Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7451152, 12 pages
https://doi.org/10.1155/2022/7451152

Research Article

Empirical Analysis of Machine Learning Algorithms for Multiclass Prediction

Umar Ishfaq,[1] Danial Shabbir,[1] Jumshaid Khan,[1] Hikmat Ullah Khan,[1] Salman Naseer,[2] Azeem Irshad,[3] Muhammad Shafiq,[4] and Habib Hamam[5,6,7,8]

[1]*Department of Computer Science, COMSATS University Islamabad, Wah Campus, Wah Cantt 470040, Pakistan*
[2]*Department of Information Technology, University of the Punjab Gujranwala Campus, Gujranwala 52250, Pakistan*
[3]*Department of Computer Science and Software Engineering, International Islamic University Islamabad, Pakistan*
[4]*Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, Republic of Korea*
[5]*Faculty of Engineering, Uni de Moncton, E1A3E9, Moncton, NB, Canada*
[6]*International Institute of Technology and Management, Commune d'Akanda, BP, Libreville 1989, Gabon*
[7]*School of Electrical Engineering, Department of Electrical and Electronic Engineering Science, University of Johannesburg, Johannesburg 2006, South Africa*
[8]*Spectrum of Knowledge Production & Skills Development, Sfax 3027, Tunisia*

Correspondence should be addressed to Muhammad Shafiq; shafiq@ynu.ac.kr

Received 4 January 2022; Revised 21 February 2022; Accepted 11 March 2022; Published 30 March 2022

Academic Editor: Alireza Souri

With the emergence of big data and the interest in deriving valuable insights from ever-growing and ever-changing streams of data, machine learning has appeared as an effective data analytic technique as compared to traditional methodologies. Big data has become a source of incredible business value for almost every industry. In this context, machine learning plays an indispensable role of providing smart data analysis capabilities for uncovering hidden patterns. These patterns are later translated into automating certain aspects of the decision-making processes using machine learning classifiers. This paper presents a state-of-the-art comparative analysis of machine learning and deep learning-based classifiers for multiclass prediction. The experimental setup consisted of 11 datasets derived from different domains, publicly available at the repositories of UCI and Kaggle. The classifiers include Naïve Bayes (NB), decision trees (DTs), random forest (RF), gradient boosted decision trees (GBDTs), and deep learning-based convolutional neural networks (CNN). The results prove that the ensemble-based GBDTs outperform other algorithms in terms of accuracy, precision, and recall. RF and CNN show nearly similar performance on most datasets and outperform the traditional NB and DTs. On the other hand, NB shows the lowest performance as compared to other algorithms. It is worth mentioning that DTs show the lowest precision score on the Titanic dataset. One of the main reasons is that DTs suffer from overfitting and use a greedy approach for attribute relationship analysis.

1. Introduction

The rapid development in web technologies resulted in the creation of immense volume of data, which requires efficient data extraction and intelligent data analysis for identifying relevant information. Machine learning (ML) is a relatively new domain of data analysis which plays an important role in emulating human intelligence in electronic devices. Resultantly, these devices can learn and progressively improve their performance on specific tasks without explicit programming [1]. A recent report suggests that ML will be the center of innovation in near future [2]. ML techniques have been successfully employed in web search [3], recommendation systems [4], email filtering [5], ad placement [6], fraud detection [5], credit scoring [7], stock trading [8, 9], and many other applications.

ML techniques are mainly divided into four categories: (1) supervised, (2) unsupervised, (3) semisupervised, and (4) reinforcement learning. In supervised learning, the classifiers are trained through examples. The classifier identifies patterns from the labeled data and learns from the

observations till achieving a certain level of performance. On the contrary, unsupervised learning interprets the structure of data and uses this information for organizing the data into groups or clusters. Unsupervised ML does not use data labels or any information about the desired outcome in advance. Similarly, semisupervised learning uses a combination of both labeled and unlabeled data for mining meaningful patterns. Research identifies that accuracy can improve significantly when unlabeled data is used in conjunction with labeled data [10]. Finally, reinforcement learning is a new area in ML that is concerned with achieving an optimal outcome through trial and error [11].

