Performance analysis of deep and machine learning algorithms for loan evaluation model

Tamiru Melese, Tesfahun Berhane*, Abdu M. Seid and Assaye Walelgn
Department of Mathematics, Bahir Dar University, Bahir Dar, Ethiopia

Received: November 15, 2022 **Accepted:** March 17, 2023 **Published:** June 30, 2023

ABSTRACT

In this study, we present a loan evaluation model that uses machine and deep learning algorithms using data obtained from a local bank in Ethiopia. We examined two important experiments: the first used a one-dimensional convolutional neural network deep learning method, while the second employed machine learning methods such as support vector machines, XGBoost, random forests, decision trees, and Naive Bayes classifiers. We train and implement the algorithms to decide whether to accept or reject a loan application. A comparison of the model performance under different performance metrics is provided. According to the experimental findings, machine learning algorithms outperform deep learning algorithms in terms of classification accuracy, precision, recall, and area under the curve (AUC). Therefore, from the experimental results, we draw the conclusion that Ethiopian banks should think about utilizing machine learning models for their loan evaluation process rather than relying on more subjective traditional methods.

Keywords: Convolutional neural network (CNN); Support Vector Machine; Decision tree; Random forest; XGB; Naive Bayes
DOI: https://dx.doi.org/10.4314/ejst.v16i2.1

INTRODUCTION

One of the main services that a commercial bank provides to its customers is loan. In providing loan, the bank is moving to a credit risk. That is, a risk with the possibility of a loss resulting from a borrower's failure to repay the loan. Therefore, evaluating a loan requires careful consideration.

Commercial banks in Ethiopia evaluate loan applications depending on the opinion of a loan officer. Such judgement is ineffective, erratic, subjective and uneven. In addition, in Ethiopian banks, the loan evaluation process takes a very lengthy time to complete. However, for any banking organization to be successful, the loan decision-making process must be completed quickly and accurately (Kumar *et al.*, 2021). Therefore, it is essential to design a model that learns from experience based on an applicant’s history in order to provide recommendations for lenders on how to grant loans. Several attempts have been made by various authors to formulate models that evaluate loan in different areas of the world. Controlling the credit default risk using digital technologies is crucial for the sustainable growth of the credit business. Traditional research on the

* Corresponding author: tesfahunb2002@gmail.com

credit default prediction model tends to focus more on the model's accuracy while conveniently ignoring the practical aspects of credit risk management. Manually extracting the data characteristics will lessen the high-dimensional connection among the analysed data, reducing the model's involvement, but will have the unintended consequence of lowering the model's predictive performance. To overcome this issue, Zhou (Zhou *et al*., 2020) created a personal credit default prediction model using CNN (convolutional neural network), and the model's performance was evaluated using accuracy and AUC (the area under the ROC curve). According to experimental findings, the model performs better than the SVM (support vector machine), Bayes, and RF.

Aleksandrova *et al*. (2021) employed a number of well-known machine learning algorithms for credit scoring in peer-to-peer lending. The data used to build the models was obtained from the official Lending Club website. Among the models used are single classifiers (such as logistic regression, decision trees, and multilayer perceptron’s), homogeneous ensembles (such as XGBoost, GBM, and Random Forest), and heterogeneous ensemble classifiers (such as Stacked Ensembles). During developing and evaluating each machine learning model, they adjust several parameters in order to assess the model's performance. They should evaluate many metrics when evaluating models, including accuracy, balanced accuracy, sensitivity, specificity, Kappa coefficient, AUC, and log loss. The findings of the experiments conducted for their study show that ensemble models outperform individual classifiers on the test set in terms of universality.

Li *et al*. (2006) examined the feasibility of using SVMs to manage consumer loans using a data set gathered from a local Taiwanese bank. They compare their performance to that of multilayer perceptron using cross-validation, and they use the paired difference test to support their general ability. They also take into account how Type I and Type II errors affect the process of determining whether to give loans or not. They come to the conclusion that SVM performs better in terms of generalization performance and visualization than neural network models, assisting decision-makers in choosing the optimal techniques for loan evaluation. Using Australian credit approval datasets and a neural network model based on the backpropagation neural learning technique, Khashman (2009) provides a system for assessing credit risk. He developed and applied seven learning techniques to train a neural network that selects whether to approve or reject credit applications. In addition to evaluating how well two neural networks with one and two hidden layers performed under various learning schemes, he also compared how well they performed under the best learning scheme. His research findings demonstrate that neural networks can be utilized to process credit applications effectively.

