

Research Article

Credit-Risk Prediction Model Using Hybrid Deep—Machine-Learning Based Algorithms

Tamiru Melese , Tesfahun Berhane , Abdu Mohammed , and Assaye Walegn 

Department of Mathematics, Bahir Dar University, Bahir Dar, Ethiopia

Correspondence should be addressed to Tesfahun Berhane; tesfahunb2002@gmail.com

Received 7 June 2023; Revised 2 October 2023; Accepted 12 October 2023; Published 6 November 2023

Academic Editor: Sadiq Hussain

Copyright © 2023 Tamiru Melese et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Credit-risk prediction is one of the challenging tasks in the banking industry. In this study, a hybrid convolutional neural network—support vector machine/random forest/decision tree (CNN—SVM/RF/DT) model has been proposed for efficient credit-risk prediction. We proposed four classifiers to develop the model. A fully connected layer with soft-max trained using an end-to-end process makes up the first classifier and by deleting the final fully connected with soft-max layer, the other three classifiers—a SVM, RF, and DT classifier stacked after the flattening layer. Different parameter values were considered and fine-tuned throughout testing to select appropriate parameters. In accordance with the experimental findings, a fully connected CNN and a hybrid CNN with SVM, DT, and RF, respectively, achieved a prediction performance of 86.70%, 98.60%, 96.90%, and 95.50%. According to the results, our suggested hybrid method exceeds the fully connected CNN in its ability to predict credit risk.

1. Introduction

Banks are financial institutions created in accordance with the rules and regulations of each country to lend, borrow, issue, exchange, receive deposits, safeguard, or handle money [1]. The banking sectors capacity to maintain a healthy influx of capital is essential to its ability to properly support a country's economic development. The risk of credit, which is exposed to a positive inflow of resources and the problem of a high-default rate in credit repayment, is one of the numerous elements that have an impact on the health of the banking sector. Credit risk, which is essentially the risk that borrowers would not be able to meet their contractual commitments and would not be able to repay bank loans on time or in whole, is the most important aspect [2, 3]. The result could be a reduction in the bank's capacity to achieve its business goals or a direct loss of capital.

Banks will have difficulty recovering disbursed credits if there is no reliable method to assess credit risk. A country's overall economic development, the banking industry, and the financial sectors are all at-risk from a declining rate of credit repayment. A nation's banking sector must successfully and consistently mobilize its financial resources for economic growth.

The ability to develop and operate an efficient credit prediction mechanism that ensures the probability of credit repayment and reduces the degree of default is thus one of the key difficulties in the banking industry. It is a difficult task for the financial sector and the banking industry to develop and implement such a mechanism because it depends on the subjective expertise of bank professionals.

The bank's credit analysts face a practical challenge when attempting to differentiate between a high-risk borrower and a credit-worthy customer who pays back the debt accurately and on time. The ability of banking institutions to accurately differentiate between a credit-worthy borrower and a delinquent one has a significant impact on both their ability to operate economically and to sustain a positive inflow of resources in the banking system of the national economy. The purpose of the credit evaluation procedure is to determine whether or not a given credit application will be able to repay the borrowed monies. And such a prediction is produced by following a few guidelines, but more importantly, by utilizing the bank's credit policies and the expertise-based knowledge of the sector experts.

In recent years, artificial intelligence has tremendously benefited the financial industry in model development and

analysis and prediction based on the historical information of the customers.

In this study, we use a hybrid approach of machine learning (ML) to predict credit risk. The aforementioned studies have done extensive research in the area of credit prediction, but to the authors' knowledge, no work has been done utilizing hybrid convolutional neural network—support vector machine/random forest/decision tree (CNN—SVM/RF/DT) algorithms. In our analysis, we consider the advantage of the convolutional layer which separates the historical credit customer data from the various borrower characteristics, and the pooling layers to minimize the size of the feature maps. The pooling layer consequently reduces the amount of computation carried out within the network as well as the number of parameters that must be derived from the credit data. The feature maps that are produced by following functional a number of convolution and pooling layers to the credit data are flattened into a 1D array and used as inputs for the SVM, RF, and DT.

In predicting a bank's credit risk, we suggest a CNN hybrid with the SVM, RF, and DT for the following reasons:

- (i) CNNs are well-known for their capacity to learn meaningful characteristics from input automatically. By adding CNNs into the hybrid models, the authors can use their feature extraction skills to detect important patterns in credit data, potentially leading to more accurate credit prediction.
- (ii) The nature of SVMs makes them effective on datasets with multiple features and efficient in situations where the number of features exceeds the number of data points. SVMs use a subset of training points in the decision function called support vectors, which reduces the memory usage.
- (iii) The nature of RF nonlinear patterns can be easily captured, variable selection can be performed more effectively, missing value prediction can be accomplished very effectively using a feature engineering algorithm, there are no distributional assumptions made, and column normalization is not required, so in this study we chose RF to hybridizes with CNN.
- (iv) The DT's characteristics include its capacity to handle missing data and predict effectively by including a subset of features at each node's splitting point.

This paper is organized into the following sections. Section 2 reviews the findings of pertinent studies that used ML and deep learning algorithms to predict credit risk. In Section 3, the preprocessing of the credit data is shown, and the techniques used—including CNN, SVM, RF, DT, hybrid CNN-SVM/RF/DT, and evaluation metrics—are thoroughly discussed. Further information about the experimental findings and a comparison of CNN hybrid with SVM, RF, and DT algorithms are covered in Section 4. In Section 5, the conclusion is delivered.

2. Related Works

Credit is one of the most crucial elements of banks and other financial institutions. Credit can also be defined as

unanticipated events that typically take the shape of either assets or liabilities [4]. The ability of the borrower to repay the credit loan on time is a key factor in determining credit risk, the primary risk that commercial banks currently face. Whether the borrower can pay back the credit loan will decide much of the problem commercial banks are currently facing [2].

Better predictive performance is linked to the use of new ML algorithms for credit default prediction; however, new model risks are also created, particularly with regard to the regulatory evaluation process. The confusion about how managers might evaluate these risks is frequently mentioned in the recent industry surveys as a potential obstacle to the invention [5]. Put out a brand-new framework to quantify model risk modifications and contrast the effectiveness of various ML techniques. They first use the internal ratings-based technique to determine up to 13-risk factors, which they then divide into three major categories: statistics, technology, and market conduct, in order to handle this difficulty. Second, using natural language processing and risk terms based on expert knowledge, they compile a number of rules and regulations pertaining to three possible application cases—regulatory capital, credit scoring, or provisioning and compute the weight of each group according to the frequency of their mentions. Finally, they put their approach to the test by quantifying certain proxies for a selection of risk components they believe to be represented using well-known ML models for credit risk and a publicly accessible database. The quantity of hyperparameters and the consistency of the forecasts are used to calculate statistical risk. The technological risk is evaluated using the algorithm's transparency and the ML training method's latency, while the market conduct risk is measured using the time it takes to run a post hoc methodology (SHapley Additive exPlanations) to interpret the results. They discover that statistical risks are more significant for regulatory capital, but risks associated with the market conduct and technology are more significant for credit scoring. Five of the most well-liked ML algorithms—RF, XGBoost, CART, Lasso, and multi-layer perceptron, were examined using their framework to assess the model risk. They can determine which of the ML models has a superior risk-adjusted performance by contrasting each model's model risk with its corresponding AUC–ROC prediction performance.

Commercial banks are very important for the growth of society and the economy. Therefore, it is crucial from both a theoretical and practical standpoint to appropriately assess their credit risk and set up a credit-risk prevention mechanism. Using a combination of a BP-neural network with a mutation genetic algorithm, [6] focuses on the credit-risk assessment of the commercial banks, uses the neural network as the primary modeling tool of the credit-risk assessment of commercial banks, and uses the mutation genetic algorithm to optimize the main parameter combination of the neural network in order to enhance the neural network's efficiency. Following the validation of several assessment models, the accuracy of the model created in their work is greater than 65%, and the evaluation outcomes improved by the mutation

genetic algorithm are more than 85% acceptable. The accuracy of the credit-risk assessment utilizing neural network technology has increased by more than 10% when compared to the accuracy of the conventional credit scoring approach, which is only approximately 50% accurate. Thus, they established that the optimized method performs better than the CNN approach. They come to the conclusion that it has significant theoretical and practical implications for the development of the commercial banks' credit-risk prevention system.