The objective of the paper is to present a comprehensive performance analysis of various classification algorithms for multiclass prediction using multiple datasets. The algorithms include Naïve Bayes (NB), decision trees (DTs), random forest (RF), gradient boosted decision trees (GBDTs), and DL-based convolutional neural networks (CNN). NB and DTs are classic ML algorithms. NB is one of the simplest and oldest classifiers which is based on Bayesian theorem. NB is particularly suited in situations where input dimensions are relatively high. Similarly, DTs present the decision results in a tree-like graph with all possible consequences, including chance event outcomes. DTs are mostly applied in decision analysis and operation research for identifying an effective strategy. On the other hand, RF [12] and GBDTs [12] are ensemble-based techniques. An ensemble technique uses multiple base algorithms for better classification results that could be obtained through any of the constituent base algorithm alone. GBDT is a stochastic prediction method that represents an ensemble or collection of single regression trees which are combined (i.e., mean) to give a final prediction. Similarly, RF takes multiple samples of training data, creates models for each data sample, and takes an average of these sample models for making a better estimate of true outcome. Finally, CNN [13] is a recently developed neural network-based classification approach. CNNs are like traditional neural network with neurons having learnable weights and biases. A neuron can receive many inputs, performs a weighted sum, and passes it to an activation function for the final output.

This study is inspired by some recent machine learning and deep learning-based studies in information technology, biology, and medicine. For instance, the study of Amiri et al. [14] employs six popular machine learning classifiers for examining radiomic features which are based on the computed tomography for predicting the risk of chronic kidney illness, particularly in patients undergoing radiation therapy for diseases such as abdominal cancer. Similarly, the study of Loreto et al. [15] addresses the challenge of discharge of patients from Intensive Care Units as ICU readmissions are linked to unfavorable outcomes such as lengthy expenses and high morality rates. The study shows that improving risk stratification for identifying patients highly susceptible of clinical deterioration might enhance the situation for chronically ill patients who are under hospital care.

This study is aimed at exploring the behavior of well-established ML and DL-based algorithms and presents a performance analysis of simple as well as ensemble-based ML algorithms against convolutional neural networks on multiple small and large datasets. The experimental setup consisted of thirteen datasets derived from different domains, publicly available at the repositories of UCI and Kaggle. The classifiers are evaluated using standard ML measures, i.e., accuracy, precision, and recall. In addition, we separately analyze the training and prediction time of DL and other established classifiers.

The key contributions of this study are summarized as follows:

(i) This study explores the behavior of well-established ML and DL-based algorithms for multiclass predictions

(ii) This study presents a performance analysis of simple as well as ensemble-based ML algorithms against convolutional neural networks on multiple small and large datasets

(iii) The study evaluates the performance of the classifiers using statistical measures such as accuracy and precision and concludes that gradient boosting decision trees (GBDTs) outperform other classifiers. This study also enlightens the researchers in choosing a baseline algorithm or proposing an ensemble-based technique using any of the examined classifiers

The rest of the paper is organized as follows: Section 2 presents the most relevant work to this study. Section 3 presents a brief introduction of the classifiers to be compared and discuss their underlying techniques. Section 4 presents the details of experimental setup including datasets and performance evaluation measures. Section 5 presents the results and discussion based on the experimental setup, and finally, Section 6 concludes the study based on the research findings.

2. Related Works

Several studies have been proposed in literature for empirically comparing the performance analysis of different classification algorithms. However, these studies do not fully analyze the classifier performance on datasets with varying attributes, types of attributes, and sizes.

In StatLog project, Caruana and Niculescu-Mizil [16] compare the performance of nine classifiers on large-scale datasets. The classifiers are selected from different branches of supervised classification including symbolic learning (using C4.5), statistics (using linear regression (LR), NB, and k-nearest neighbor (kNN)), and neural networks (NN). The findings of the study indicate that the performance of classifiers is solely dependent on the characteristics of datasets under investigation. Class imbalance is one of the leading reasons of performance degradation for classifiers, even for well-established classifiers such as latent Dirichlet allocation (LDA), support vector machine (SVM), and classification trees. Traditional

classifiers show serious deficiencies in predicting the instances of minority class [17].

Similarly, Brown and Mues [18] empirically evaluate the predictive power of eight supervised classifiers by gradually increasing class imbalance through random undersampling. The results of the study prove that RF and GBDTs perform significantly better on credit scoring datasets with pronounced class imbalances as compared to other classifiers. Research identifies that in credit scoring, data quality issues pose a challenge in scorecard development and risk measurement. However, with specific domain knowledge, the accuracy of credit scoring models can be significantly enhanced [19]. In addition, the predictive nature of data such that the captured characteristics are directly associated to the customer defaulting or not poses a serious challenge.