To improve the accuracy and stability of default discrimination, Li *et al*. (2021) considers creating a personal credit default discrimination model based on a Super

Learner heterogeneous ensemble. They initially select three categories of homogenous ensemble classifiers, including random forest, and six categories of single classifiers, including SVM and logistic regression, in order to build a base classifier candidate library for Super Learner. The ten-fold cross-validation approach is then used to assess the base classifier in an effort to improve its robustness. The overall loss of the base classifier is calculated as the difference between the predicted and actual values. The ideal weight of the base classifier is then solved for, minimizing the weighted total loss of all base classifiers, using a base classifier-weighted optimization model. This results in the creation of the heterogeneous ensembled Super Learner classifier. Finally, they use three real credit datasets from the UCI database for Australia, Japan, and Germany as well as the enormous credit dataset GMSC published by the Kaggle platform to evaluate the performance of the ensembled Super Learner model. AUC, type I error rate, type II error rate, and accuracy rate are the four commonly used evaluation metrics. The outcomes show that the Super Learner heterogeneous ensemble model constructed has better discrimination accuracy and robustness than other single classifiers and ensembled classifiers.

Óskarsdóttir and Bravo (2021) develops a multilayer customized PageRank algorithm that makes it possible to determine the network-wide strength of every borrower's default exposure. In an agricultural loan setting, where it has long been hypothesized that default correlates between borrowers when they are subjected to the same structural risks, they test their methods. According to their findings, adding multilayer network information about centrality can significantly improve prediction, and more complicated information, like multilayer PageRank variables, can further improve prediction. According to their findings, default risk increases when a person is connected to a lot of defaulters, but it is reduced by the size of the person's neighbourhood, demonstrating that both default risk and financial stability spread throughout the network. Neural networks are used as the main modelling technique by Wang *et al*. (2022), which concentrates on the evaluation of commercial banks' credit risk. It combines mutation genetic algorithms and BP neural networks. Using mutation genetic algorithms, neural network critical parameter combinations are adjusted, boosting their effectiveness. The evaluation model put forward in their work is more accurate than 65%, according to the validation of different assessment models, and the evaluation results enhanced by the mutation genetic algorithm are more acceptably correct than 85%. 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. It has been demonstrated that the improved method outperforms the conventional neural network approach in terms of performance. It is significant both theoretically and practically for the development of the commercial banks' credit risk prevention system. In emerging nations like Ethiopia, loan facilities are extremely vital. In order to ensure continuing growth and development in all areas of life, including social and economic elements, it is essential to evaluate loan applicants and find additional emphasis criteria for the loan decision. Most banks in Ethiopia have been hesitant to apply deep learning and machine learning approaches in their decision-making in spite of the rise in loan needs and competition in the banking sector. Instead, they use subjective evaluation. Additionally, Ethiopian civil servants hardly ever have the opportunity to obtain a mortgage, auto, education, personal, or medical loan. Therefore, it is necessary to provide lenders with recommendations on how to grant loans to civil servants and to address the issues we highlighted above.

METHODOLOGY

The data for building the model is taken from a local bank in Ethiopia which is derived from individual customer's profiles covering the time from 2011 to 2018 G.C. The target variable in the dataset is grouped into two, which are accepted applicants, and rejected applicants for the loan.

By interviewing the loan team of the bank we take amendments on the types of loan, loan duration, interest rate, and types of collateral input variables. In types of loan input variables, we include education, personal, and medical loan. Moreover, we extend the loan duration from 25 years to 30 years, increase the current interest rate by 2% in the interest rate input variables, and include salary as collateral in the collateral input variables in addition to physical assets and other collateral types. The variables are as follows:

Dependent variables: loan (accepted or rejected). The output is binary data representation "1" for accepted a loan and "0" for rejected a loan (see Figure 1).

Total income, loan collateral and amount of loan are significantly different for both accepted and rejected loan and can greatly impact the performance of binary classification models. Co-applicant income is right skewed (Figure 2).

The data set is partitioned into two, (80%) are used as the training set for building up the classification model, and the remaining holdout fold (20%) as the test set for justifying the generalization performance of the model. Based on the input feature qualities, either categorical or continuous, the task of filling in missing values in the data set is carried out using the median and mode. Additionally, box plots and chi-square tests are used to determine whether the input features are correlated with the target variable.