ML algorithms are applied worldwide to carry out default risk prediction in the big data era. Repetitive features and unbalanced datasets are the two key issues that can hinder the effectiveness of ML models. From the standpoint of these challenges, [7] examines the feature selection order as well as alternative balancing ratios. Thus, they first obtain 32-derived datasets with different ratios of balance and feature combinations for each dataset using data rebalancing and feature selection. In order to choose the best-derived dataset with the ideal balance ratio and feature combination, they secondly offer a comprehensive metric model based on multi-ML algorithms. Their study makes two contributions. The ideal balance ratio is initially identified through classification accuracy in order to address the issue in the prior research that samples are imbalanced or the balance ratio is 1:1. This guarantees the classification model's accuracy as well. Second, they suggested a complete metric model built on a ML algorithm, which can concurrently select the best features and determine the ideal balance ratio. Their experimental outcomes show that their strategy might greatly improve CNN's performance, and CNN outperforms the other four popular ML models on four benchmark datasets for predicting default risk.

Financial crises are very likely to be brought on by poor decision-making in the financial institutions. Numerous studies conducted recently have shown that artificial intelligence systems can be employed as alternatives to traditional credit rating methodologies [8], developed a prediction model for credit approval that combines feature selection, instance selection, and classifiers. The first step in feature selection is the usage of a measure (gain ratio). In order to cluster the training dataset into k -clusters beforehand, a clustering algorithm (EM) is used. Finally, k -DT classifiers for k -clusters of examples are constructed using the C4.5 technique. The EM clustering approach is used to decide which cluster-based DT should be employed in order to forecast the class labels of previously unknown records. Their proposed CBDT strategy is superior to the other five methods (DT, MLP, NB, RF, and SVM) in the measures (F1, Accuracy, and CostEffect), according to the experimental findings produced using the survey data. Additionally, they suggested three methods (MLP, NB, and SVM) be combined with two hybrid approaches (feature selection and instance selection) to create three new ways (CBMLP, CBNB, and CBSVM). They claim that their proposed CBDT methodology outperforms the other five approaches (DT, MLP, NB, RF, and SVM) in the measurements (F1, Accuracy, and CostEffect) of their

experiments, which were conducted using the survey data. In addition, they suggested creating three new methods (CBMLP, CBNB, and CBSVM) by combining two hybrid approaches (feature selection and instance selection) with three existing methods (MLP, NB, and SVM). From their experimental findings, they recommend that managers of the banking and auditing sectors might take into account hybrid ideas, feature, and instance selections, when establishing their information systems for credit approval. As a result, this type of technology can assess client dependability with greater precision, significantly lowering the cost of bad debts.

The aforementioned studies have done extensive research in the area of credit prediction, but to the authors' knowledge, no work has been done utilizing hybrid CNN-SVM, CNN-RF, and CNN-DT algorithms. For the purpose of predicting credit risk, we create a hybrid ML algorithm and compare it against a fully connected CNN. We also discuss how each model's various learning parameters affect it.

3. Materials and Methods

3.1. Dataset. We implement our proposed model for predicting credit risk using data from a local bank in Ethiopia. The credit data comprises 7,631 examples of creditworthy applicants who are nondefaulter and 6,823 applicants' of defaulter.

The input features used to predict credit risk are summarized in Table 1 below.

Table 2 summarizes the numerical input features used in this model. The distribution of all the numerical inputs is not symmetrical, which shows that the mean and the median of all the numerical input features are different. A second quartile value of 39 for age implies that 50% of customers are 39 years old, while the remaining 50% are more than 39 years old. The age distribution is positively skewed, indicating that most consumers are younger than the average and that a high number of customers are forced to the right side of the distribution. In addition, age is a positive excess kurtosis which shows age distribution is leptokurtic (heavy tails on either side), indicating high outliers. Total income is negatively skewed, which shows most customers' total income is lower than the mean value.

Missing values are frequently observed during data collection. Missing values reduce the amount of data that can be processed, which lowers the study's statistical power and, eventually, lowers its ability to draw findings that are trustworthy. Additionally, it makes the data less useful and significantly biases the outcomes. In order to overcome these difficulties, we employ several methods for addressing missing values depending on the variables. Remove the rows with missing values if there are only a few records, fill in the missing values for continuous variables using the median value, and fill in the missing values for categorical variables using the mode value.

The input data, which are the numerical values of the attributes, are separately normalized to range from "0" to "1". Finding the highest or maximum value for each attribute across all dataset cases can help you achieve this. Divide

TABLE 1: Inputs and their representation.

Inputs	Representation of inputs
Age	Age of customer
Gender	Gender of customer
Marital status	Customer's marital status: married: 0 and not married: 1, others: 2
Job	Job of customers
Education	Diploma: 0, degree holder: 1, second degree: 2, Ph.D.: 3
work experience	<5, 5–10, 11–15, 16–20, 21–25 and >25
Type of credit	Mortgage, auto, personal, education, agriculture, medical
Annual income	Annual income of the customers
Coapplicant income	The annual income of the customers coapplicant
Total income	The sum of annual income and the coapplicant annual income
Duration of credit	0–60, 61–120, 121–180, 181–240, and 241–300
Collateral	Physical asset: 0, salary: 1
Credit history	0, 1, 2, 3, 4, and 5
Credit grade	A: 0, B: 1, C: 2, and D: 3
Amount of credit	The amount of credit based on the total income of the customers
Credit interest rate	13%, 11%, and 7%

TABLE 2: Numerical input features.

Numerical variables	Mean	Median	SD	Q1	Q2	Q3	Skewness	Kurtosis
Age	40.1967	24	5.9991	36	39	44	0.7178	0.5217
Annual income	4.7499	4.7782	0.3383	4.5017	4.7782	4.9542	−0.1607	−0.5815
Co-applicant income	3.5614	3.3979	0.4048	3.2553	3.3979	4	0.3688	−0.7050
Total income	4.7996	4.8351	0.3060	4.5723	4.8351	4.9845	−0.0883	0.4911

Note. SD, standard deviation Q_1 , 1st quartile, Q_2 , 2nd quartile, Q_3 , 3rd quartile.

the highest value by this number to get all other values contained within that attribute.

In our study, we divided the dataset into two parts: the training dataset and testing dataset. We tested with a range of learning techniques and training-to-testing data ratios in order to identify the ideal training-to-testing ratio for the application of the credit-risk prediction model. In our proposed model we used search approach, in which, different parameter values were considered and fine-tuned through-out testing to select appropriate parameters by exhaustively searching through all possible combinations.

3.2. Convolutional Neural Network. The CNN, also known as ConvNet, is a special type of neural network that has a deep feed-forward architecture and amazing generalizing abilities compared to the other networks with FC layers [9]. It can learn highly abstracted features of objects, particularly spatial data, and can identify them more effectively. It is a special kind of feed-forward neural network [10], which utilizes convolution, ReLU, and pooling layers [11]. Every convolution layer comprises of many kernels of the same size, and they are all used to extract features from the input [12]. The pool layer comes next, performing (average or max) pooling, and sending the output to the prediction or fully connected layer [12]. There are multiple layers (multibuilding blocks) in the CNN architecture. Below, we go into further detail about the function of each layer in the CNN architecture [13].