Surprisingly, over the past decade, DL has shown remarkable success in various research domains of artificial intelligence. DL-based techniques show superior performance [20] as compared to other ML methods in areas such as natural language processing, image, and voice recognition, among others. Luo et al. [21] present a performance analysis of Deep Belief Networks (DBN) against popular credit scoring methods such as LOGREG, multilayer perceptron (MLP), and SVM on a credit scoring dataset and identified that DBN outperforms other classifiers. However, the experiments conducted are restricted to Boltzmann machines only. Similarly, Sewak et al. [22] compared the performance of deep neural networks (DNN) against RF for malware classification using different sets of features. Though RF shows better performance as compared to DNN, however, the performance difference is negligible which requires further testing on complex datasets.

On the other hand, Abellán and Castellano [23] identify that a simple classifier with imprecise probabilities, when used as a base classifier in an ensemble scheme, can enhance the performance of other more complex classifiers for predicting credit risks. However, the study did not specify a standard criterion for selecting a base classifier. In this context, Lessmann et al. [24] proposed an ensemble-based approach which creates various noise-free balanced segments from large-scale raw datasets and builds multiple classifiers on these segments using a specific classification technique. The model combines results from multiple classifiers using specific ensemble rules. The results of the study from forty-six imbalanced datasets identify that the ensemble rule MaxDistance performs better with data balancing methods of SplitBal and ClusterBal as compared to other ensemble rules. In addition, the proposed ensemble-based approach outperforms the conventional external data balancing methods.

In addition to proposing a novel ensemble approach, a review of relevant literature identifies several studies which present a performance analysis of existing ensemble-based techniques. The work of Lorena et al. [25] identifies that RF shows better performance as compared to other classifiers through modelling the potential distribution of plant species using nine supervised ML classifiers. Each classifier extracted a different representation of relations between the distribution profile of plant species and environmental conditions. However, performance analysis was based only on image data with no multiclass attributes. Li et al. [26] investigate the performance of DTs, RF, and SVM. The authors have modeled the potential distribution of various local forest communities in New York State's Huntington Wildlife Forest (HWF). The results of the study indicate that RF and SVM produce better multitemporal predictions as compared to DTs. In addition, RF and SVM reflect changes in forest type much more effectively. On the other hand, Macià and Bernadó-Mansilla [27] presented the design of a mindful repository with properly characterized ML datasets. Consequently, the design of the repository lays the foundation of a well-supported methodology which can effectively assess a learner and provides a rich set of artificial benchmarks.

Similarly, automatic extraction of keywords is significant for text mining, information retrieval, and natural language processing. The study of Onan et al. [28] empirically analyzes effectiveness of the statistical keyword extraction approaches in conjunction with the ensemble learning methods. On the other hand, the study of Onan [29] proposed a recurrent neural network-based approach for opinion mining on instructor review database using an ensemble of classical text representation and word-embedding schemes. The results show the superiority of deep learning-based techniques over conventional machine learning-based algorithms. In sentiment analysis, sarcasm is a form of nonliteral language where users usually express negative emotions by using words having positive literal meanings. The study of Onan [30] presents a deep learning-based model for detecting sarcasm by comparing the predictive performance of topic-based word-embedding schemes against conventional word-embedding approaches. In addition, the study incorporates several implicit and explicit word-embedding-based features. Similarly, in their study [31], Onan and Toçoğlu presented an inverse gravity-based term weighted framework of word embedding with trigrams. The study assigns higher priority to critical words by considering word-ordering information. In addition, the authors introduce a three-layered architecture based on an efficient stacked bidirectional memory scheme. Finally, the study of Onan [32] presented supervised hybrid clustering that is based on k-means and cuckoo search algorithm for partitioning data samples from each class into different clusters resulting in higher diversity of training subsets.

Diagnostic classification of fatal diseases such as cancer can greatly improve the surveillance and treatment procedures for patients. The study of Ma et al. [33] proposed an extreme gradient boosting-based classification model by employing dense multiomics data for segregating early and late stages of cancer. On the other hand, predicting PPI (protein-protein interaction) sites can be significant for getting an insight into the biological activity. The study of Wang et al. [34] proposes a machine learning algorithm that employs eXtreme gradient boosting enhancing the prediction of PPI sites and alleviating heavy expenses associated with running costly and time taking biological experiments.