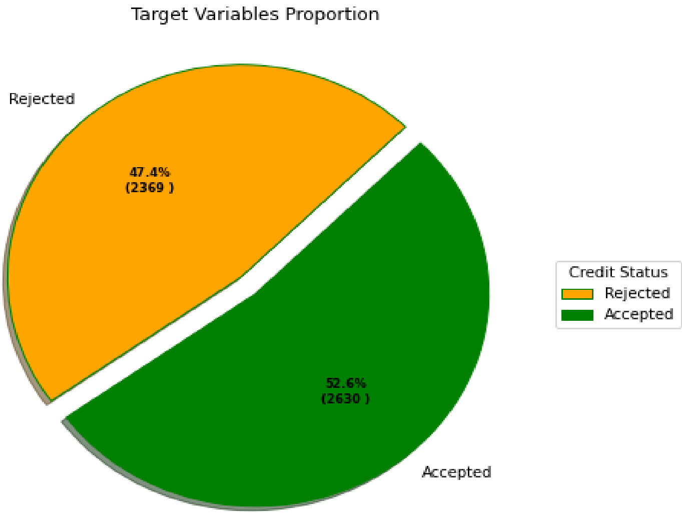

Figure 1. Proportion of target variable: loan status

A descriptive statistical analysis of the input features is crucial to machine learning since it makes it simpler to understand your data. This comes from machine learning's emphasis on prediction. While this is a crucial first step, statisticians concentrate on drawing conclusions from data. The descriptive analysis of the input features is displayed in Table 1.

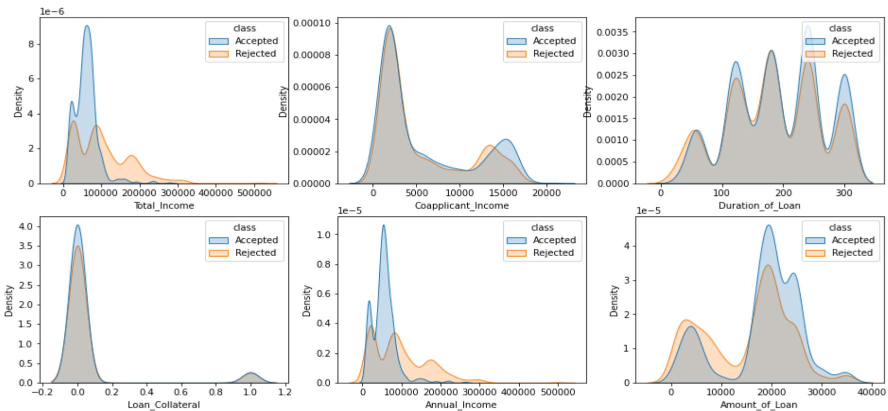

Figure 2. Relation between input features and target variables: The density plot is a particular type of data visualization tool. The density plots show the distribution of the key features among clients whose loan applications the bank approves and denies.

Table 1. Descriptive statistics of input features

Input features	Count	Mean	Std	Min	Quartile1	Quartile2	Quartile3	Max
Age	17	24.00	6.53	21.00	22.00	24.00	25.00	144.00
Gender	17	0.7	0.46	0.00	0.00	1.00	1.00	1.00
Marital status	17	0.35	0.48	0.00	0.00	0.00	1.00	1.00
Job	17	1.16	1.26	0.00	0.00	1.00	2.00	3.00
Education	17	0.90	1.08	0.00	0.00	0.00	2.00	3.00
Work experience	17	3.94	2.91	0.00	2.00	3.00	6.00	11.00
Type of credit	17	1.99	1.56	0.00	1.00	1.00	3.00	5.00
Annual income	17	4.75	0.34	3.98	4.50	4.78	4.95	5.70
Co-applicant income	17	3.56	0.40	2.70	3.26	3.40	4.00	4.33
Total income	17	4.80	0.31	4.09	4.57	4.84	4.98	5.71
Duration of credit	17	188.45	75.50	24.00	120.00	180.00	240.00	300.00
Collateral	17	0.06	0.25	0.00	0.00	0.00	0.00	1.00
Credit history	17	2.39	1.59	0.00	1.00	2.00	4.00	5.00
Credit grade	17	1.64	1.28	0.00	1.00	1.00	2.25	6.00
Amount of credit	17	4.12	0.37	2.70	3.95	4.30	4.35	4.54
Credit interest rate	17	12.02.	3.15	5.12	9.99	11.99	14.17	21.21

