Different methods have been proposed in this domain ranging from statistical algorithms to artificial intelligence models. A novel approach to bankruptcy prediction was proposed in this paper. The novel method incorporates an improved particle swarm optimization (PSO) approach for determining key features and optimizing the long short-term memory system for bankruptcy prediction. This paper employs self-learning and multi-learning strategies to develop the cognitive and social learning parts of the PSO algorithm. The PSO algorithm then determines the optimal LSTM parameters. Our final bankruptcy prediction approach combines optimal feature selection with the LSTM model. Therefore, an improved method of feature selection was developed for the LSTM model. The data for evaluation is available in the machine learning repository at the University of California Irvine. We compare our improved model with the ordinary LSTM model, and the results show better performances in all the metrics for the improved LSTM model compared to the other models.**

***Index Terms*—bankruptcy prediction, particle swarm optimization, Long Short Term Memory, feature selection.**

I. Introduction

THE advancement of computing technology has caused massive amounts of data and information to accumulate. The amount of data available is overwhelming, and a large proportion of this data is yet to be understood. There is a gap between the available large volume of data and the understanding of the data itself [1]. An increase in data size leads to an inexorable decrease in people's understanding of the data. Therefore, the issue of copious data but low information [1]. In recent years, big data analytics has been the driving force behind the advancement of artificial intelligence. As a result, data analysis will provide a vital contribution to building more sustainable societies. In finance, a large amount of real-time data, which includes transaction records, is generated. Analyzing the bankruptcy data will lead to the discovery of valuable insights.

Bankruptcy can happen to any enterprise. In reality, bankruptcy has many processes, forms, different results, and various consequences. The effects inspire research into the possibilities of methods and models that researchers can employ in predicting failure and determining what the future holds for businesses over the coming years.

Financial distress prediction (FDP) in the form of corporate failures and bankruptcies is more problematic than in the past due mainly to globalization and the interdependence of economies [2]. In addition, businesses and the economies of different countries are increasingly interdependent and interact intensively [2]. Therefore, a financial crisis can spread from one country to another and produce global effects in no time to all [2].

Horak *et al* [3] emphasized that a good prediction helps business owners and management with strategic and operational decision-making. However, predicting a company's failure is difficult, especially during a crisis period [4], [5]. Bankruptcy prediction depends on the unique characteristics of a country, such as socio-economic and legal environments [6], [7]. Therefore, the adaptation of those models to the specific conditions of a country is a challenge [6], [7]. Despite this fact, especially in the context of global economics and financial instability, there is a need for generally valid models which transcend regionally localized prediction systems Horak *et al* [8] and Alaminos *et al* [9].

Researchers have previously used statistical tools and artificial intelligence tools [10], [11] for FDP. According to a study conducted by Barboza et al. [12] and Horak *et al* [13], traditional models created based on statistical methods are 10% less accurate on average than those constructed using machine learning algorithms. The rapid development and improvement of machine learning models in many other areas is another factor contributing to their popularity [14].

In statistical models, the functional relationships between variables, which are dependent and independent, are determined by the researcher [15]. In addition, traditional neural networks do not incorporate the "temporal effects" of past major events [15]. Artificial neural networks (ANN) cannot handle sequential data effectively. Hence, the introduction of deep learning [15].

Long short-term memory (LSTM) unit is a deep neural network and variant of recurrent neural networks (RNNs)

Manuscript received April 07, 2022; revised February 17, 2023.

This work was supported by National Research Foundation (NRF) South Africa, Grant number 12908. The authors acknowledge the Computer Systems Engineering department of the Tshwane University of Technology for providing the facilities used for this research.

Juliana. A. Adisa is a PhD Student at Tshwane University of Technology, Department of Computer Systems Engineering, Soshanguve Campus, South Africa (corresponding phone: +27848845930; e-mail: adeolaa33@gmail.com).

Samuel Ojo is a PhD Student at Tshwane University of Technology, Department of Computer Systems Engineering, Soshanguve Campus, South Africa (e-mail: 208322842@tut4life.ac.za).

Pius Adewale Owolawi is the Head of Department of Computer Systems Engineering, Tshwane University of Technology, Soshanguve Campus, South Africa (e-mail: owolawipa@tut.ac.za).

Anieta Pretorius is the Deputy Dean Faculty of ICT, Tshwane University of Technology, Soshanguve Campus, South Africa (e-mail: pretoriusab1@tut.ac.za).

Sunday Olusegun Ojo is a Professor of Computer Science, Department of Information Technology, Durban University of Technology, Durban, South Africa (e-mail: sundayO1@dut.ac.za).

[16]. In practice, learning using RNN becomes ineffective because of the vanishing gradient problem of long-term dependencies. Finance crises are often not confined to short-term economic factors, but long-term conditions must also be considered [16]. Hence, there is a need for a model that can capture the long-term dependencies context. Deep learning LSTM models are applied in a variety of applications because it produces excellent results in the area of dynamic counting [17]. LSTM models are often used to predict financial market or real estate prices. They are employed to analyze high-frequency trading patterns [18] and predict future trends in prices [19]. The results produced by the LSTM models are better than those with vector regression, backpropagation neural networks, etc.

Although LSTM networks are powerful tools for pattern recognition problems and they play a significant role in solving time series problems, there are several drawbacks associated with their use [20]. The first problem with neural network models, such as LSTMs, is that they cannot provide clear indications of the final decision made by the models. In addition, the models of neural networks employ a highly complex computational method to achieve the target solution [20]. Furthermore, the researchers need to change some of the parameters of the LSTM network. Unfortunately, time constraints and computation power prevent finding the optimal parameters for the LSTM network. Traditionally in previous research, the control parameters have been determined by the experience and knowledge of the researchers [20].

This study incorporates big data and artificial intelligence technologies for financial distress prediction to manage potential risks on financial institutions and help achieve sustainable socio-economic. Therefore, we propose to develop a new failure prediction model using artificial neural networks with an LSTM layer to integrate available financial data. Our approach relies on deep learning because of its excellent capacity to learn from large data sets. Therefore, we posed the following research question: Can LSTM-based neural networks be used to forecast potential bankruptcy in companies?

Furthermore, an optimization approach was proposed. A hybrid model that combines the LSTM network with the particle swarm optimization (PSO) approach was developed. The PSO is employed to find and identify the suitable features and optimal model parameters for bankruptcy prediction. The best optimum solution derived from PSO was employed for feature selection, and the performance was evaluated using different metrics.

There are six sections in this paper. Sections I and II present the introduction and literature reviews respectively. The theory is presented in Section III, and Section IV presents the methodology. Section V discuss the results. Section VI presents the research summary and conclusion.

II. Literature Review

A. An Early Approach to Financial Distress Prediction

In the past years, there have been numerous studies on the bankruptcy prediction of organizations. Authors have developed several different predictive models based on various algorithms and variables. The goal of those predictive models by authors (early warning systems) is to employ past information/results to determine a company's financial health. The studies by Altman [21] and Beaver [22] introduced the single-variable and multi-variable models. They paved the way for numerous subsequent research studies that focused on financial distress prediction. Several research studies and reviews have been conducted, and models were developed to reflect the specific condition of a particular country. In addition, companies from different economic sectors generally make up the data set. Lastly, models' development relies on different methods.

In 1968, Altman [21] applied multivariate discriminant analysis to distinguish between solvent and insolvent firms. The early applied statistical models for FDP include linear discriminant analysis, multivariate discriminant analysis, quadratic discriminant analysis, logistic regression, factor analysis, probit analysis, and linear programming etc. However, there are limitations to these conventional statistical methods due to their strict assumptions regarding linearity, normality, and independence of predictors or input variables [23].

B. Deep Learning and hybrid approach to Financial Distress Prediction

Over the past few years, machine learning and artificial intelligence techniques have rapidly grown and proven to be more appropriate than conventional statistical models regarding failure prediction [23]. In comparison to traditional statistical methods, artificial intelligence is an efficient method to predict the financial state of a company as it does not depend on prior assumptions about how the variables affect the data.