3. Performance Analyses of ML Algorithms

This study is aimed at presenting the performance analysis of well-established ML techniques against newly developed DL-based algorithms for multiclass prediction; thereby assessing to what extent these classifiers are affected by increasing the complexities of the datasets in terms of size, attributes, and types of attributes. A brief explanation of each of the techniques applied in this study is given in the following.

3.1. Machine Learning-Based Classifiers. ML-based algorithms range from classic Naive Bayes (NB) to ensemble-based decision trees (DTs), random forest (RF), and gradient boosted trees (GBDTs).

3.1.1. Naïve Bayes (NB). Naïve Bayes (NB) is a supervised ML technique which uses probabilistic Bayesian rule for classification. The probabilistic rule allows representing the uncertainty about the model by determining the probabilities of outcome. Given the class label, NB assumes that the features of a dataset are conditionally independent. In [35], the Bayes theorem is given as

$$P() = P()P(h)P(D), \tag{1}$$

where $P(h)$ and $P(D)$ are prior probabilities of hypothesis h and training data D, respectively. On the other hand, $P(h/D)$ represents probability of hypothesis h given the training data D. Similarly, $P(D/h)$ represents the probability of training data D given the hypothesis h. NB is among the most practical models because of speed and space efficiency. It is widely used in text classification, diagnostic, and predictive problems. However, in datasets where the features are not conditionally independent, such as gene expression data due to coregulation, NB suffers performance deficiencies.

3.1.2. Decision Trees (DTs). A DT generates a tree structured decision rule from a large input sample and extracts knowledge to classify the sample input into one of its possible classes. The existing literature presents various DT-based algorithms. However, this paper uses C4.5 as the underlying DT algorithm for classifying the input datasets. C4.5 [36] is an extended version of Iterative Dichotomiser 3 (ID3). ID3 starts from the given set of attributes (S) as the root node. And, for each of the iterations, it computes the information gain and entropy of every unused attribute of the set (S). The attributes with maximum information gain (or minimum entropy) are selected, and the set (S) is partitioned based on the selected attributes resulting in subsets of data. The algorithm continues by considering only the attributes never selected before on the subsets of data and stops when there are no more attributes left for selection or each element in the subset belongs to same class thereby turning into a leaf node. ID3 is based on greedy search. Using the concept of information gain, ID3 selects a test and avoids other possible choices. Information gain is computed as in [36]

$$\text{Gain}(S, A) = \text{entropy}(S) - \sum_{v} \frac{|S_v|}{|S|} \text{entropy}(S_v), \tag{2}$$

where S represents the training set, A indicates a specific attribute, v denotes all possible values of attribute A, and $|S|$ and $|S_v|$ are number of elements in S and S_v, respectively. Similarly, C4.5 works in the same manner as ID3. However, the splitting is based on normalized information gain (NIG) which effectively solves overfitting problem. DTs offer many advantages such as the ability to classify numerical, nominal, and textual input types. DTs can handle datasets with missing values and are available in different data mining packages or platforms.

3.1.3. Gradient Boosted Trees (GBDTs). Gradient boosting [37, 38] is an ensemble approach for classification and regression problems, which employs forward-learning mechanism. GBDT produces a prediction result through an ensemble of weak prediction models, mostly decision trees. Weak learners are iteratively integrated into a single and strong prediction model. The algorithm begins by training a decision tree where each observation is given an equal weight. After evaluating the first tree, the weights are increased for those observations which are difficult to classify and decreased for the observations which can be easily classified. The second tree is grown upon the improved predictions of the first tree and computes the classification error from this 2-tree ensemble model. Similarly, the algorithm continues to grow a third decision tree for predicting the revised residuals. This process continues to repeat for a specified number of iterations. Therefore, the final prediction of GBDT is based on the weighted sum of predictions made by the previous trees resulting in improved classification of observations which are not well classified earlier. Gradient boosting can be easily explained in terms of least-squares regression setting where the aim is to "teach" a model F for predicting values of the form $\widehat{y} = F(x)$ by minimizing mean square error given as

$$\text{Mean square error} = \frac{1}{n}(\widehat{y}_i - y_i)^2, \tag{3}$$

where i is an index over some training dataset of size n and y is the response or output variable. At each iteration m such that $1 \le m \le M$, it is assumed that there exists some weak learner F_m and each subsequent learner F_{m+1} is an improvement to its predecessor F_m by adding an estimator h given as in [38]

$$F_{m+1}(x) = F_m(x) + h(x). \tag{4}$$

In [38], we can also find the perfect value of h:

$$F_{m+1}(x) = F_m(x) + h(x) = y \tag{5}$$

or equivalently,

$$h(x) = y - F_m(x). \tag{6}$$

Finally, GBDTs fit h to the remaining $y - F_m(x)$. These remaining or residuals for a given model $F(x)$ represent the negative gradients of squared error loss function given as under

$$\text{Squared error loss function} = \frac{1}{2}(y - F(x))^2. \tag{7}$$

Therefore, GBDTs are, in fact, gradient descent functions. Gradient boosting is simple and effective, particularly, for learning nonlinear functions. One of the biggest advantages of GBDTs is that it decreases human interpretability. However, GBDTs take longer time to produce classifier outcome.