Convolutional neural network

Convolutional Neural Network (CNN), also known as ConvNet, is a type of Artificial Neural Network (ANN) with a deep feed-forward architecture and incredible generalizing ability when compared to other networks with fully connected layers. It can learn highly abstracted features of objects and identify them more efficiently than other networks with fully connected layers (Ghosh *et al.*, 2020). A Convolutional Neural Network is composed of multiple building blocks (known as layers of the architecture) (Ghosh *et al.*, 2020) such as the convolution layers, a convolution operation is defined, in which a filter is used to map the activations from one layer to the next (Aggarwal, 2018), a pooling layer which is inserted after a convolution layer that helps to reduce the size of feature maps and network parameters (Gholamalinezhad and Khosravi, 2020), activation function refers to the feature of activated neurons can be retained and mapped out by a non-linear function, which can be used to solve nonlinear problems (Wang *et al.*, 2020).

Support vector machine

Support Vector Machine is the state-of-the-art neural network technology based on statistical learning (Vapnik, 1999). SVM seeks a hyper-plane that simultaneously minimizes empirical error and maximizes the margin, including the kernel function alongside its parameters, as well as the slack penalty coefficient C (Dudzik *et al.*, 2021). It was originally designed for binary classification in order to construct an optimal hyper-plane so that the margin of separation between the negative and positive data set will be maximized (Li *et al.*, 2006).

We consider a machine learning approach to 2-class hyper-plane separation. Given a training set of instance pairs $\{ (x_i, y_i): x_i \in R^n, i = 1,2, \ldots, n$ and class labels $y_i = \pm 1\}$, we want to find a hyper-plane which has a maximum margin for $w \in R^n$ and offset scalar b such that

$$\begin{cases} w.x_i + b_i \ \geq 0\,, & \text{for } \ y_i = +1 \\ w.x_i + b_i \ < \ 0\,, & \text{for } \ y_i = -1 \end{cases} \qquad (1)$$

The constrained optimization problem for maximum margin is

$$\text{minimize } \ \emptyset(w) \ = \ \frac{1}{2} w.w \qquad (2)$$

$$\text{subject to } \ y_i(x_i, w + b) \geq 1, \quad \text{for } \ i = 1,2, \ldots, m$$

Using Lagrange multipliers equation (2) and simplifying, we get

$$minimize \ \ L(w, b, \lambda) = \frac{1}{2} w.w - \sum_{i=1}^{m} \lambda_i (y_i (w, x_i + b) - 1) \qquad (3)$$

where λ_i is Lagrangian multiplier.
To find the minimum of (3) over w, b (while fixing all λ_i), we set the gradient vector to zero

$$w = \sum_{i=1}^{m} \lambda_i \, x_i \, y_i \qquad and \qquad \sum_{i=1}^{m} \lambda_i \, y_i \ = 0 \qquad (4)$$

The cost function which gives the maximum hyper-plane used to classify the two classes is

$$L(w, b, \lambda) = \sum_{i=1}^{m} \lambda_i - \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i \lambda_j y_i y_j x_i x_j \qquad (5)$$

$$\text{subject to } \ \sum_{i=1}^{m} \lambda_i y_i, \quad \forall \lambda_i \geq 0$$

XGBoost

To counter the challenges of support vector machine for the development of the loan evaluation model and to increase the accuracy of the model we propose a XGBoost model. XGB is a gradient boosting variant that uses a cutting-edge tree search methodology (Gupta *et al.*, 2022). So, XGB which is an ensemble additive model that is composed of several base learners' algorithm used to train a standalone random forest and random forest for testing the model.

Consider a set of data points (x_i, y_i), where $\{ x_i = x_1, x_2, \ldots, x_n \}$, for $\{ i = 1,2, \ldots, n \}$ and $y_i \in \{-1, +1\}$, where $y_i = -1$ is the loan which is rejected and $y_i = +1$ is the loan which is accepted. The objective function of the XGBoost model is given by the sum of the loss function and regularization

$$L(\emptyset) = \sum_{i} l(y_i, \hat{y}) + \sum_{k} \Omega(f_k) \quad (6)$$

where $\Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2$. l is a loss function which controls the predictive power of the model by considering the difference between the actual value y_i and the predicted value $\widehat{y_i}$, and Ω is a regularization which helps to control the complexity of XGBoost models (i.e., the regression tree functions) and prevents over-fitting by smoothing the final learnt weight.