Deep learning describes artificial neural networks (ANNs) where the input layer is interconnected with multiple hidden layers. In general, deep learning algorithms have received considerable attention because they demonstrated excellent predictability results in natural language processing (NLP) and image classification. Deep belief networks (DBN), convolutional neural networks (CNN), and recurrent neural networks (RNN) are examples of deep learning algorithms. This subsection presents the recent application of deep and machine learning models for bankruptcy predictions given by [24], [25], [26], [27], and [28].

Shetty *et al* [24] implement a very simple and user-friendly model to distinguish between bankruptcy and non-bankruptcy. Cao *et al* [25] applied the least absolute shrinkage selection operator (LASSO) to select financial ratios and develop a Bayesian network for bankruptcy prediction. Brygala [26] applied logistic regression for bankruptcy prediction and established that the prediction performance of the balanced sample compared to the imbalance sample is more effective using the logit model. Jabeur *et al* [27] employed the XGBoost algorithm for feature importance selection, and the results show an accurate prediction over traditional methods. Perez-Pons *et al* [28] employed attribute reduction by automated binary classification and metrics evaluation to solve the bankruptcy problem.

Luo et al. [29] investigate the performances of deep belief networks (DBN) with Restricted Boltzmann Machines for corporate credit scoring using CDS data sets. The performance criteria employed were accuracy, AUROC, FN, and

FP. The results show that DBN with RBM performed better among the models tested [29]. Yu *et al* [30] employed a cascading hybrid model to solve the imbalanced data problem in credit classification. The cascading hybrid model combined a DBN with a support vector machine (SVM). The accuracy results achieved were above 80%–90% [30]. Models with DL showed good performance at predicting crises. Deep learning models were highly accurate than machine learning models [31].

III. Theory and Methods

A. Long Short Term-Memory (LSTM)

A recurrent neural network (RNN) is a deep neural network used for modelling time series with long-range structural dependencies. RNN used the time delay and feedback connection features to import information from the previous time step to the present state. Therefore, the LSTM network will receive temporal sequences as input and produce temporal sequences as output [16], [32]. One of the challenges of RNN is its ineffectiveness in modelling relationships as the time distance grows between cause and effect. It means that RNNs are not effective for long-term dependency problems. One key feature of learning models is the ability to capture long-term dependencies. A Crisis does not depend only on the short-term economic context but the long-term context [16]. Therefore, LSTM networks were applied to the financial domain [16].

In 2017, Gilardoni [16] gave an overview of LSTMs applications in the literature. LSTM has been found successful in the following areas: language modelling, natural language generation, machine translation, text classification, handwriting recognition/generation, video translation to natural language, anomaly detection, speech recognition, question-answering systems, image caption generation and Android mobile phones.

An LSTM network has a repeating module made of four connected layers known as the four gates[33]. These gates of the LSTM were developed from a sigmoid and a pointwise multiplication operation. These gates are optionally used to let information through; steps 1 to 4 give the equations of these gates [20]:

The forget gate is the gate that identifies the information that needs to be removed from the memory cell state.

$$f^t = \sigma(W_f[o^{(t-1)}, x^t] + b_f) \tag{1}$$

The input gate is the gate that decides, identifies and add new information to the memory cell state.

$$i^t = \sigma(W_i[o^{(t-1)}, x^t] + b_i) \tag{2}$$

$$\hat{C}^t = tanh(W_c[o^{(t-1)}, x^t] + b_c) \tag{3}$$

Memory Cell updates the old cell state $C^(t-1)$ into a new cell state C^t and stores some output.

$$c^t = f^t * c^{(t-1)} + i^t * \hat{C}^t \tag{4}$$

The output gate is the gate that decides on the output. That is, what is the effects of the memory cell on the output of the LSTM cell.

$$h^t = \sigma(W_h[o^{(t-1)}, x^t] + b_h) \tag{5}$$

$$o^t = h^t * tanh(c^t) \tag{6}$$

$$y^t = \sigma(W^{out}o^t + b^{out}) \tag{7}$$

The weight matrices in the equations are denoted W_f, W_i, W^out at the forget gate, input gate and output gate respectively. The bias for the respective gates are given as b_f, b_i, and b^out. The the input at time t is represented as x^t and the output at time t-1 is denoted as $o^{(t-1)}$.

B. Baseline Classification Model

The artificial neural network (ANN) is a technique that learns and generalizes from previous experiences. They are simulated like neurons in the human brain using mathematics modelling, and it is an efficient model for statistical pattern recognition [34], [35]. Bishop et al [36] described ANN as a black box non-parametric classifier that does not need assumptions about the distribution densities. The ANN create an architecture that connects neurons among layers; the sets of input variables are mapped to the output variables [34], [37].

A decision tree (DT) is a supervised machine learning technique that employs the "divide-and-conquer" rule to splits the data into leaves which denote outcomes, nodes are attributes, and the branches represent decision rules [38]. The root node represents the node with the highest information gain. The features are compared until no more attributes are available for partitioning [38].

Vapnik et al [34] introduced the support vector machine (SVM) as a machine learning technique used for classification and regression problems (tasks). SVM is highly efficient for linearly separable and high-dimensional feature space data. SVM deals with the non-separable data by employing different kernel functions that transform the data into a higher-dimensional space and hence linearly separable [34].

C. Particle swarm optimization (PSO)

The concept of particle swarm optimization (PSO) is a new embranchment of swarm intelligence and an evolutionary algorithm. Kennedy, Eberhart [39], and Shi [40] attribute swarm-like movements to organisms in frenzied groups like birds and fish. PSO is based on swarm intelligence that simulates predation among birds or fishes [38]. PSOs allow particles to learn from one another through information-sharing mechanisms to develop a population. In a PSO algorithm, every solution to a given problem is considered a particle that flies through space. Scholars and researchers have shown a great interest in PSO because of its simplicity, easy implementation, fast convergence, strong manoeuvrability and few parameters that need to be adjusted. So far, PSO has proven to be successful in many fields: artificial neural network training [41], fuzzy system control [42], power systems [43], manipulator motion planning [44], optimal control [45], image processing [46], artificial neural networks [47].

Initially, random particles (solutions) are generated, and each fitness function is examined for an optimal solution [48]. During each iteration of the particle, there are two kinds of fitness values: first, there is pbest, which represents the best location of the particle based on the history of the person. Additionally, any particle within the population can obtain a global best (gbest) value [49]. A particle updates its position based on two vectors. The position vector presents a particle's position in a landscape, while the velocity vector shows the direction and intensity of the movement. After finding the pbest and gbest, the PSO updates its velocity and position.

1) Basic PSO Algorithm: Particles in the PSO algorithm have their trajectories. Therefore, each particle moves according to its position x_i and velocity v_i in the search space. Particle positions reflect the potential solutions, and they evolved according to fitness functions. In the d-dimensional search space, equations 8 and 9 are used to describe the particle positions and velocity, respectively, as given by [49]:

$$X_i = [x_i^1, x_i^2, ..., x_i^D] \tag{8}$$

$$V_i = [v_i^1, v_i^2, ..., v_i^D] \tag{9}$$

During the search process, PSO employs the pbest, gbest and the current velocity to adjust the trajectory of the ith particle. Consequently, the velocity vector and position vector of particles along the d dimensions are updated using equations (8) and (9) as follows [49]:

$$\begin{aligned} v_i^d(t+1) = & w.v_i^d(t) + c_1.rand_1^d.(pbest_i^d(t) - x_i^d(t)) \\ & + c_2.rand_2^d.(gbest^d(t) - x_i^d(t)) \end{aligned} \tag{10}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{11}$$

The inertia weight w, represents the percentage change in velocity, and it is used to maintain the direction. The acceleration parameters are c_1 and c_2 while $rand_1^d$ and $rand_2^d$ in the range of [0, 1] represent uniformly distributed random numbers. The $pbest_i^d(t)$ denotes the personal history optimal position while the $gbest^d(t)$ is the best position in the global history of the *ith* particle. There are different terms for each component: inertia which maintains the current velocity and direction; cognitive component or individual component here a particle swarm considers distances between her best location and her current location on a cognitive level; social component, the particle calculates the distance between its current location and the swarm's best location found by the entire group. Generally, the best solution in a particle's neighbourhood determines the social component of the PSO algorithm.