3.2.1. Convolutional Layer. The most crucial component of the CNN architecture is the convolutional layer. It is made up of a set of convolutional filters (kernels), a feature map, and input data. Convolution requires a number of spatial variables, such as the size of the kernels (N), stride (S), and padding (P), to produce activation maps of a particular size. The kernel will run a convolution operation on a portion of the input that corresponds to the size of its window in order to produce results in its activation map [11]. The stride, or the distance between two subsequent kernel points, serves as the basis for the convolution operation [14]. It controls and adjusts the amount of movement across the input data. Zero padding is the practice of encircling a matrix with zeroes (adding zeroes to the input boundaries), which helps to maintain features that are present at the margins of the original matrix and to regulate the size of the output feature map. Convolutional filters are applied to 1D input using a 1D convolutional layer. By moving the filters along the input, computing the dot product of the weights and the input, and then adding a bias term, the layer convolves the input [15]. Figure 1 shows the functionality of 1D convolutional layer. Sliding convolutional filters are applied to 1-D input via a 1-D convolutional layer. By moving the filters along the input, computing the dot product of the weights and the input, and then adding a bias component. The input layer's (x_1, x_2, \dots, x_{17}) and outputs' (y_1, y_2) strides will all have the same weights w_1, w_2, w_3 because the kernel size is set to 3.

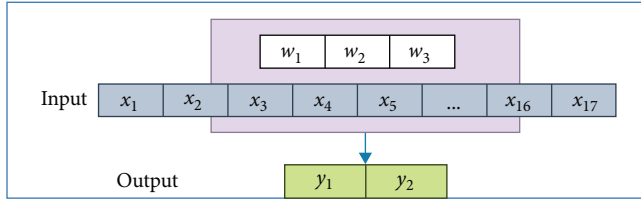


FIGURE 1: 1D convolution layer.

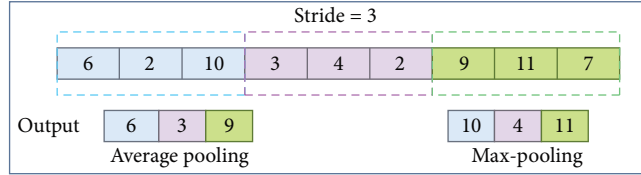


FIGURE 2: Max pooling 1D.

Weights are added to the input values in the kernel window (w_1, w_2, w_3).

3.2.2. Pooling Layer. A pooling layer is frequently interspersed between subsequent convolutional layers in a CNN architecture [10, 16]. The pooling layer can minimize the dimension of the feature map by sliding a filter of a particular size with a certain stride size and computing the maximum or average of the input, which speeds up the computation by lowering the number of trainable parameters [17]. The 1D max pooling block computes the maximum in each distinct window by moving a pool (window) of a defined size over the incoming data with a set stride. To provide the inner convolutional layers access to more of the original vector's information, max-pooling layers are added after one or more convolutional layers. If convolutional layers are viewed as feature detectors, max pooling only keeps the "strongest" value of a feature inside the pooling rectangle. A 1D average pooling layer separates the input into 1D pooling regions and then computes the average of each region to perform down sampling. The layer pools the input by moving the pooling regions along the input horizontally. Figure 2 illustrates the computation of the maximum and average pooling layers. By pooling layers, one can reduce the size of the feature maps, as shown in Figure 2. The amount of network computation and the number of parameters that need to be learned are thereby decreased. Additionally, it provides a summary of the characteristics found in a particular area of a feature map created by a convolution layer. The pooling size prior to pooling is three, while the map size is nine. After reducing the output values to three, the maximum values for the following network layer will be chosen. Max pooling assists in lowering over-fitting by providing an abstracted form of representation.

3.2.3. Dropout Layer. A dense layer or completely connected layer can be used as an input after flattening all of the pooled feature maps into a single vector. The flatten function reduces the multidimensional input tensors to a single dimension, allowing you to effectively model your input layer and construct your

neural network model before feeding those data to each and every model neuron.

3.2.4. Fully Connected Layer. CNNs, which have had great success in the financial sector, feature a very unique and important component called the fully connected layer. Convolution and pooling are the first steps in the CNN method, which divides the input into a vector of features and then analyzes each feature individually. The process results in a completely integrated final decision. A single vector is created by reshaping (flattening) the output of the network's preceding levels. They each represent the likelihood that a specific feature is a class label. The feature map is used as an input, and weights are used to determine the proper label. The final probabilities for each label are provided by FC's output layer.

3.3. Support Vector Machine. SVM, the most effective supervised ML algorithm [18], is used for classification, regression, and outlier detection. A SVM performs classification by constructing an N -dimensional hyper plane that optimally separates the data into two categories [19, 20]. A margin is a distance between any two lines on the closest data points, and this closest data point to the hyperplane is the support vectors. The margin is determined as the perpendicular distance between the line and the nearest data points or support vectors [21, 22].

Consider the credit customers dataset, there are two classes of target variables in \mathbb{R}^n : nondefaulter and defaulter. Assume that the nondefaulter and defaulter customer's classes are linearly separable, which means that a hyperplane $\{x \in \mathbb{R}^n : w^T x + b = 0\}$ exists such that the nondefaulter belongs to one half-space and the defaulter belongs to the other [23].

To be more specific, there exists a vector $w \in \mathbb{R}$ and a scalar $b \in \mathbb{R}$ such that

$$\begin{cases} w \cdot x_i + b \geq 1, & \text{for } y_i = +1 \\ w \cdot x_i + b < -1, & \text{for } y_i = -1 \end{cases} \quad (1)$$

We need to find a hyperplane which has a maximum margin for $w \in \mathbb{R}^n$ and offset scalar b for the prediction of the credit customer's as nondefaulter and defaulter such that

$$\begin{aligned} & \underset{w, b}{\text{minimize}} && \phi(w) = \frac{1}{2} w^2 \\ & \text{subject to} && y_i(x_i \cdot w + b) \geq 1, \quad \text{for } i = 1, 2, \dots, n. \end{aligned} \quad (2)$$

Due to the complexity of the constraints, it is challenging to directly solve this problem. The Lagrangian duality theory is the preferred mathematical technique for deriving a solution to this problem [24, 25]. The goal is to identify a maximum margin hyperplane that aids in appropriately predicting credit risk depending on the classes, either defaulter or non-defaulter. A Lagrange multiplier (α) can be used in these circumstances to enforce the constraint, as shown below:

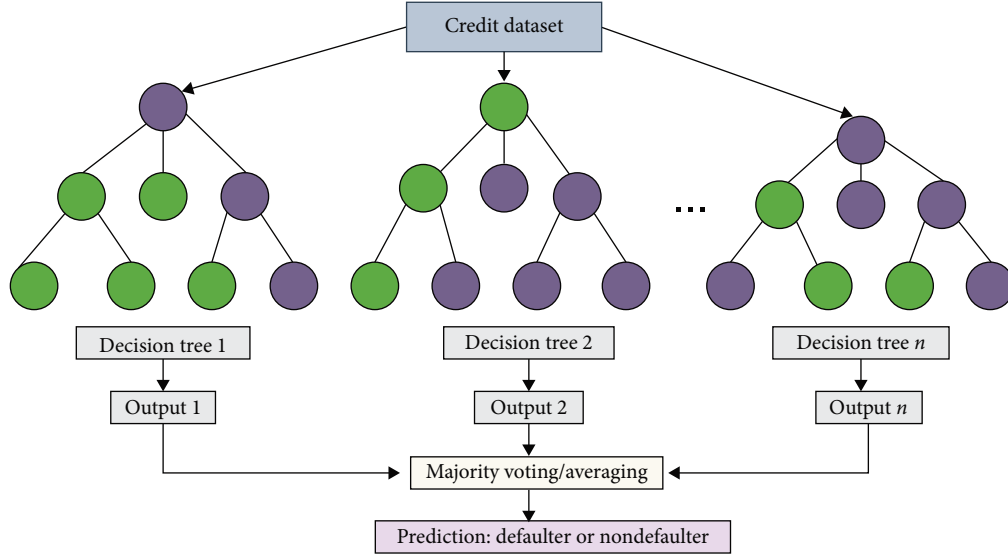


FIGURE 3: Architecture of random forest.