3.1.4. Random Forest (RF). Random forest (RF) consists of multiple DTs which operate as an ensemble [12]. Each individual tree in RF produces a class prediction, and the class with maximum votes is selected as the model's prediction. The algorithm performs an implicit feature selection using a small subset of "strong variables" which leads to superior performance on high-dimensional data [39]. The results of feature selection can be visualized through Gini index [40] which indicates the importance or feature relevance.

Gini index measures an attribute's impurity with respect to each class. At each node within binary trees of the random forest, the optimal split is achieved using Gini impurity which is computationally efficient as compared to entropy. Gini impurity measures how well a potential split is separating samples of the two classes at a particular node. In addition, Gini impurity indicates the frequency of selecting a specific feature for a split and the extent of its overall discriminative score for the given classification problem. Let T be a given training dataset, selecting an attribute at random and checking whether it belongs to some class C_i; Gini index is computed as in [40]

$$\sum\sum_{j \neq i} \left(\frac{f(C_i, T)}{|T|}\right)\left(\frac{f(C_i, T)}{|T|}\right), \tag{8}$$

where $(f(C_i, T))/|T|$ represents the probability with which the selected attribute belongs to the class C_i. RF selects the best feature among the random subset of features while splitting a node, and it has only two parameters to adjust, i.e., number of variables in a random subset at each node and number of DTs in the forest. RF has many advantages over other ensemble methods. It works well for a large range of items and shows less variance as compared to a single DT. In addition, RF are flexible and output higher accuracy even without scaling of data. However, generating RF is much harder and time-consuming as it requires more computational resources.

3.2. Deep Learning-Based Classifiers. Conventional ML methods are limited in their ability for processing natural data in raw form. DL-based methods are representation learning methods which allow a machine to be fed with the raw data and automatically discover the representations required for classification. In addition, DL exploits simple but nonlinear modules for transforming the representation at raw input level into a representation at higher or abstract level. Several deep learning techniques have been developed so far; however, this study analyzes only convolutional neural network (CNN) given as follows.

3.2.1. Convolutional Neural Network (CNN). DL has proven to be an outstanding classification technique in image/speech recognition and other relevant applications [13]. The classification process in DL begins by training large multilayer neural networks (MLPs), also called deep neural networks (DNN). MLPs are, in fact, feedforward networks which are trained with standard backpropagation algorithm.

Currently, several DL-based techniques have been proposed. However, this paper employs convolutional neural network (CNN) which is a type of DNN. CNNs utilize multiple layers for multiclass predictions such as one or more pairs of max-pooling layer, a convolutional layer, and one or more fully connected hidden layers. These layers use neurons with tanh, maxout, and rectifier functions for the purpose of identifying a set of locally connected neurons. CNNs continuously extract several low-level characteristics into compressed high-level abstractions and representations.

One of the advantages of CNNs includes fewer parameters and easier training as compared to other deep neural networks. In addition, CNNs show higher accuracy using advance features such as adaptive learning, momentum training, rate annealing, dropout, and L1 or L2 regularization techniques. DL has become a popular research domain in recent years. Therefore, a comprehensive performance analysis is required against well-established machine learning techniques. Table 1 presents a comparison between the machine learning algorithms discussed in the study.

4. Experimental Setup

The choice of an appropriate algorithm in response to a specific classification problem is based not only on prior knowledge about the classifiers' performance but also on systematic evaluation in order to replicate and generalize the results. The recent progress in publicly available datasets has led the machine-learning community to effectively validate and share the experimental results. The experiments were performed on Intel Quad-Core i5-82500U at 1.8 GHz with 8 GB RAM, running 64-bit Windows 10 Home Edition. The datasets were partitioned using 10-fold cross-validation.