2) Improved PSO: A PSO algorithm performance can be improved by controlling the probability distribution of the population [49]. Learning strategies such as self-learning probability and multi-learning strategy led to an improvement of the basic PSO algorithm. A self-learning probability and multi-learning strategy are applied empirically to improve diversity in population and convergence speed.

A self-learning strategy is used for a particle to learn in the cognitive learning part of a PSO algorithm, and it must learn from its pbest only. Recall that pbest is the particle's historical best position. However, a better particle swarm will not necessarily result from using the pbest within the current evolutionary search space. Consequently, there is an ineffective utilization of information available from the best particles. Self-learning strategies provide a way to improve the cognitive learning part of this algorithm. The fitness function is employed to evaluate the pbest, and all particles in the personal best solution will learn stochastically from any better particles. Following the self-learning strategy, pbest can be expressed as $pbest_{si}^d$ for the *ith* particle on dimension d [49]. Thus, the $pbest_{si}^d$ can be substituted into equation (10) to replace the cognitive learning part $pbest_i^d$.

Multilearning strategy: The gbest is the social learning part of equation 10, which improves the convergence speed, but the diversity of particles in the population is diminished. Hence, the particles gravitate toward the local optimal region. The global search capabilities of the PSO algorithm depend on population diversity. A multi-learning strategy is employed to update the velocity to enhance the social component of equation 10. Therefore, the need to develop an improved PSO.

Based on self-learning and multi-learning strategies, the pbest and gbest of the cognitive and social components can be replaced with $pbest_{si}^d$ and $gbestk^d$, respectively. That is, $pbest_i^d$ and $gbest^d$, replaced with $pbest_{si}^d$ and $gbestk^d$, respectively.

We can write the velocity updating equation (10) as:

$$\begin{aligned} v_i^d(t+1) = & w.v_i^d(t) + c_1.rand_1^d.(pbest_{si}^d(t) - x_i^d(t)) \\ & + c_2.rand_2^d.(gbestk^d(t) - x_i^d(t)) \end{aligned} \tag{12}$$

IV. Proposed Method

The process involves the collection of the bankruptcy dataset, preprocessing, finding the best features using PSO, the LSTM architecture, optimizing PSO parameters, splitting of the dataset in the ratio of training set 70% and testing set 30% and evaluating the results. These stages are illustrated in Figure 1.

A. Classifiers

Figure 1 depicts the experimental process. At the initial stage, no feature selection method was applied to the data sets. These classifiers are the baseline models for this research work. They include ANN, DT, SVM and LSTM.

Table I contains the various data sets, such as Australian and Polish data. The research applied SMOTE technique to the Polish data set. Therefore, Table I also includes the data sets after the application of SMOTE on the Polish data set.

B. Collection of data sets

The data sets are from the repository of the UC Irvine Machine Learning. These are Australian credit and Polish data sets. The Australian data has 690 instances and 14 attributes. The Australian data is balanced. Therefore, the oversampling technique is unnecessary, and the data link is given as https://archive.ics.uci.edu/ml/datasets/Statlog+ (Australian+Credit+Approval). The Polish data has 1000 companies, where 19.4% of them were bankrupt between 2000-2012, and the non-bankrupt companies were evaluated from 2007 to 2013. In the first year, the Polish data set had 271 bankrupted companies and 6756 non-bankrupts. Polish data contains 7027 instances and 64 attributes. The Polish data set is imbalanced and has high dimensionality. One important evaluation metric for imbalanced data is AUC [50]. First, we generated synthetic samples of the

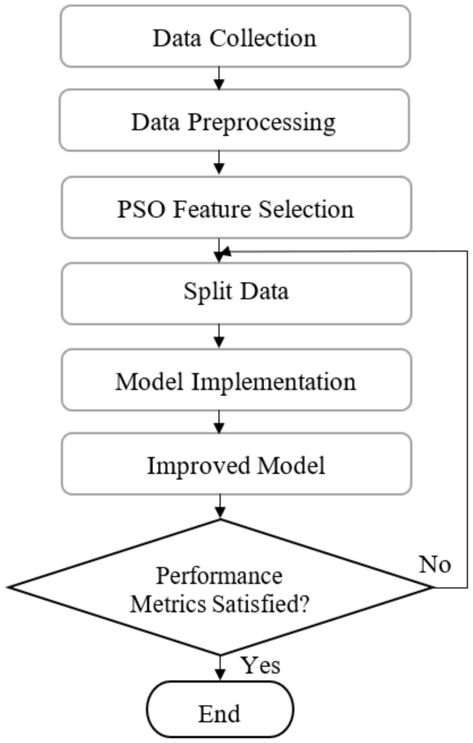

Fig. 1. Methodology Framework

attributes in the minority class using the synthetic minority over-sampling technique (SMOTE) method. The SMOTE method allows for the addition of a few copies of each instance from an underrepresented class. In addition, an oversampling technique that up-sampled the minority class was applied and compared with the SMOTE application.

TABLE I
DATASET INFORMATION

Dataset	Total Instances	No of Attributes	Minority Class	Majority Class	IR
Australian Data	690	14	307	383	0.80
Polish Data	7027	65	271	6756	0.040
Polish + SMOTE	11092	65	4336	6756	0.64
Polish + UP-Sample	11092	65	5546	5546	0.50

Table I summarizes the data sets. It shows the summary of the Australian credit and Polish bankruptcy data sets before and after the application of SMOTE. The IR represents the minority class ratio. The Australian data is balanced, while the Polish data is highly skewed, hence the application of SMOTE operation to make the data balanced.

C. Performance Measures

There are several evaluation metrics proposed in the literature for measuring classifier performance. These metrics evaluate or measure the ability of the learning algorithms. In most cases, accuracy is the metric used for performance evaluation. However, accuracy is not a suitable metric for an imbalanced data set because the effect of the negative course on the classification rate (accuracy) is higher than the positive course of instruction. Hence, there is a need to employ other metrics for the performance assessment of classification algorithms for an imbalanced data set. The performance metrics used in this paper are computed from the confusion matrix.

Accuracy measures the ratio between the positive results and the overall data set, while error is the amount of incorrectly classified data. Recall or sensitivity refers to the proportion of correctly classified positive instances, and specificity measures the true negative results. Precision is a measure of exactness; it determines the rate of positive prediction over total positive instances. A systematic measure that includes both precision and recall is the F-Measure.

The receiver operating characteristic curve (ROC curve) is a chart that demonstrates the diagnostic capability of a binary classifier over a range of discrimination threshold values. The ROC curve explains whether the model is able to distinguish between true positives and true negatives classes.

Kappa statistics K measures the true agreement achieved that is not the result of chance [51]. In this metric, the agreement level between observations is reported quantitatively and useful when dealing with either binary or multiclass classification problems. P_o is the probability of agreement observed, while P_e is the probability of agreement expected by chance [51].

Matthews correlation coefficient (MCC) is a metric that determines the quality of the classification problem, and its values typically range between -1 and +1. As the MCC becomes closer to +1, the classification task becomes more accurate. G-mean determines the sensitivity and specificity ratios of classification. It gives a sense of the overall performance of the model. Computation time is the time the model takes to complete execution.

Type I error occurs when a bankrupt company is classified into a non-bankrupt group. Table II gives the mathematical expressions for the performance metrics used in this paper and the equation for the MCC given by [52] as:

$$MCC = \frac{(TN * TP) - (FN * FP)}{(TP + FP)(TP + FN)(TN + FP)(TN + FN)} \quad (13)$$

where TP, FP, TN, FN denote True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) respectively.

D. Data Preprocessing

To avoid bias in the dataset, we perform the following preprocessing operations: (1) The missing values are replaced with the mean and mode values of the data. (2) Remove outlier data. (3) Normalization: The data is normalized to remove the dimensional influence between indexes and accelerate the gradient descent speed for optimal solution generation. Normalization follows the following principle:

$$\hat{x} = \frac{x - min(x)}{max(x) - min(x)} \quad (14)$$

where the maximum and minimum values of *x* are represented as min (x) and max (x) respectively.