$$L(w, b, \alpha) = \frac{1}{2}w \cdot w - \sum_{i=1}^m \alpha_i (y_i (w \cdot x_i + b) - 1). \quad (3)$$

$$\begin{aligned} \mathcal{L}(w, b, \alpha) &= \sum_{i=1}^m \alpha_i - \sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j x_i x_j \\ &\text{subject to } \sum_{i=1}^m \alpha_i y_i, \text{ and } 0 \leq \alpha_i \leq C, \end{aligned} \quad (7)$$

In order to determine the stationary (saddle) point of Equation (3), the following conditions must be satisfied:

$$\frac{\partial L}{\partial w} = 0 \implies w = \sum_{i=1}^N \alpha_i x_i x_j, \quad (4)$$

$$\frac{\partial L}{\partial b} = 0 \implies \sum_{i=1}^N \alpha_i y_i = 0. \quad (5)$$

It should be emphasized that α won't be equal to zero unless the accompanying input data, x_i , is a support vector [26, 27]. The general equations of the SVM are constructed by substituting Equations (4) and (5) into Equation (3). These results in the maximum hyperplane required to predict the two classes as either nondefaulter or defaulter using a linearly separable credit data.

$$\begin{aligned} \mathcal{L}(w, b, \alpha) &= \sum_{i=1}^m \alpha_i - \sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j x_i x_j \\ &\text{subject to } \sum_{i=1}^m \alpha_i y_i, \forall \alpha_i \geq 0. \end{aligned} \quad (6)$$

We assume that the nondefaulter and defaulter classes of our target variable in the credit dataset are not linearly separable. In other words, using Equation (1), it may be difficult to separate data linearly because of the homogeneity of a few features in the credit dataset. To solve that problem, let's utilize the slack variables ξ_i and the solution of Equation (3) will be

where $C > 0$ is the penalty parameter of the error term. It handles the tradeoff between accurately identifying the training points and smooth decision boundaries.

3.4. Random Forest. The RF algorithm creates a forest in the shape of a collection of DTs, increasing randomization as the trees grow. The technique seeks for the best features from the random subset of features when splitting a node, adding more diversity and improving the model. RF is an ensemble of T DTs, which is typically created by building many DTs in order to improve performance [28]. Through the use of tree ensembles and random selection of input variables, RF can minimize overfitting issues and increase variety [29].

The RF classification method is demonstrated in Figure 3 in the following ways [30, 31]: (i) using the initial data, create n_{tree} bootstrap samples. (ii) Grow an unpruned classification tree for each of the bootstrap samples with the following modification: at each node, select the best split among the m_{try} predictors instead of the best split among all predictors. (iii) By aggregating the n_{tree} trees' predictions (i.e., majority votes for classification), predict new data. Due to the nature of handling missing data, predicting effectively without hyperparameter tuning, solves overfitting problems in the DFs, and at the node's splitting point in every RF tree, a subset of feature, in this study we chosen RF to hybrid with the 1D-CNN.

3.5. Decision Tree. One of the most used supervised learning algorithms is the DF, which works by building a training model that can be used to predict the class or value of the target variable by learning straightforward decision rules inferred from historical data (training data) [32, 33]. The

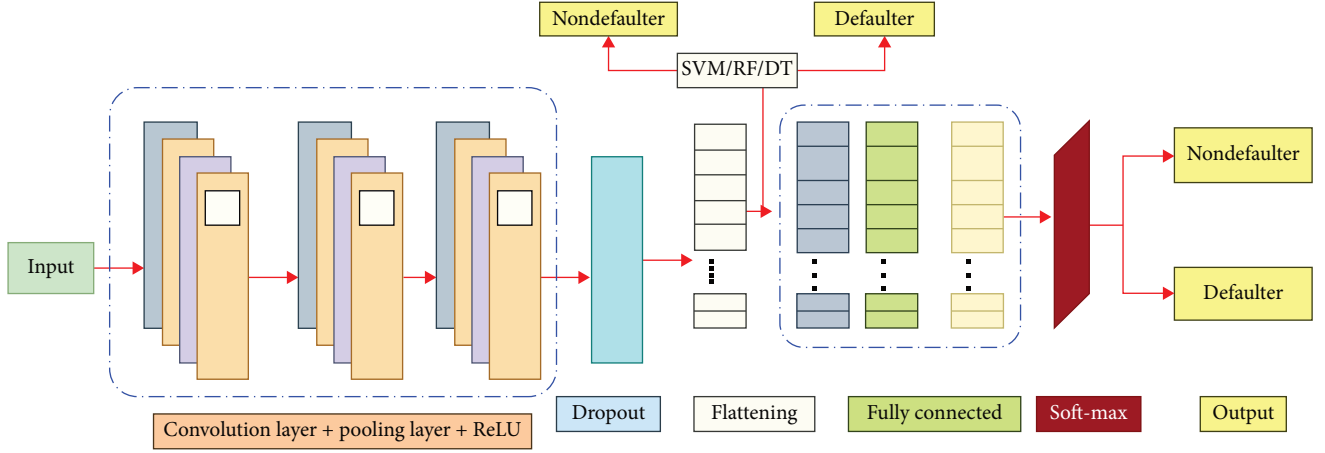


FIGURE 4: Architecture of the model: this technique uses four classifiers. The first classifier is a fully connected layer with soft-max that is trained using an end-to-end process, whereas the other three classifiers are binary SVM/RF/DT classifier that is piled on top by removing the final fully connected and soft-max layer. The feature maps that are created by applying a variety of convolutional and pooling layers to the credit data are flattened into a 1D array and utilized as inputs for the support vector machine, random forest, and decision tree.

primary objective in DTs is choosing the best attribute from the dataset complete features list for both the root node and the subnodes. The two methods for evaluating attribute selection are information gain and Gini index.

Entropy, which is used to train DTs, is essentially the variance of the data. Information entropy for a credit dataset with two classes, which are defaulter and nondefaulter is given by

$$E = -\sum_{i=1}^2 P_i \log_2 P_i. \quad (8)$$

where p_i is the probability of randomly picking an element of class i .

Information gain is a metric used to describe changes in entropy value following the splitting or segmentation of the dataset based on an attribute. It explains how much information a feature or characteristic gives us, and based on the knowledge gathered, node splitting, and DT building are carried out. A node or attribute with the highest value of the information gain is split first in the DT, which always aims to maximize the value of the information gain.

The Gini index is used as a metric of impurity or purity when developing a DT using a classification method [34]. A low-Gini index value should be preferred over a high-Gini index value if possible. Only binary splits are produced by it, and the Gini index is used in the classification process to produce those binary divides.

3.6. Hybrid CNN-Machine Learning Algorithms. 1D convolution layers, pooling layers, dropout layers, and activation functions are used by CNN to handle the 1D data. A CNN's convolutional layer, which forms its central component, is also where most processing takes place. It calls for input data, a filter, and a feature map among other things. Convolution layer-based filtering is done to the input. As a result of using the filtering procedure repeatedly, a feature map that displays the specific properties associated with the data points is

produced. Convolution is a linear process where the input multiplication is controlled by a set of weights. The weight often referred to as kernels, of the 1D array is multiplied by the inputs. This process creates a feature map by producing many values, each with a unique value for each iteration.

In this study, a hybrid CNN/SVM/RF/DT model has been proposed to predict credit risks. The important characteristics of both classifiers are combined in the suggested hybrid model. CNN functions automatically as a feature extractor, whereas ML algorithms function as a classifier [35]. Due to this, a CNN is used by both ML algorithms and fully connected networks because of their automatic feature extraction capabilities. Therefore, no additional feature extraction or selection stages are required prior to the SVM/RF/DT-based classification [29].