For an iterative algorithm, equation (1) can be written as

$$L^{(t)} = \sum_{i=1}^{N} L\left(y_i, \widehat{y_i}^{(t)}\right) + \sum_{i=1}^{t} \Omega(f_t) \quad (7)$$

$$= \sum_{i=1}^{N} L(y_i, \widehat{y_i}^{(t-1)} + f_t(x_i) + \sum_{i=1}^{t} \Omega(f_t) \quad (8)$$

where $\hat{y}_i^t$ is the prediction of the i^{th} iteration and it is mandatory to add f_t to optimize the objective function. But it is difficult to optimize the objective function using traditional optimization methods in Euclidean space. After we approximate equation (8) based on Taylor series expansion up to second order, we get

$$L^{(t)} = \sum_{i=1}^{n} \left[l(y_i, \hat{y}^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \quad (9)$$

where $g_i = \partial\, \hat{y}^{(t-1)}\; l(y_i, \hat{y}^{(t-1)})$ and $g_i = \partial^2\, \hat{y}^{(t-1)}\; l(y_i, \hat{y}^{(t-1)})$ are the first and the second partial derivatives of the loss function.

The objective function can be simplified as

$$\hat{L}^{(t)} = \sum_{j=1}^{T} \left[G_j w_j + \frac{1}{2}\left(H_j + \lambda\right) w_j^2 \right] + \gamma T \quad (10)$$

where $G_j = \sum_{i \in I_j} g_i$ and $H_j = \sum_{i \in I_j} h_i$.

For each quadratic function: equation (10) the optimal weight is

$$w_j^* = -\frac{G_j}{H_j + \lambda} \quad (11)$$

And the corresponding optimal value is

$$min\hat{L}^{(t)} = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T \quad (12)$$

The gain of the split is

$$L_{split} = \frac{1}{2}\left[\frac{(\sum_{i \in IL} g_i)^2}{\sum_{i \in IL} h_i + \lambda}\right] + \frac{1}{2}\left[\frac{(\sum_{i \in IR} g_i)^2}{\sum_{i \in IR} h_i + \lambda}\right] + \frac{1}{2}\left[\frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda}\right] \quad (13)$$

Random forest

A random forest is a classifier consisting of a collection of tree structured classifiers $h(x, \theta_k), k = 1,2, \dots,$ where the θ_k are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x (Breiman, 2001).

The random forest classifier is ensemble decision based supervised machine learning that can be used for classification and regression problems. The random forest approach can reduce bias because each tree is trained on a subset of data, is highly stable, performs well when data contains missing values and resolves the over-fitting problem in decision trees. So, due to this characteristics, we can apply random forest to evaluate loan based on the input features of the customers past experience (i.e., historical data of the customers). The development of the tree to evaluate the model is dependent on a random vector from the input data and each tree construction is independent each other, but the distribution of the trees is the same in all the forest. In this instance, the random forest classifier, which is a meta-learner of various trees constructed independently of one another, is used as the classification algorithm.

$$G = \sum_{i=1}^{C} p(i)(1 - p(i)) \quad (14)$$

Where C is the total number of class in dataset T and $p(i)$ is the probability of picking the data point with the class i.

Decision trees

A decision tree is a classification algorithm that is expressed as a recursive division, which refers to a divide-and-conquer strategy of the input space depending on the values of the attributes (Mashat *et al.*, 2012). There is a root node, a few branches, and leaf nodes in every decision tree. Due to the decision tree's ability to swiftly classify unknown records, quite effective—providing the parameters are calibrated optimally, useful for decision-related problems, and simple to create and interpret. In this study, we evaluate a loan using a decision tree algorithm. The Decision tree algorithm evaluates a loan applicant's customers using a tree-like data structure. A collection of pre-processed loan data are fed into the algorithm because the decision tree uses a supervised methodology, and we train the algorithm using this data. When the decision tree's maximum depth is achieved, the partitioning procedure is terminated (Al Aghbari, 2015).

Naive Bayes Classifier

A Naive Bayes classifier uses the Bayes theorem with strong (naive) independence assumptions to create a straightforward probabilistic classifier (Hu *et al.*, 2010).