E. Feature Selection

The data set needs to be cleaned of redundant and irrelevant attributes. Feature selection is a filter that mutes ineffective characters of the data set. Therefore, we opted to use an improved particle swarm optimization and the filter method as feature selection methods. The filter method assigned a relevant score to each feature in the data set and ordered them according to their relevance score.

Features with high scores are selected, while low scores are eliminated. We applied the improved PSO for feature selection using learning strategies based on self-learning and multi-learning strategies to obtain the cognitive and social components. It means that we substituted the cognitive and social learning components of the PSO algorithm with the improved equation: $pbest_i^d$ and $gbest^d$ replaced with $pbest_{si}^d$ and $gbestk^d$ respectively. The algorithm drops the features that have zero contribution to the classification process. Table III shows the attributes after PSO feature selection process.

F. LSTM Architecture

In this research, we employ the LSTM model, and the experimental framework is implemented in Google collaboratory using the Python deep learning library called KERAS.

G. LSTM Feature selection Optimized with Improved PSO

We improved the PSO by adopting the learning strategy. Therefore, the pbest and gbest of the cognitive and social components were replaced with $pbest_{si}^d$ and $gbestk^d$, respectively. Figure 2 illustrates the optimized LSTM architecture with improved feature and parameter selection. PSO algorithm is being implemented with Pyswarm for this research. An experimental framework is implemented in Google collaboration using the Python deep learning library KERAS. For the configuration, the size of the swarm and the maximum number of iterations for the PSO are set to 5. The lower and upper bounds are set from 1 to 31 for the PSO configuration.

V. RESULTS AND DISCUSSION

A. Results

1) Baseline Classifiers: The baseline model results summarized in Table IV are based on the performance metrics employed in section IV (C). It shows the baseline classifiers performed well on the Polish data, while the kappa statistic and AUC values justify the need for improvement to derive the best bankruptcy prediction. The AUC of 0.856 indicates the model has an 86% chance of differentiating between positive and negative classes. The same interpretation is true for the other AUC values. The overall experimental outcome showed that the baseline DT model is better than other models, with an accuracy of 86%.

TABLE IV
BASELINE MODEL RESULTS

Models	Precision	Recall	F-measure	Kappa St.	AUC	MCC	G-mean	Time
DT	0.852	0.850	0.850	0.507	0.856	0.774	0.845	3.96
SVM	0.845	0.835	0.835	0.483	0.842	0.648	0.808	252.11
MLP	0.827	0.827	0.827	0.461	0.827	0.724	0.831	135.7
LSTM	0.910	0.910	0.901	0.511	0.85	0.890	0.902	5448

In addition, the baseline models in Table IV include metrics such as the matthews correlation coefficient (MCC), the G-means and computation time. The time refers to the model execution time, where the DT takes a few seconds for execution, and the LSTM model takes hours before completion. Figure 3 compares the accuracy of all the models with and without SMOTE application, and Figure 4 shows the MCC graph for all the models.

2) Improved LSTM-PSO Classifier: Table V presents the improved LSTM-PSO model results and compares the results to the baseline LSTM model. Table V shows the application of SMOTE and the PSO feature selection on the Polish data set. The results showed the positive impact of the SMOTE and PSO application on the LSTM model. The kappa value of 0.94 for LSTM-PSO models on the scale of 0-1 [53] suggests that the actual and predicted classes are in considerable agreement. Furthermore, this is an affirmation of the 96% accuracy that was achieved. The LSTM-PSO model with SMOTE applied has the best accuracy for the Polish data set. Table V shows an AUC of 0.99, which means that the model can distinguish between classes effectively. Figure 5 shows the ROC-generated curve after SMOTE application for the improved LSTM-PSO model.

TABLE V
IMPROVED LSTM-PSO MODEL COMPARED TO LSTM MODEL

Models	Accuracy	Class Accuracy	Precision	Recall	F-measure	Kappa St.	AUC
LSTM	0.91560	class 0 = 0.90302 class 1 = 0.91560	0.910	0.910	0.901	0.511	0.850
LSTM + PSO	0.96209	class 0 = 0.96521 class 1 = 0.96083	0.97	0.97	0.97	0.944	0.99

TABLE II
PERFORMANCE METRICS

Accuracy	Error	Precision	G-mean
$\frac{TP+TN}{TP+TN+FP+FN}$	1 - Accuracy $= \frac{(FP+FN}{TP+TN+FP+FN}$	$\frac{TP}{TP+FP}$	$\frac{TP}{TP+FN} * \frac{TN}{TN+FP}$
Recall/Sensitivity	**F-Measure**	**Kappa statistics**	**Type 1 error**
$\frac{TP}{TP+FN}$	$\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$	$K = \frac{(P_o - P_e)}{(1 - P_e)}$	$\frac{FP}{FP+TN}$

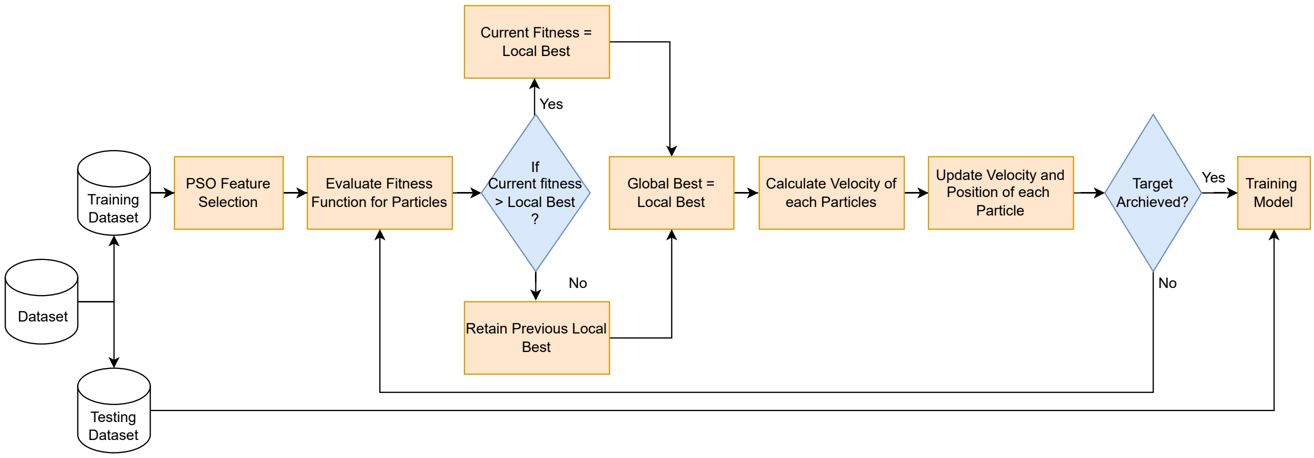

Fig. 2. LSTM optimized with PSO.

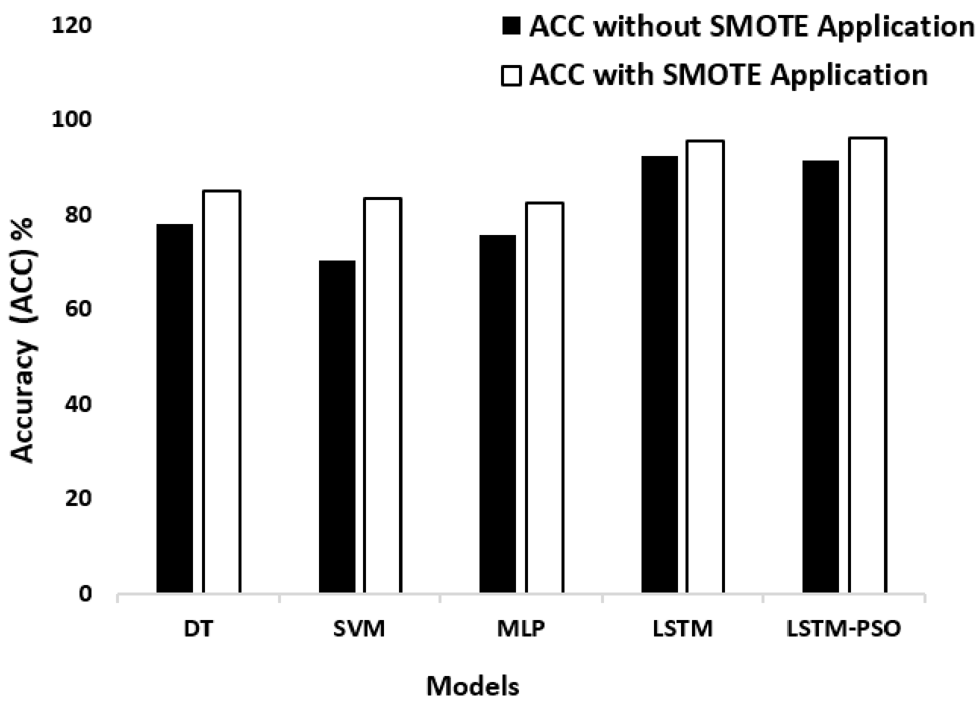

Fig. 3. Accuracy for the Models (ACC).