The proposed approach considers four classifiers, as shown in Figure 4. A fully connected layer with soft-max that was trained using an end-to-end scenario makes up the first classifier. Soft-max helps to convert a vector of numbers into a vector of probabilities, where the probability of each value is proportional to the relative scale of each value in the vector. For an input vectors x and output a vector of probability P either defaulter or nondefaulter classes through a soft-max function at the end of the model architecture is given by

$$P_i = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}, \quad (9)$$

where j is the index and n is the number of outputs.

Based on the characteristics of the CNN and ML methods such as SVM, RF, and DT, we suggest two stages to construct a credit-risk prediction. In the first stage, features from the original credit customer's data are extracted using the CNN's impressive feature extraction characteristics to create a new feature matrix. Second, SVM/RF/DT algorithms

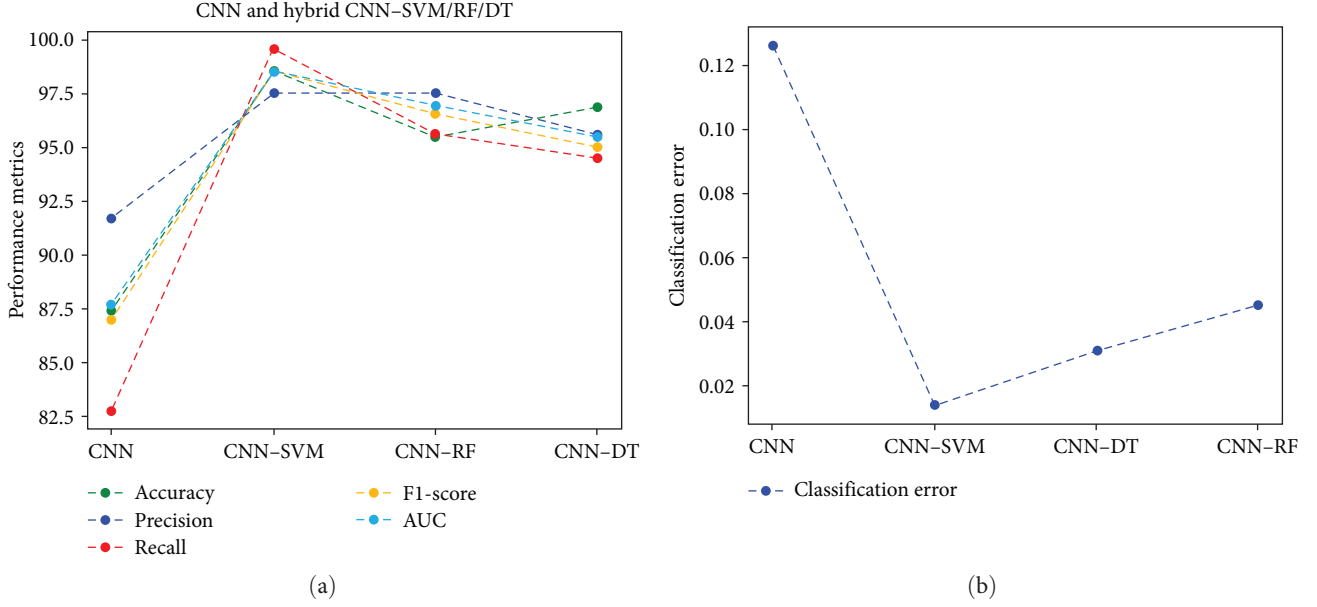


FIGURE 5: Model performance and classification error: (a) model performance of CNN and hybrid CNN-SVM/RF/DT and (b) classification error.

are utilized to build the prediction model utilizing the updated feature matrix as input data.

The convolutional layer separates the historical credit customer data from the various borrowers attributes, and the pooling layers minimize the feature maps' size. The pooling layer thereby reduces both the number of parameters that must be calculated from the historical credit data and the amount of computation required within the network. After the historical credit data are flattened into a 1D array, and the feature maps formed by iteratively applying multiple convolutions and pooling layers serve as the inputs for the SVM, RF, and DT. Hence, in the architecture of our suggested models, SVM/RF/DT takes the position of the fully connected and soft-max layer of a CNN [16, 35–39].

3.7. Performance Evaluation Metrics. The accuracy, precision, sensitivity, and specificity numbers are all included in a classification analysis along with the F1 score [40]. The accuracy of a model is one way to measure the number of correctly predicted data points out of all the data points [28, 41].

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \cdot \quad (10)$$

Precision is the proportion of accurately predicted positive outcomes out of all predicted positive outcomes. It can be expressed as the proportion of true positives (TP) to the total of true and false positives (TP + FP) [42].

$$\text{Precision} = \frac{TP}{TP + FP} \cdot \quad (11)$$

Recall also called sensitivity is a proportion of accurately predicted positive outcomes among all actual positive

outcomes. The ratio of true positives (TP) to the sum of true positives and false negatives (TP + FN), which helps to identify the proportion of correctly predicted actual positives.

$$\text{Recall} = \frac{TP}{TP + FN} \cdot \quad (12)$$

The F1 score, a ML evaluation metric, rates a model's accuracy. It incorporates a model's recall and precision values. The accuracy statistic shows how often a model predicts accurately over the entire dataset [43].

$$F1 - \text{Score} = \frac{2TP}{2TP + FP + FN} \cdot \quad (13)$$

4. Results and Discussion

Different values of the parameters were taken into consideration and fine-tuned throughout testing in order to choose appropriate parameters for both a CNN fully connected layer and SVM, RF, and DT model.

We run four experiments. In the first experiment, a fully connected CNN served as a feature extractor and a fully connected network with soft-max served as a classifier. We used the Adam optimizer, ReLU activation function, and max-pooling during training, and after 25 epochs, the network converges to the desired results. A SVM, RF, and DT classifier were used in the other three tests. The input feature created by applying several convolutional and pooling layers to the credit data is flattened into a 1D and utilized as an input for the SVM, RF, and DT. We used a radial basis function kernel with $C = 100$ and $\gamma = 0.8$ to train a hybrid CNN-SVM. We applied the n -estimators = 100,

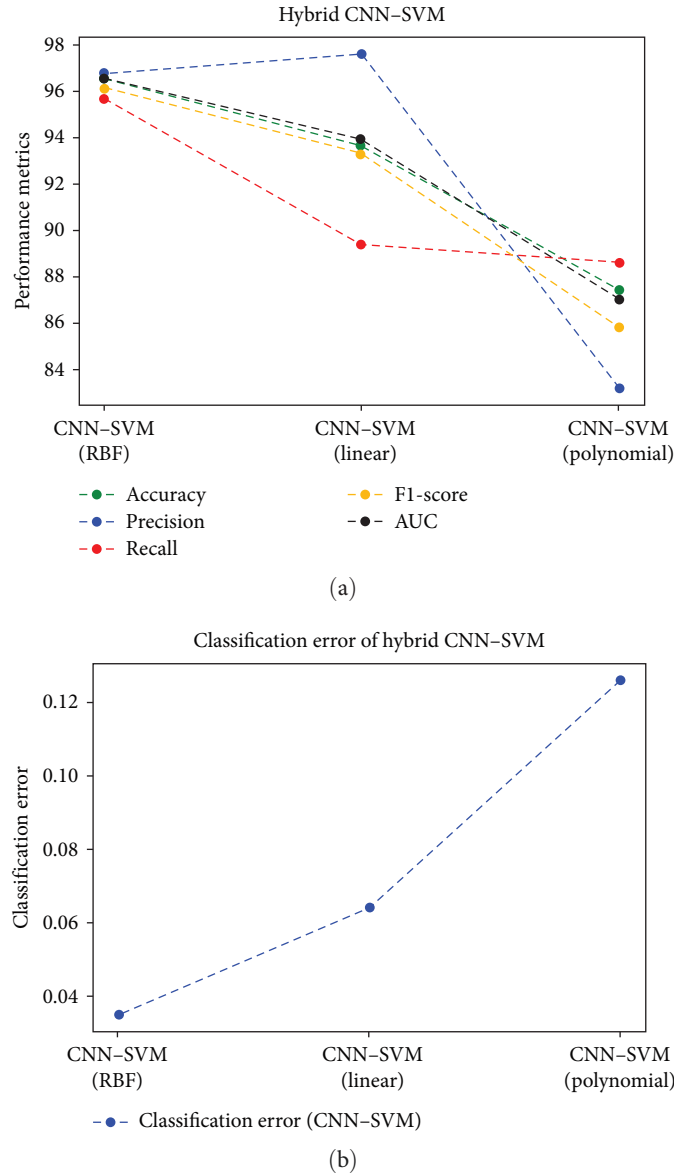


FIGURE 6: Performance and classification error of hybrid CNN-SVM: (a) model performance of hybrid CNN-SVM and (b) hybrid CNN-SVM classification error.

max-depth 5, and Gini criterion to train the hybrid CNN-RF and CNN-DT models.