RESULTS AND DISCUSSION

The accuracy of 1D-CNN model with learning scheme 80:20% is 82.20%. The quality of a positive prediction made by the model (i.e., the precision of the model) is 93.75%. The percentage of correctly predicted positive outcomes out of all the actual positive outcomes (recall) is 75.98%, which is the ratio of true positive (TP) to the sum of true positives and false negatives (TP + FN). From figure 3 we can observe that the measure of the percentage of false positives against all positive predictions which is the false positive rate is 0.0799.

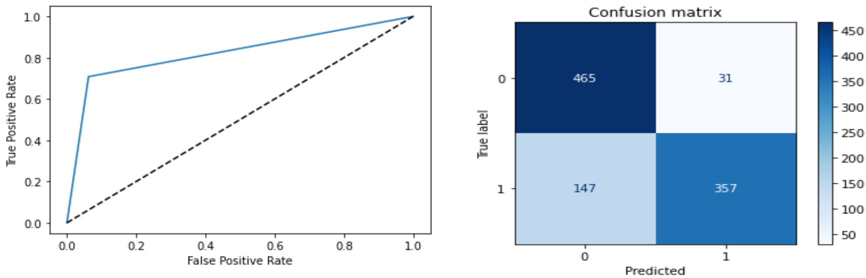

Figure 3. 1D-CNN ROC & confusion matrix

The accuracy of the SVM model with radial bases kernel function and learning scheme 80:20% is 87.83%. The quality of a positive prediction made by the model (i.e., the precision of the model) is 86.59%. The percentage of correctly predicted positive outcomes out of all the actual positive outcomes (recall) is 84.63%, which is the ratio of true positive (TP) to the sum of true positives and false negatives (TP + FN). The measures of the percentage of false positives against all positive predictions which is the false positive rate is 0.0432. The ROC curve which consists of the TPR and FPR at various classification threshold levels. The model classified more items being positive when we lower the threshold levels. The area under the curve (i.e., the ROC AUC) is 87.89% (see Figure 4).

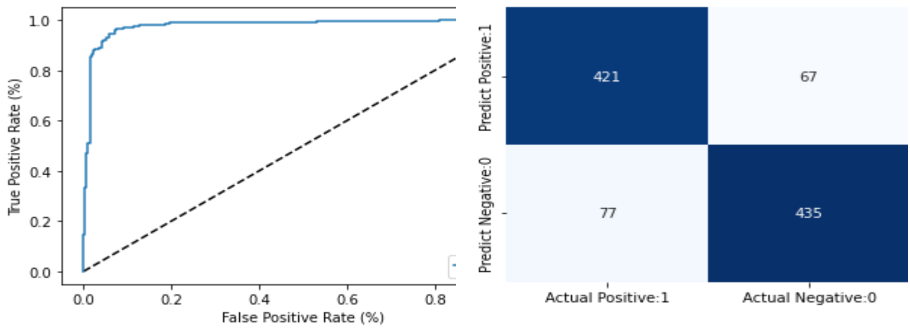

Figure 4. SVM ROC & confusion matrix

The classification accuracy of random forest with learning scheme 80:20% is 89.83%. The quality of a positive prediction made by the model (i.e., the precision of the model) is 87.89%. From figure 5 one can observe that the percentage of correctly predicted positive outcomes out of all the actual positive outcomes (recall) is 89.53%, which is the ratio of true positive (TP) to the sum of true positives and false negatives (TP + FN). The measures of the percentage of false positives against all positive predictions which is the false positive rate is 0.0992.The area under the curve (i.e., the ROC AUC) is 89.67%.