TABLE III
THE PSO SELECTED FEATURES

ID	Description	ID	Description
X2	$\frac{\text{total liabilities}}{\text{total assets}}$	X33	$\frac{\text{operating expenses}}{\text{short-term liabilities}}$
X5	$\frac{\text{cash + short-term securities + receivables - short-term liabilities}}{\text{operating expenses - depreciation}} \times 365$	X34	$\frac{\text{operating expenses}}{\text{total liabilities}}$
X6	$\frac{\text{retained earnings}}{\text{total assets}}$	X35	$\frac{\text{profit on sales}}{\text{total assets}}$
X8	$\frac{\text{book value of equity}}{\text{total liabilities}}$	X37	$\frac{\text{current assets - inventories}}{\text{long-term liabilities}}$
X12	$\frac{\text{gross profit}}{\text{short-term liabilities}}$	X38	$\frac{\text{constant capital}}{\text{total assets}}$
X14	$\frac{\text{gross profit + interest}}{\text{total assets}}$	X39	$\frac{\text{profit on sales}}{\text{sales}}$
X15	$\frac{\text{total liabilities} \times 365}{\text{gross profit + depreciation}}$	X41	$\frac{\text{total liabilities}}{\text{profit on operating activities + depreciation}} \times \frac{12}{365}$
X16	$\frac{\text{gross profit + depreciation}}{\text{total liabilities}}$	X42	$\frac{\text{profit on operating activities}}{\text{sales}}$
X17	$\frac{\text{total assets}}{\text{total liabilities}}$	X43	rotation receivables + inventory turnover in days
X19	$\frac{\text{gross profit}}{\text{sales}}$	X44	$\frac{\text{receivables} \times 365}{\text{sales}}$
X20	$\frac{\text{inventory} \times 365}{\text{sales}}$	X45	$\frac{\text{net profit}}{\text{inventory}}$
X21	$\frac{\text{sales (n)}}{\text{sales (n-1)}}$	X48	$\frac{\text{EBITDA (profit on operating activities - depreciation)}}{\text{total assets}}$
X23	$\frac{\text{net profit}}{\text{sales}}$	X52	$\frac{\text{short-term liabilities} \times 365}{\text{cost of products sold}}$
X24	$\frac{\text{gross profit in 3 years}}{\text{total assets}}$	X53	$\frac{\text{equity}}{\text{fixed assets}}$
X27	$\frac{\text{profit on operating activities}}{\text{financial expenses}}$	X55	working capital
X28	$\frac{\text{working capital}}{\text{fixed assets}}$	X57	$\frac{\text{(current assets - inventory - short-term liabilities)}}{\text{(sales - gross profit - depreciation)}}$
X29	logarithm of total assets	X58	$\frac{\text{total costs}}{\text{total sales}}$
X30	$\frac{\text{(total liabilities - cash)}}{\text{sales}}$	X59	$\frac{\text{long-term liabilities}}{\text{equity}}$
X32	$\frac{\text{(current liabilities} \times 365)}{\text{cost of products sold}}$	X60	$\frac{\text{sales}}{\text{inventory}}$
X62	$\frac{\text{(short-term liabilities} \times 365)}{\text{sales}}$	X64	$\frac{\text{sales}}{\text{fixed assets bankruptcy status after 5 years}}$

We applied two different feature selection methods called PSO and filter gain. In addition, there are two oversampling techniques; SMOTE and Up-sampling methods. Table VI shows the improved LSTM feature selection at different SMOTE applications. Table VI shows the output results of the models at different percentages of the SMOTE application. The Results showed better performance in the feature selection at 100% for both PSO and filter gain methods.

TABLE VI
IMPROVED LSTM FEATURE SELECTION AT DIFFERENT SMOTE APPLICATION.

Model	SMOTE	Accuracy %	Precision	Recall	F-measure	Kappa St.	AUC
LSTM + PSO	10%	90.51	0.902	0.905	0.903	0.752	0.922
	50%	92.45	0.926	0.922	0.919	0.796	0.903
	100%	96.21	0.970	0.970	0.970	0.944	0.990
LSTM + filter	10%	89.35	0.905	0.894	0.890	0.763	0.900
	50%	91.71	0.915	0.917	0.916	0.771	0.904
	100%	95.87	0.934	0.956	0.938	0.783	0.950

Figures 6 and 7 show the confusion matrix (CM) for the improved LSTM model at 100% SMOTE application. The CM show the miss-classification of data in the algorithm. According to Figure 6, the LSTM-filter selection gain models isolated legitimate bankrupt and non-bankrupt companies at 100% SMOTE applications; however, the model made quite a few mistakes in the bankruptcy prediction. Figure 7 depicts the CM for the LSTM-PSO model at 100% SMOTE application. The LSTM-PSO model has a better confusion matrix Figure 7 compared to Figure 6. Therefore, the LSTM-PSO model achieved the optimal performance for the bankruptcy prediction at 100% SMOTE application, as shown in Table VI.

3) Results on Australian Credit and Polish Bankruptcy data sets: Tables VII depicts the results of various classifiers on the Australian credit data set. The results are intriguing because feature selection generally improves the prediction accuracy of the classifiers beyond that of the baselines. However, the classification process after feature selection does not always imply that they can offer lower Type I errors than the baselines.

In addition, the PSO-based feature selection method produced the highest prediction accuracy over the LSTM model. However, the filter feature selection method over the LSTM model has the best Type 1 error. Overall, the PSO selection performs better than the filter, and the findings are supported by Hua et al. [54] that wrapper-based approaches which include PSO may outperform filter-based ones.

Over the Polish bankruptcy data set depicted in Table VIII, the outcomes demonstrate that PSO feature selection approaches typically outperform filter-based ones and the baseline models. In particular, the PSO feature selection has the highest accuracy and the least Type I error on all the models employed.

However, on average, for the Polish data set, implementing feature selection improves the performance of the classifiers relative to those without feature selection in terms of predic-

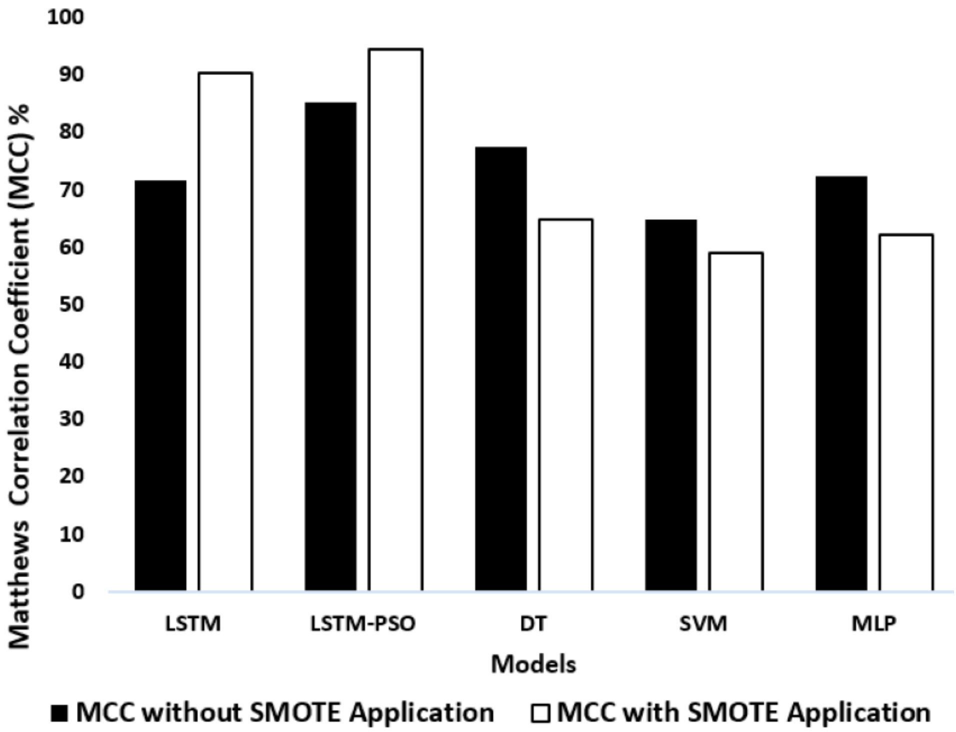

Fig. 4. MCC for all Models.