The accuracy, precision, and recall of the suggested model, which is calculated as the proportion of correctly identified defaulter and nondefaulter credit applicants, is used to assess its performance. According to the results shown below in Table 2, our proposed hybrid CNN-ML algorithms exceed the fully connected network in its ability to classify credit applicants as defaulter or nondefaulter with an overall accuracy of 98.60%, 95.50%, and 96.90% when employing SVM, RF, and DT, respectively. Based on the precision that is the potential of accurately predicted positive outcomes out of all predicted positive outcomes CNN-RF exceeds the CNN-SVM and CNN.

Figures 5(a) and 5(b), Figures 6(a) and 6(b), Figures 7(a) and 7(b), and Figures 8(a) and 8(b) display the CNN model's

performance and classification error, as well as the hybrid CNN-SVM, CNN-RF, and CNN-DT models. These models were evaluated using an 80 : 20 learning method with varied parameter settings for each algorithm.

The CNN architecture is combined with a SVM in the hybrid model, which uses a radial basis function kernel. The SVM parameters include a regularization parameter (C) set at 100.

The CNN architecture is integrated with the RF and DT algorithms in the hybrid model. The RF and DT are configured with the entropy criterion for splitting, a maximum depth of 5 and 100 estimators.

Table 3 shows the results of a credit-risk prediction experiment employing a hybrid CNN-SVM model. The analysis included a variety of kernels, including linear, polynomial, and radial basis functions (RBF). The experimental

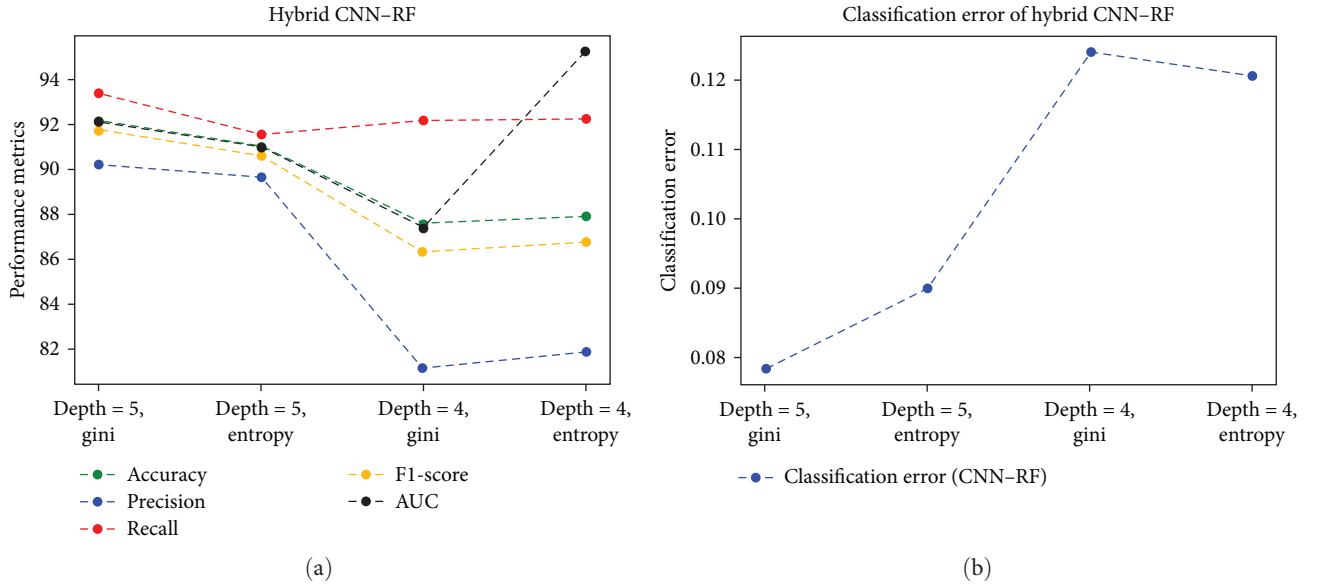


FIGURE 7: Performance and classification error of hybrid CNN-RF: (a) model performance of hybrid CNN-RF and (b) hybrid CNN-RF classification error. Hybrid CNN-RF classification error.

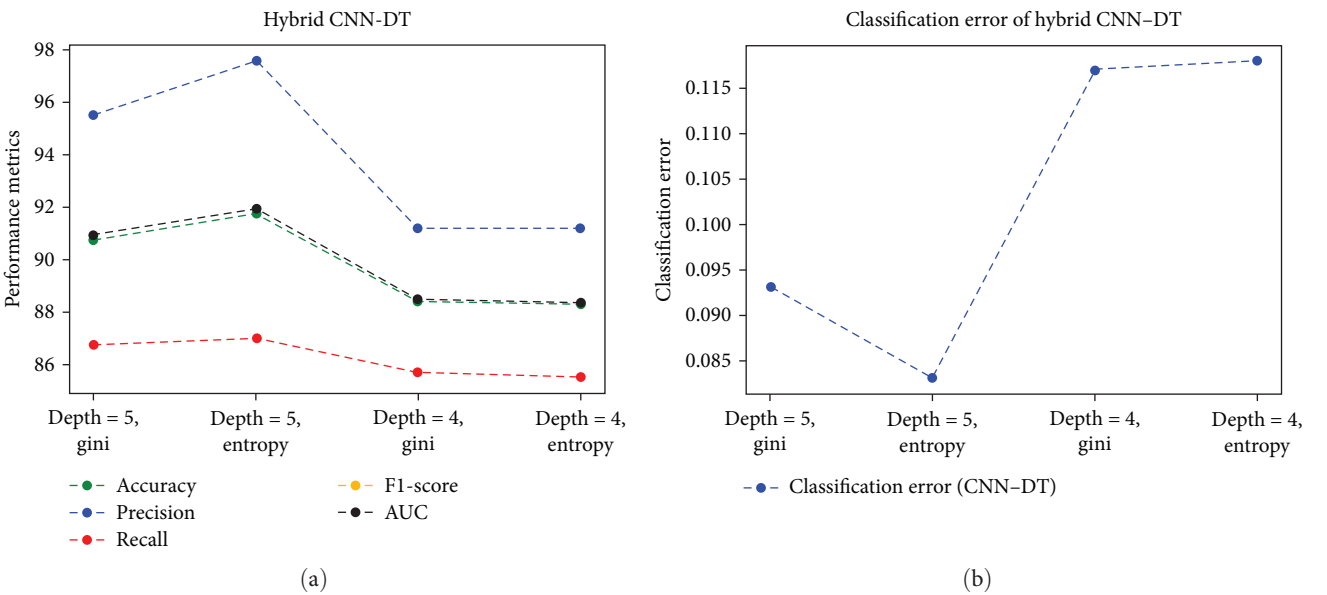


FIGURE 8: Performance and classification error of hybrid CNN-DT: (a) model performance of hybrid CNN-DT and (b) hybrid CNN-DT classification error.

TABLE 3: Parameters used to 1D-CNN.