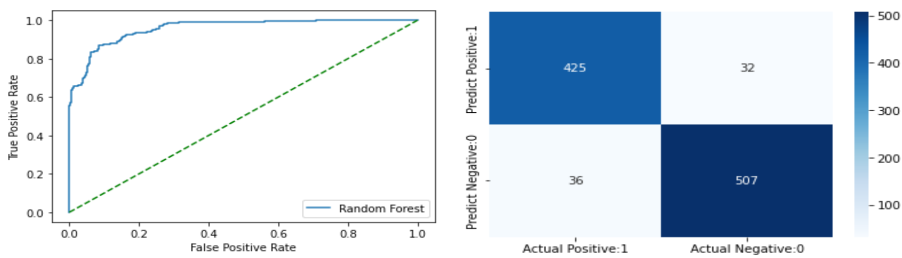

Figure 5. RF ROC & confusion matrix

The accuracy of the decision tree model with learning scheme 80:20% is 91.33%. The precision of the model is 96.85%. The percentage of correctly predicted positive outcomes out of all the actual positive outcomes (recall) is 86.56% (see Figure 6). The measures of the percentage of false positives against all positive predictions which is the false positive rate is 0.0867. The area under the curve (i.e., the ROC AUC) is 91.58%. The accuracy of the XGBoost model with learning scheme 80:20% is 95.08%. The quality of a positive prediction made by the model (i.e., the precision of the model) is 97.06%. The percentage of correctly predicted positive outcomes out of all the actual positive outcomes (recall) is 93.05%, which is the ratio of true positive (TP) to the sum of true positives and false negatives (TP + FN). From Figure 7, we understand that the measures of the percentage of false positives against all positive predictions which is the false positive rate is 0.0285. The area under the curve (i.e., the ROC AUC) is 95.15%.

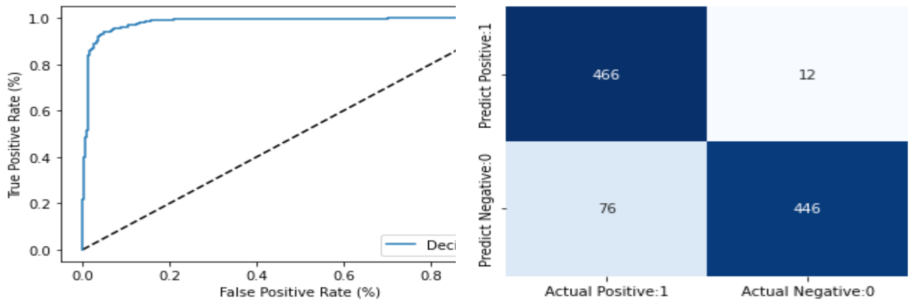

Figure 6. DT ROC & confusion matrix

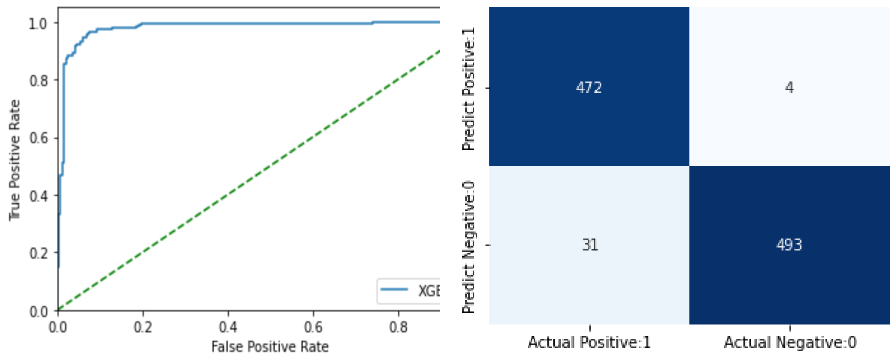

Figure 7. XGBoost ROC & confusion matrix

Table 2. Classification (%)

Classifiers	Accuracy	Precision	Recall	ROC-AUC	Classification Error	FPR
CNN	82.20	93.75	75.98	84.11	0.1780	0.0799
SVM	85.60	86.27	84.54	85.62	0.1440	0.1335
DT	91.20	97.49	85.98	91.47	0.0880	0.0262
RF	93.20	93.00	92.19	93.18	0.0680	0.0594
XGB	96.50	99.16	93.52	96.62	0.0350	0.0080
NB	93.20	98.74	88.37	93.44	0.0680	0.0350

Figure 8 shows that the performance of deep and machine learning models for loan evaluation.