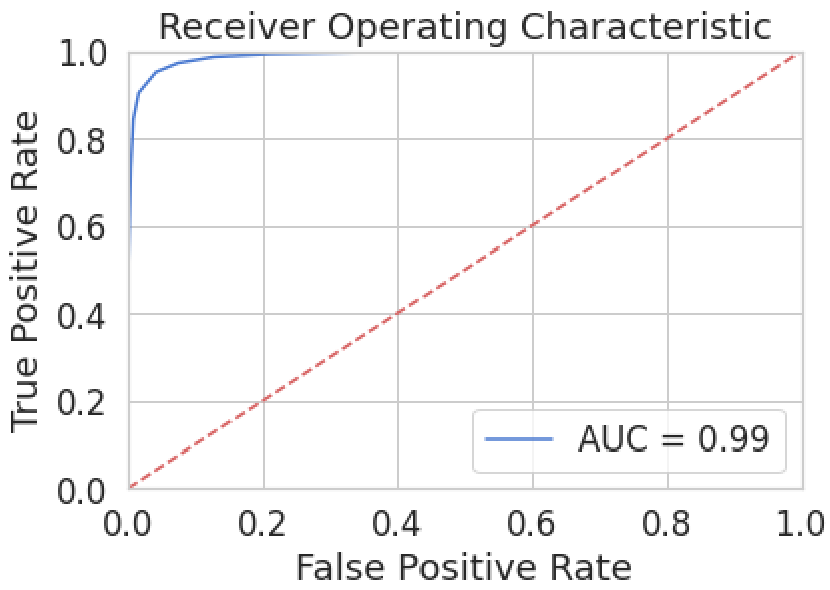

Fig. 5. Improved LSTM-PSO Model AUC-ROC Curve.

tion accuracy and the Type 1 error.

However, the overall feature selection processes enhance the performance of the classifiers on the Polish data set in terms of prediction accuracy and low error rate.

B. Discussion

1) The impact of feature selection on the classifiers: This section discusses the impact of feature selection on classification approaches based on Tables VII and VIII results.

DT: In most situations, the DT classifier outperforms the baseline model when paired with feature selection. The only exception is the application of feature selection techniques based on filters to the Australian data set. Precisely, the DT classifier performs better with the PSO-based than the filter-based method. Performing feature selection has the potential to improve DT performance. This result is consistent with the improvement in accuracy over the two data set employed. DT classifier is sensitive to feature selection.

SVM: Over the two data sets presented, the combination of the feature selection process with linear SVM does significantly outperform the baseline SVM model. However, baseline SVM produced the best Type 1 error over all the classifiers employed for the Australian data. SVM performs better when employed on filter-based than PSO-based feature

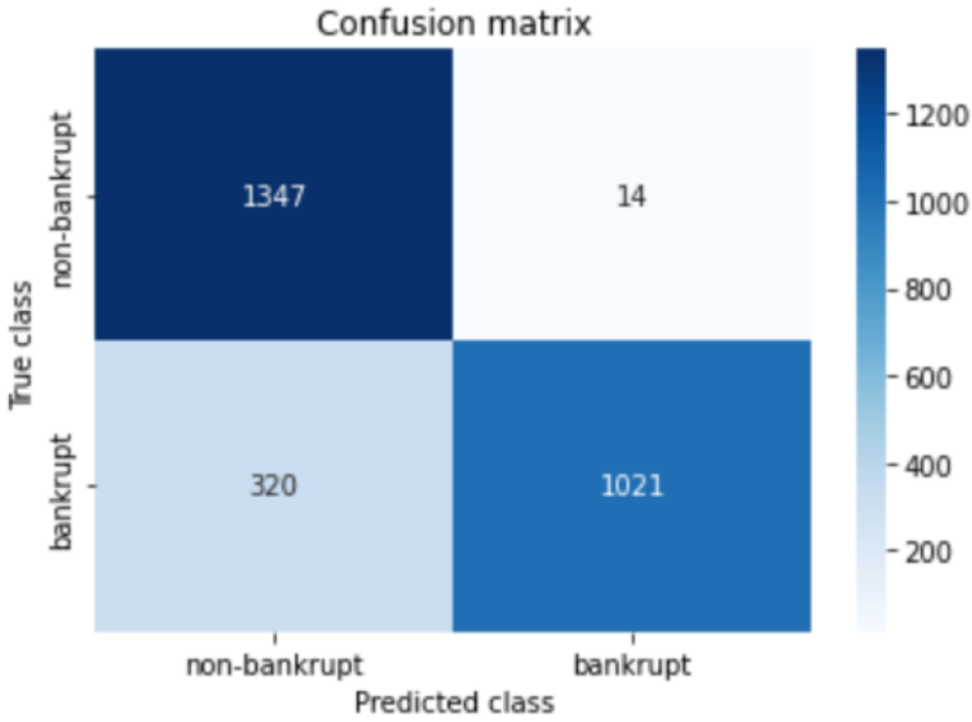

Fig. 6. Confusion matrix for the LSTM-filter model at 100% SMOTE Application.

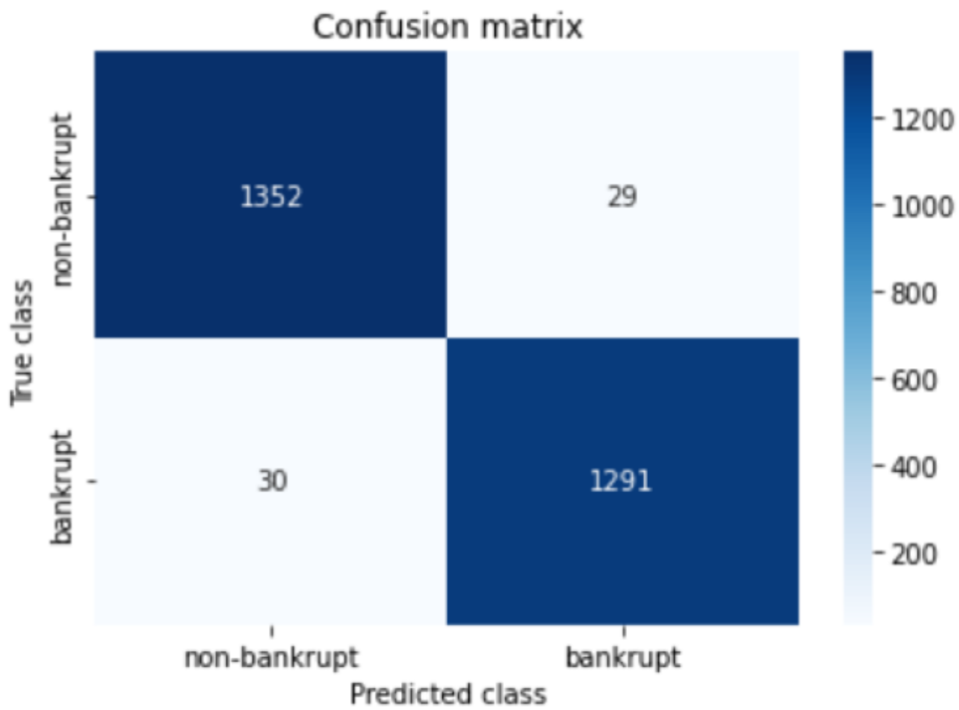

Fig. 7. Confusion matrix for the LSTM-PSO model at 100% SMOTE Application.