Parameters	Activation function	Optimizer	Dropout	Epoch	Batch size	Convolution kernel size
Values	ReLU	Adam	0.5	25	16	32/64/128

results show that both RBF and linear kernels are effective at predicting credit risk. Notably, the hybrid CNN-SVM obtained 96.5% accuracy. This result was obtained by using the radial basis function with specified parameters, specifically a regularization parameter (C) of 70, a gamma value of 0.7, and an 80 : 20 learning scheme.

The performance and classification error of a hybrid CNN-SVM with an 80 : 20 learning scheme, RBF, linear, and polynomial kernels, regularization parameter $C = 70$, and gamma = 0.7 are shown in Figure 6.

Table 4 provides insights into the performance evaluation of the hybrid CNN-RF prediction model using various

TABLE 4: Performance of CNN and hybrid CNN—SVM/RF/DT (%).

Performance metrics	CNN	CNN-SVM	CNN-DT	CNN-RF
Accuracy	86.70	98.60	96.90	95.50
Precision	91.74	97.49	95.57	97.56
Recall	82.75	99.57	94.52	95.65
F1-score	87.01	98.52	96.59	95.04
AUC	87.72	98.55	95.51	96.96

Learning scheme 80 : 20, support vector machine with radial basis function and $C = 100$, random forest and decision tree with criterion entropy, max-depth = 5, and n -estimators = 100.

TABLE 5: Performance of hybrid CNN-SVM (%) with different kernel tricks.

Proposed methods	Accuracy	Precision	Recall	F1-score	AUC
CNN-SVM (kernel = RBF)	96.50	96.72	95.68	96.19	96.52
CNN-SVM (kernel = linear)	93.60	97.60	89.40	93.32	93.91
CNN-SVM (kernel = polynomial)	87.40	83.19	88.60	85.81	87.07

Note. $C = 70$, gamma = 0.7, and learning scheme 80 : 20.

TABLE 6: Performance of hybrid CNN-RF with learning scheme = 80 : 20.

Proposed methods	Criterion	Accuracy	Precision	Recall	F1-score	AUC
CNN-RF (max-depth = 5)	Gini	92.17	90.16	93.38	91.74	92.10
CNN-RF (max-depth = 5)	Entropy	91.00	89.64	91.53	90.58	90.95
CNN-RF (max-depth = 4)	Gini	87.58	81.17	92.16	86.32	87.37
CNN-RF (max-depth = 4)	Entropy	87.92	81.87	92.22	86.74	95.19

Note. n -Estimators = 80, different criterion, and different max-depth.

TABLE 7: Performance of hybrid CNN-DT with learning scheme = 80 : 20.

Proposed methods	Criterion	Accuracy	Precision	Recall	F1-score	AUC
CNN-DT (max-depth = 5)	Gini	91.70	97.53	86.95	91.94	91.87
CNN-DT (max-depth = 5)	Entropy	90.70	95.46	86.70	90.87	90.84
CNN-DT (max-depth = 4)	Gini	88.30	91.13	85.66	88.31	88.38
CNN-DT (max-depth = 4)	Entropy	88.20	91.13	85.49	88.22	88.29

Note. n -Estimators = 80, different criterion, and different max-depth.

maximum depths and criteria. The reported accuracy of the hybrid CNN-RF model is 92.17%, achieved with a learning scheme of 80 : 20, 80 estimators, the Gini criterion, and a maximum depth of 5.

The data show that increasing the maximum depth from 4 to 5 and changing the criterion from entropy to Gini improves all performance measures, resulting in improved outcomes.

The increased maximum depth enables the RF component's DTs to grow deeper, capturing more complicated relationships and potentially boosting the model's capacity to discern credit occurrences. The model can capture finer patterns and make more sophisticated predictions with a deeper tree structure.

Changing the criterion from entropy to Gini also improves performance. Both entropy and Gini are impurity metrics used to assess the quality of splits in a RF. The model emphasizes impurity minimization based on Gini impurity, which may

correlate better with the characteristics of the credit prediction task when employing the Gini criterion. This criterion update enables the model to generate more relevant splits while improving overall accuracy.

According to the stated accuracy of 92.17%, the hybrid CNN-RF model works well in credit prediction. The model performs considerably better when the maximum depth is increased from 4 to 5 and the Gini criterion is used. These findings emphasize the significance of parameter adjustment and the impact it might have on the hybrid CNN-RF model's predictive capabilities.

The performance of a hybrid CNN-DT with an 80 : 20 learning scheme, 80 estimators, and various criteria is shown in Table 5 below. Accuracy, precision, recall, F1-score, and AUC performance indicators for the hybrid CNN-DT improve when the Gini criterion is used, and the maximum depth is increased from 4 to 5. As a result, increasing the depth enhances the performance of the model.

Tables 6 and 7 illustrate the performance and classification error of hybrid CNN-RF and CNN-DT with a learning scheme of 80:20, the Gini and entropy criterion, max-depth = 5(4), and n -estimators = 80. When we use the Gini criterion and increase the maximum depth from 4 to 5, the classification error is reduced. As a result, applying the Gini criterion and increasing the depth help the model perform better.

Increasing the DT's maximum depth enables for more complicated and comprehensive decision limits. The model can capture more information and customize the categorization process to the individual properties of the credit data using a deeper tree. This additional complexity allows the model to make finer differences and potentially improve its effectiveness in credit instance classification.

5. Conclusions

Despite the rise in credit needs and the level of competition in the banking sector, the majority of banks has been reluctant to use machine and deep learning algorithms to minimize credit risk. Therefore, to overcome this problem, in this study, we developed a hybrid CNN—SVM/RF/DT model to predict credit risk. Four classification approaches were examined in order to build the model. The first classifier is a fully connected layer with soft-max that is trained using an end-to-end process, whereas the other three classifiers are binary SVM/RF/DT classifiers which are piled on top by removing the final fully connected and soft-max layer.

The pooling layer consequently reduces the amount of computation carried out within the network as well as the number of parameters that must be derived from the credit data. The feature maps that are produced by following functional a number of convolution and pooling layers to the credit data are flattened into a 1D array and used as inputs for the SVM, RF, and DT. Different values of the parameters were taken into consideration and fine-tuned throughout training and testing the model in order to choose appropriate parameters. In accordance with the experimental findings, a fully connected CNN and a hybrid CNN with SVM, DT, and RF, respectively, achieved a prediction performance of 86.70%, 98.60%, 96.90%, and 95.50%. According to the results, our proposed hybrid method exceeds the fully connected CNN in its ability to predict credit applicants.