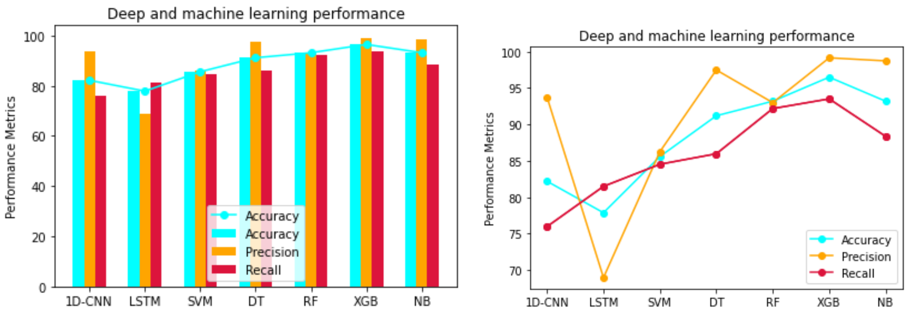

Figure 8. Performances of deep and machine learning models

CONCLUSION

In this study, we have presented a performance analysis of deep learning and machine learning for loan evaluation. The suggested approach was trained and tested using data collected from loan applicants from Ethiopian banks, classifying the borrowers into accepted and rejected. We examined two important experiments: the first used a one-dimensional convolutional neural network deep learning method, while the second employed machine learning methods such as support vector machines, XGBoost, random forests, decision trees, and Naive Bayes classifiers. A number of performance metrics, such as classification accuracy, precision, recall, and area under the curve are used to compare machine learning and deep learning algorithms. According to the experimental findings, machine learning algorithms outperform deep learning algorithms in terms of classification accuracy, precision, recall, and area under the curve. As a result, given the limited amount of data available, machine learning algorithms are preferred over deep learning methods for classifying loan applicants as accepted and rejected. Therefore, from the experimental results, we draw the conclusion that Ethiopian banks should think about utilizing machine learning models for their loan evaluation process rather than relying on more subjective traditional methods. Also, they are able to give loans to civil servants who are not deserving of this kind of service by modifying the loan period and interest rate.

REFERENCES

Aggarwal, C.C. (2018). Neural networks and deep learning. Springer Cham.

Al Aghbari, Z. (2015). Mining big data. *Proceedings of the 17th International Conference on Enterprise Information Systems* **1**: 379–384.

Aleksandrova, Y. *et al*. (2021). Comparing performance of machine learning algorithms for default risk prediction in peer-to-peer lending. *TEM Journal* **10**(1): 133–143.

Breiman, L. (2001). Random forests. *Machine learning* **45**(1): 5–32.

Dudzik, W., Nalepa, J and Kawulok, M. (2021). Evolving data-adaptive support vector machines for binary classification. *Knowledge-Based Systems* **227**(3): 107221. DOI:10.1016/j.knosys.2021.107221

Gholamalinezhad, H and Khosravi, H. (2020) Pooling methods in deep neural networks: A review. arXiv preprint arXiv:2009.07485.

Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A and De, D. (2020). Fundamental concepts of convolutional neural network. In book: Recent Trends and Advances in Artificial Intelligence and Internet of Things (pp.519-567). January 2020. DOI:10.1007/978-3-030-32644-9_36

Gupta, R., Yadav, A. K., Jha, S and Pathak, P.K. (2022) Enhancing classification performance of binary class imbalanced data for weather forecasting using machine learning classifiers. https://doi.org/10.21203/rs.3.rs-1546284/v1

Khashman, A. (2009) A neural network model for credit risk evaluation. *International Journal of Neural Systems* **19**(4): 285–294.

Kumar, A., Sharma, S and Mahdavi, M. (2021). Machine learning (ml) technologies for digital credit scoring in rural finance: A literature review. *Risks* **9**(11): 192.

Li, G., Shen, M., Li, M and Cheng, J. (2021). Personal credit default discrimination model based on super learner ensemble, mathematical problems in engineering. *Mathematical Problems in Engineering*. Volume 2021, Article ID 5586120. https://doi.org/10.1155/2021/5586120

Li, S.-T., Shiue, W and Huang, M.-H. (2006). The evaluation of consumer loans using support vector machines. *Expert Systems with Applications* **30**(4): 772–782.

Mashat, A.F., Fouad, M.M., Yu, P.S and Gharib, T.F. (2012). A decision tree classification model for university admission system. *Editorial Preface* **3**(10). DOI: 10.14569/IJACSA.2012.031003

Óskarsdóttir, M and Bravo, C. (2021). Multilayer network analysis for improved credit risk prediction. *Omega* **105**: 102520

Vapnik, V. (1999). The nature of statistical learning theory. *Springer New York, NY*

Wang, Y., Li, Y., Song, Y and Rong, X. (2020). The influence of the activation function in a convolution neural network model of facial expression recognition. *Applied Sciences* **10**(5): 1897. https://doi.org/10.3390/app10051897

Zhou, X., Zhang, W and Jiang, Y. (2020). Personal credit default prediction model based on convolution neural network. *Mathematical Problems in Engineering*. Volume 2020 | Article ID 5608392. https://doi.org/10.1155/2020/5608392.