TABLE VII
PERFORMANCE EVALUATION OF VARIOUS CLASSIFIERS ON THE AUSTRALIAN DATA SET

Model	Metric	PSO-Feature selection (%)	Filter-Feature selection (%)	Baseline (%)
DT	Accuracy	74.25	72.08	72.38
	Type 1 error	20.98	17.50	17.49
SVM	Accuracy	70.88	71.46	70.81
	Type 1 error	15.00	18.64	13.18
MLP	Accuracy	74.03	72.24	71.66
	Type 1 error	22.53	19.45	20.44
LSTM	Accuracy	77.18	75.27	73.15
	Type 1 error	16.55	21.34	19.55
Avg.	Accuracy	74.09	72.76	72.00
	Type 1 error	18.77	19.23	17.67

TABLE VIII
PERFORMANCE EVALUATION OF VARIOUS CLASSIFIERS ON THE POLISH DATA SET

Model	Metric	PSO-Feature selection (%)	Filter-Feature selection (%)	Baseline (%)
DT	Accuracy	91.73	90.26	85.18
	Type 1 error	7.11	7.86	12.70
SVM	Accuracy	84.10	84.65	83.52
	Type 1 error	10.01	10.14	10.12
MLP	Accuracy	89.92	88.63	82.65
	Type 1 error	7.21	7.05	13.24
SVM	Accuracy	96.54	95.87	85.00
	Type 1 error	6.58	5.99	10.18
Avg.	Accuracy	90.57	89.85	84.09
	Type 1 error	7.72	7.76	11.56

selection method. The SVM classifier has some weights applied to the input features. In other words, higher weights are assigned to important input attributes. Consequently, it is essential to execute feature selection.

MLP: Over the two data sets, the combination of feature selection and MLP outperform the baseline MLP model. Therefore, the MLP performance is improved by feature selection. In MLP construction, the weights assigned to the attributes cause overfitting during the classifier training phase. Consequently, the appropriate feature selection methods will reduce overfitting and improve the model's accuracy.

LSTM: The feature selection combined with the LSTM classifier generates a good improvement over the baseline LSTM model with regard to accuracy and Type 1 error over the two data sets. The PSO-based feature selection produces the highest accuracy for both data sets. However, for Type 1 errors, the filter-based feature selection method performs best for Polish data, and PSO-based feature selection is the best for the Australian data set.

2) The best way to combine classifiers and feature selection techniques: This section discusses the ideal ways to combine the classifiers with feature selection techniques for the two data sets employed.

Australian data set: In this data set, the three best-combined feature selection and classifiers in terms of accuracy are PSO + LSTM, Filter + LSTM, and PSO + DT. On the other hand, the SVM baseline model produced the minimum Type I error, followed by PSO + SVM and PSO + LSTM. The best-combined feature selection and classifier for this data set is PSO + LSTM. Therefore, feature selection has a significant influence on this data set.

Polish data set: In this data set, the combinations that significantly upgrade the results of the prediction are PSO + LSTM, filter + LSTM, PSO + DT, and filter + DT. The two combinations that offer the maximum rates of prediction accuracy and the least error are PSO + LSTM and filter + LSTM respectively. Precisely, PSO + LSTM is the second position for the error rate. Therefore, PSO + LSTM is the ideal combination method for the Polish data.

In summary, there is no definitive solution for the optimal feature selection approach and classification strategy across the two data sets. Although, choosing the best feature selection methods enhanced the performance of the models. The average prediction outcomes for accuracy and Type I error as shown in Tables VII and VIII.

Overall, the improvements in the performance produced by PSO on Australian and Polish data sets are 2.09% and 7.25% respectively. The filter-based feature selection method generates a prediction improvement of 0.76% and 5.76% for the Australian and Polish data set respectively. These results show the significance of selecting the appropriate feature selection methods to combine with the right classifiers.

VI. CONCLUSIONS

This study presented novel feature and parameter selection methods in financial distress prediction. In our models, we show the basic idea and a prototype of employing PSO and filter gain for features and parameter selection in the LSTM algorithm for bankruptcy prediction. First, we applied the baseline machine learning algorithms such as ANN, SVM, DT and LSTM to the data set and obtained results.

The introduction of SMOTE, up-sampling technique, PSO, and information gain filter for feature selection methods enhance our models. Furthermore, we applied the learning strategies to the PSO feature selection method to improve the performance of the classifiers.

Further study would employ more feature selection methods and additional data sets to allow comparison. In addition, we would consider classifier ensembles compared with the single classifiers. It will also include the implementation of synthetic feature generation from the original features. It will allow studying the data features before the PSO algorithm for feature selection purposes.

REFERENCES

[1] A. L'heureux, K. Grolinger, H. F. Elyamany, and M. A. Capretz, "Machine Learning With Big Data: Challenges and Approaches," *IEEE Access*, vol. 5, pp. 7776–7797, 2017.

[2] M. Kovacova, T. Kliestik, K. Valaskova, P. Durana, and Z. Juhaszova, "Systematic review of variables applied in bankruptcy prediction models of Visegrad group countries," *Oeconomia Copernicana*, vol. 10, no. 4, pp. 743–772, 2019.

[3] J. Horák and V. Machová, "Comparison of neural networks and regression time series on forecasting development of US imports from the PRC," *Littera Scripta*, vol. 12, no. 1, pp. 28–42, 2019.

[4] F. Antunes, B. Ribeiro, and F. Pereira, "Probabilistic modeling and visualization for bankruptcy prediction," *Applied Soft Computing*, vol. 60, pp. 831–843, 2017.

[5] V. Machová and J. Mareček, "Estimation of the development of Czech Koruna to Chinese Yuan exchange rate using artificial neural networks," in *SHS Web of Conferences*, vol. 61, 2019.

[6] B. Gavurova, M. Packova, M. Misankova, and L. Smrcka, "Predictive potential and risks of selected bankruptcy prediction models in the Slovak business environment," *Journal of Business Economics and Management*, vol. 18, no. 6, pp. 1156–1173, 2017.

[7] M. Nakajima, "Assessing bankruptcy reform in a model with temptation and equilibrium default," *Journal of Public Economics*, vol. 145, pp. 42–64, 2017.

[8] J. Horák and T. Krulickỳ, "Comparison of exponential time series alignment and time series alignment using artificial neural networks by example of prediction of future development of stock prices of a specific company," in *SHS Web of Conferences*, vol. 61, 2019.

[9] D. Alaminos, A. Del Castillo, and M. Á. Fernández, "A global model for bankruptcy prediction," *PloS one*, vol. 11, no. 11, 2016.

[10] H. A. Alaka, L. O. Oyedele, H. A. Owolabi, V. Kumar, S. O. Ajayi, O. O. Akinade, and M. Bilal, "Systematic review of bankruptcy prediction models: Towards a framework for tool selection," *Expert Systems with Applications*, vol. 94, pp. 164–184, 2018.

[11] G. Eysenck, E. Kovalova, V. Machova, and V. Konecny, "Big data analytics processes in industrial internet of things systems: Sensing and computing technologies, machine learning techniques, and autonomous decision-making algorithms," *Journal of Self-Governance and Management Economics*, vol. 7, no. 4, pp. 28–34, 2019.

[12] F. Barboza, H. Kimura, and E. Altman, "Machine learning models and bankruptcy prediction," *Expert Systems with Applications*, vol. 83, pp. 405–417, 2017.

[13] J. Horak, J. Vrbka, and P. Suler, "Support vector machine methods and artificial neural networks used for the development of bankruptcy prediction models and their comparison," *Journal of Risk and Financial Management*, vol. 13, no. 3, p. 60, 2020.

[14] J. Vrbka and Z. Rowland, "Using artificial intelligence in company management," in *International Scientific Conference "Digital Transformation of the Economy: Challenges, Trends, New Opportunities"*, vol. 84. Springer, 2019, pp. 422–429.