Data Availability

Data supporting this research article are available on request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] S. A. M. Pasha and B. Mintesinot, "Assessment of credit risk management system in Ethiopian banking," *International Journal of Business and Management Invention*, vol. 6, no. 4, pp. 98–110, 2017.
- [2] Y. Hu and J. Su, "Research on credit risk evaluation of commercial banks based on artificial neural network model," *Procedia Computer Science*, vol. 199, pp. 1168–1176, 2022.
- [3] N. Bussmann, P. Giudici, D. Marinelli, and J. Papenbrock, "Explainable machine learning in credit risk management," *Computational Economics*, vol. 57, no. 1, pp. 203–216, 2021.
- [4] P. Sharifi, V. Jain, M. A. Poshtkahi, E. Seyyedi, and V. Aghapour, "Banks credit risk prediction with optimized ann based on improved owl search algorithm," *Mathematical Problems in Engineering*, vol. 2021, Article ID 8458501, 10 pages, 2021.
- [5] A. A. Robisco and J. M. C. Martínez, "Measuring the model risk-adjusted performance of machine learning algorithms in credit default prediction," *Financial Innovation*, vol. 8, no. 1, Article ID 70, 2022.
- [6] X. Wang, "Analysis of bank credit risk evaluation model based on BP neural network," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 2724842, 11 pages, 2022.
- [7] Z. Li, Z. Jiang, and X. Pan, "Default risk prediction of enterprises based on convolutional neural network in the age of big data: analysis from the viewpoint of different balance ratios," *Complexity*, vol. 2022, Article ID 5139562, 18 pages, 2022.
- [8] C.-H. Weng and C.-K. Huang, "A hybrid machine learning model for credit approval," *Applied Artificial Intelligence*, vol. 35, no. 15, pp. 1439–1465, 2021.
- [9] A. Ghosh, A. Sufian, F. Sultana, A. Chakrabarti, and D. De, "Fundamental concepts of convolutional neural network," in *Recent Trends and Advances in Artificial Intelligence and Internet of Things*, V. Balas, R. Kumar, and R. Srivastava, Eds., vol. 172 of *Intelligent Systems Reference Library*, pp. 519–567, Springer, Cham, 2020.
- [10] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.
- [11] F. Emmert-Streib, Z. Yang, H. Feng, S. Tripathi, and M. Dehmer, "An introductory review of deep learning for prediction models with big data," *Frontiers in Artificial Intelligence*, vol. 3, Article ID 4, 2020.
- [12] M. G. Ragab, S. J. Abdulkadir, N. Aziz et al., "A novel one-dimensional CNN with exponential adaptive gradients for air pollution index prediction," *Sustainability*, vol. 12, no. 23, Article ID 10090, 2020.
- [13] M. Gholami, M. Ranjbargol, R. Yousefzadeh, and Z. Ghorbani, "Integrating three smart predictive models using a power-law committee machine for the prediction of compressive strength in masonry made of clay bricks and cement mortar," *Structures*, vol. 55, pp. 951–964, 2023.
- [14] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, pp. 611–629, 2018.
- [15] R. Yousefzadeh, A. Kazemi, M. Ahmadi, and J. Gholinezhad, "Dimensionality reduction methods used in history matching," in *SpringerBriefs in Petroleum Geoscience and Engineering*, pp. 75–91, Springer Nature, United States, 2023.
- [16] H. T. Weldegebriel, H. Liu, A. U. Haq, E. Bugingo, and D. Zhang, "A new hybrid convolutional neural network and extreme gradient boosting classifier for recognizing handwritten Ethiopian characters," *IEEE Access*, vol. 8, pp. 17804–17818, 2020.
- [17] L. Alzubaidi, J. Zhang, A. J. Humaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, pp. 1–74, Article ID 53, 2021.

- [18] S. Kumar, V. K. Gunjan, M. D. Ansari, and R. Pathak, "Credit card fraud detection using support vector machine," in *Proceedings of the 2nd International Conference on Recent Trends in Machine Learning, IoT, Smart Cities and Applications*, V. K. Gunjan and J. M. Zurada, Eds., vol. 237 of *Lecture Notes in Networks and Systems*, Springer, Singapore, 2022.
- [19] S. Sahu, M. Prasad, and B. Tripathy, "A support vector machine binary classification and image segmentation of remote sensing data of Chilika Lagloon," *IJRIT International Journal of Research in Information Technology*, vol. 3, no. 5, pp. 191–204, 2015.
- [20] S. Purohit and A. Kulkarni, "Credit evaluation model of loan proposals for Indian banks," in *2011 World Congress on Information and Communication Technologies*, pp. 868–873, IEEE, Mumbai, India, December 2011.
- [21] M.-C. Lee and C. To, "Comparison of support vector machine and back propagation neural network in evaluating the enterprise financial distress," *International Journal of Artificial Intelligence & Applications*, vol. 1, no. 3, pp. 31–43, 2010.
- [22] I. Smeureanu, G. Ruxanda, and L. M. Badea, "Customer segmentation in private banking sector using machine learning techniques," *Journal of Business Economics and Management*, vol. 14, no. 5, pp. 923–939, 2013.
- [23] V. Piccialli and M. Sciandrone, "Nonlinear optimization and support vector machines," *Annals of Operations Research*, vol. 314, pp. 15–47, 2022.
- [24] D. P. Bertsekas, "Nonlinear programming," *Journal of the Operational Research Society*, vol. 48, no. 3, 334 pages, 2017.
- [25] L. Bottou, O. Chapelle, D. DeCoste, and J. Weston, "Support vector machine solvers," in *Large Scale Kernel Machines*, pp. 1–27, MIT Press, 2007.
- [26] S. Abe, *Support Vector Machines for Pattern Classification*, vol. 2 of *Advances in Computer Vision and Pattern Recognition*, Springer, London, 1st edition, 2005.
- [27] R. Gholami and N. Fakhari, "Support vector machine: principles, parameters, and applications," in *Handbook of Neural Computation*, pp. 515–535, Elsevier, 2017.
- [28] X. Ye, L.-A. Dong, and D. Ma, "Loan evaluation in p2p lending based on random forest optimized by genetic algorithm with profit score," *Electronic Commerce Research and Applications*, vol. 32, pp. 23–36, 2018.
- [29] G.-H. Kwak, C.-W. Park, K.-D. Lee, S.-I. Na, H.-Y. Ahn, and N.-W. Park, "Potential of hybrid CNN-RF model for early crop mapping with limited input data," *Remote Sensing*, vol. 13, no. 9, p. 1629, 2021.
- [30] A. Liaw and M. Wiener, "Classification and regression by randomforest," *R news*, vol. 2, no. 3, pp. 18–22, 2002.
- [31] J. Luan, C. Zhang, B. Xu, Y. Xue, and Y. Ren, "The predictive performances of random forest models with limited sample size and different species traits," *Fisheries Research*, vol. 227, Article ID 105534, 2020.
- [32] M. Gupta and C. Vanmathi, "A study and analysis of machine learning techniques in predicting wine quality," *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 10, no. 1, pp. 314–321, 2021.
- [33] M. Rakhra, P. Soniya, D. Tanwar et al., "WITHDRAWN: crop price prediction using random forest and decision tree regression—a review," *Materials Today: Proceedings*, 2021.
- [34] J. Morgan, *Classification and Regression Tree Analysis*, Boston University, Boston, 2014.
- [35] S. Ahlawat and A. Choudhary, "Hybrid CNN–SVM classifier for handwritten digit recognition," *Procedia Computer Science*, vol. 167, pp. 2554–2560, 2020.
- [36] B. Belay, T. Habtegebrial, G. Belay, and D. Stricker, "Using automatic features for text-image classification in amharic documents," in *Proceedings of the 9th International Conference on Pattern Recognition Applications and Methods ICPRAM*, pp. 440–445, SciTePress, 2020.
- [37] T. Tao and X. Wei, "A hybrid CNN–SVM classifier for weed recognition in winter rape field," *Plant Methods*, vol. 18, no. 1, Article ID 29, 2022.
- [38] X. Sun, J. Park, K. Kang, and J. Hur, "Novel hybrid CNN–SVM model for recognition of functional magnetic resonance images," in *2017 IEEE international conference on systems, man, and cybernetics (SMC)*, pp. 1001–1006, IEEE, Banff, AB, Canada, October 2017.
- [39] M. Madanan, B. T. Sayed, N. A. M. Zulkefli, and N. C. Velayudhan, "An artificial intelligence approach based on hybrid CNN–XGB model to achieve high prediction accuracy through feature extraction, classification and regression for enhancing drug discovery in biomedicine," *International Journal of Biology and Biomedical Engineering*, vol. 15, pp. 190–201, 2021.
- [40] P. Bhardwaj, P. Tiwari, K. Olejar Jr, W. Parr, and D. Kulasiri, "A machine learning application in wine quality prediction," *Machine Learning with Applications*, vol. 8, Article ID 100261, 2022.
- [41] A. Mustaqeem, S. M. Anwar, and M. Majid, "Multiclass classification of cardiac arrhythmia using improved feature selection and SVM invariants," *Computational and Mathematical Methods in Medicine*, vol. 2018, Article ID 7310496, 10 pages, 2018.
- [42] M. Hossin and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," *International Journal of Data Mining & Knowledge Management Process*, vol. 5, no. 2, pp. 1–11, 2015.
- [43] G. Varoquaux and O. Colliot, *Evaluating Machine Learning Models and their Diagnostic Value*, Springer, 2022.