[15] M. Vochozka, J. Vrbka, and P. Suler, "Bankruptcy or Success? The Effective Prediction of a Company's Financial Development Using LSTM," *Sustainability*, vol. 12, no. 18, p. 7529, 2020.

[16] G. Gilardoni, "Recurrent neural network models for financial distress prediction," *Master's thesis*, 2017.

[17] C. Zheng, S. Wang, Y. Liu, C. Liu, W. Xie, C. Fang, and S. Liu, "A Novel Equivalent Model of Active Distribution Networks Based on LSTM," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 9, pp. 2611–2624, 2019.

[18] F. Rundo, "Deep LSTM with Reinforcement Learning Layer for Financial Ttrend Prediction in FX High Frequency Trading systems," *Applied Sciences*, vol. 9, no. 20, 2019.

[19] R. Liu and L. Liu, "Predicting housing price in China based on long short-term memory incorporating modified genetic algorithm," *Soft Computing*, vol. 23, no. 22, pp. 11 829–11 838, 2019.

[20] R. Hendri, N. Sari, and A. Wibowo, "Timeseries Forecasting using Long Short-Term Memory Optimized by Multi Heuristics Algorithm," *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 8, no. 4, pp. 11 492–11 500, 2019.

[21] E. I. Altman, "Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy," *The journal of finance*, vol. 23, no. 4, pp. 589–609, 1968.

[22] W. H. Beaver, "Market prices, financial ratios, and the prediction of failure," *Journal of accounting research*, vol. 6, no. 2, pp. 179–192, 1968.

[23] W.-W. Wu, "Beyond business failure prediction," *Expert systems with applications*, vol. 37, no. 3, pp. 2371–2376, 2010.

[24] S. Shetty, M. Musa, and X. Brédart, "Bankruptcy Prediction Using Machine Learning Techniques," *Journal of Risk and Financial Management*, vol. 15, no. 1, 2022.

[25] Y. Cao, X. Liu, J. Zhai, and S. Hua, "A two-stage Bayesian network model for corporate bankruptcy prediction," *International Journal of Finance & Economics*, vol. 27, no. 1, pp. 455–472, 2022.

[26] M. Brygała, "Consumer Bankruptcy Prediction Using Balanced and Imbalanced Data," *Risks*, vol. 10, no. 2, 2022.

[27] S. Ben Jabeur, N. Stef, and P. Carmona, "Bankruptcy Prediction using the XGBoost Algorithm and Variable Importance Feature Engineering," *Computational Economics*, pp. 1–27, 2022.

[28] M. E. Pérez-Pons, J. Parra-Dominguez, G. Hernández, E. Herrera-Viedma, and J. M. Corchado, "Evaluation metrics and dimensional reduction for binary classification algorithms: a case study on bankruptcy prediction," *The Knowledge Engineering Review*, vol. 37, 2022.

[29] C. Luo, D. Wu, and D. Wu, "A deep learning approach for credit scoring using credit default swaps," *Engineering Applications of Artificial Intelligence*, vol. 65, pp. 465–470, 2017.

[30] L. Yu, R. Zhou, L. Tang, and R. Chen, "A DBN-based resampling SVM ensemble learning paradigm for credit classification with imbalanced data," *Applied Soft Computing*, vol. 69, pp. 192–202, 2018.

[31] S. P. Chatzis, V. Siakoulis, A. Petropoulos, E. Stavroulakis, and N. Vlachogiannakis, "Forecasting stock market crisis events using deep and statistical machine learning techniques," *Expert systems with applications*, vol. 112, pp. 353–371, 2018.

[32] Z. Hu, X. Si, Y. Luo, S. Tang, and F. Jian, "Speaker Recognition Based on 3DCNN-LSTM," *Engineering Letters*, vol. 29, no.2, pp463-470, 2021.

[33] A. S. Girsang and D. Tanjung, "Fast Genetic Algorithm for Long Short-Term Memory Optimization." *Engineering Letters*, vol. 30, no.2, pp528-536, 2022.

[34] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[35] N. Jiang, X. Zheng, H. Zheng, Q. Zheng *et al.*, "Long Short-Term Memory based PM2. 5 Concentration Prediction Method," *Engineering Letters*, vol. 29, no.2, pp765-774, 2021.

[36] C. M. Bishop *et al.*, *Neural Networks for Pattern Recognition*. Oxford university press, 1995.

[37] S. Gite and K. Kotecha, "Evaluating the Impact of ANN Architecture for Driver Activity Anticipation in Semi-autonomous Vehicles." *Engineering Letters*, vol. 29, no.3, pp873-880, 2021.

[38] H. Zhai and J. Che, "Combining PSO-SVR and Random Forest Based Feature Selection for Day-ahead Peak Load Forecasting." *Engineering Letters*, vol. 30, no.1, pp201-207, 2022.

[39] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*. IEEE, 1995, pp. 1942–1948 vol.4.

[40] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence (Cat. No. 98TH8360)*. IEEE, 1998, pp. 69–73.

[41] V. G. Gudise and G. K. Venayagamoorthy, "Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks," in *Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No. 03EX706)*. IEEE, 2003, pp. 110–117.

[42] Y. Zhang, S. Wang, and G. Ji, "A comprehensive survey on particle swarm optimization algorithm and its applications," *Mathematical problems in engineering*, vol. 2015, 2015.

[43] H. Ramadan, A. Bendary, and S. Nagy, "Particle swarm optimization algorithm for capacitor allocation problem in distribution systems with wind turbine generators," *International Journal of Electrical Power & Energy Systems*, vol. 84, pp. 143–152, 2017.

[44] J.-J. Kim and J.-J. Lee, "Trajectory Optimization With Particle Swarm Optimization for Manipulator Motion Planning," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 620–631, 2015.

[45] R. Ruiz-Cruz, E. N. Sanchez, F. Ornelas-Tellez, A. G. Loukianov, and R. G. Harley, "Particle Swarm Optimization for Discrete-Time Inverse Optimal Control of a Doubly Fed Induction Generator," *IEEE transactions on cybernetics*, vol. 43, no. 6, pp. 1698–1709, 2013.

[46] Y. Li, X. Bai, L. Jiao, and Y. Xue, "Partitioned-cooperative quantum-behaved particle swarm optimization based on multilevel thresholding applied to medical image segmentation," *Applied Soft Computing*, vol. 56, pp. 345–356, 2017.

[47] G. Das, P. K. Pattnaik, and S. K. Padhy, "Artificial Neural Network trained by Particle Swarm Optimization for non-linear channel equalization," *Expert Systems with Applications*, vol. 41, no. 7, pp. 3491–3496, 2014.

[48] H. Miyajima, N. Shigei, H. Miyajima, and N. Shiratori, "Securely Distributed Computation with Divided Data for Particle Swarm Optimization," in *Proceedings of the International MultiConference of Engineers and Computer Scientists (IMECS)*, 2021, pp. 1–6.

[49] E. Lu, L. Xu, Y. Li, Z. Ma, Z. Tang, and C. Luo, "A Novel Particle Swarm Optimization with Improved Learning Strategies and Its Application to Vehicle Path Planning," *Mathematical Problems in Engineering*, vol. 2019, 2019.

[50] J. A. Adisa, S. O. Ojo, P. A. Owolawi, and A. B. Pretorius, "Financial Distress Prediction: Principle Component Analysis and Artificial Neural Networks," in *2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC)*. IEEE, 2019, pp. 1–6.

[51] O. O. Olatunji, S. Akinlabi, N. Madushele, and P. A. Adedeji, "Property-based biomass feedstock grading using k-Nearest Neighbour technique," *Energy*, vol. 190, p. 116346, 2020.

[52] E. Ileberi, Y. Sun, and Z. Wang, "Performance Evaluation of Machine Learning Methods for Credit Card Fraud Detection Using SMOTE and AdaBoost," *IEEE Access*, vol. 9, pp. 165 286–165 294, 2021.

[53] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977.

[54] J. Hua, W. D. Tembe, and E. R. Dougherty, "Performance of feature-selection methods in the classification of high-dimension data," *Pattern Recognition*, vol. 42, no. 3, pp. 409–424, 2009.