In this study, 10-fold cross-validation technique [41] is utilized for measuring accuracy of the classifiers. In this setting, the training dataset is divided into ten equal-sized subsets such that each of these ten subsets is tested using the classifier that has been trained on the remaining nine subsets. The advantages of tenfold cross-validation include reduced computation cost as the process is repeated only ten times. In addition, 10-fold cross-validation results in less biasness as compared to other validation techniques where

Table 1: Comparison of ML- and DL-based classifiers.

Classifier	Underlying methodology	Classifier applicability	Nature of prediction/ label class	Advantage(s)	Disadvantage(s)
Naïve Bayes [35]	Bayes theorem	Classification	Categorical	Less parameter tuning, less data learning requirements, computationally fast	Conditional independence between attributes
Decision trees [36]	Iterative Dichotomiser 3 (ID3)	Classification, regression	Categorical, continuous	Simple to interpret, shows higher accuracy	Target attribute must have discrete values; dataset must not have complex and many attributes (i.e., imbalance); uses greedy approach for generating DTs; prone to overfitting
Random forest [12]	Aggregation of (decision) trees using bagging with C4.5 algorithm	Classification, regression	Categorical, continuous	Not susceptible to overfitting, reduces error rate while generating DTs	Generates parallel DTs, computationally slow on large and complex datasets
Gradient boosted trees [37, 38]	Adaptive boosting using C4.5 algorithm	Classification, regression	Categorical, continuous	Boosting reduces error by reducing bias and to some extent variance sequential tree generation with improved learning in each iteration	Uses shallow weak learner trees, computationally faster than RF, harder parameter tuning
Deep learning [13]	Convolutional neural networks	Classification, regression	Categorical, continuous	Higher accuracy sometimes exceeds human-level performance; DL algorithms scale with data; CNNs require relatively little preprocessing	Requires large amounts of labeled data and substantial computing power

each data point is tested for exactly once and is utilized in training (10-1) times.

4.1. Datasets. The performance analysis of the classifiers is visualized on eleven datasets from the popular UCI [42] and Kaggle [43] repositories. Table 2 summarizes the characteristics of the datasets. The datasets can be divided into three categories: small, medium, and large based on number of instances and type of attributes. Datasets having less than 10,000 instances are taken as small datasets. Thus, Horse Colic, Titanic, CTG, Spambase, and NYS Dept. of State Business Filings fall under the category of small datasets. On the other hand, datasets with a number of instances between 10,000 and 50,000 are considered as medium datasets. Therefore, Avila, WHO Suicide Statistics, and Adult datasets are categorized as medium-sized datasets. Finally, datasets with a number of instances between 50,000 and 250,000 are taken as large datasets. The study includes TripAdvisor Restaurant, NYS Nyserda, and Black Friday as the large datasets.

4.2. Performance Evaluation Measures (PEMs). The performance of the classifiers is evaluated using the widely used confusion matrix-based metrics, namely, accuracy, precision, and recall. The confusion matrix represents the relation between predicted values and actual values. Therefore, accuracy, precision, and recall play a significant role in determining an algorithm's strength.

4.2.1. Accuracy. The accuracy of a classifier is computed as the number of correctly predicted instances divided by total number of predictions. In other words, accuracy is the overall percentage of correctly predicted values given as

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{9}$$

where TP and TN represent true positive and true negative, respectively. Similarly, FP and FN represent false positive and false negative, respectively. TP and TN show that model predictions agree with the original class values whereas FP and FN indicate the incorrect prediction of the model as compared to original class values.

4.2.2. Precision. Precision represents exactness, and it shows the percentage of correctly predicted positive results (i.e., TP) from all positive predictions given as

$$\text{Precison} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{10}$$

Table 2: Characteristics of selected datasets.

Dataset	No. of attributes	No. of instances	Attribute types	No. of prediction classes	Dataset library
Small datasets					
Horse Colic [42]	27	368	Categorical, integer, real	02	UCI
Titanic [43]	12	891	Categorical, integer, real	02	Kaggle
CTG [42]	23	2126	Real	03	UCI
Spambase [42]	57	4601	Integer, real	02	UCI
NYS Dept. of State Business Filings [43]	24	9745	Categorical, integer	10	Kaggle
Medium-sized datasets					
Avila [42]	10	20867	Real	10	UCI
WHO Suicide Statistics [43]	6	43800	Categorical, integer	06	Kaggle
Adult [42]	14	48842	Categorical, integer	02	UCI
Large-sized datasets					
TripAdvisor Restaurant [43]	11	126000	Categorical, integer, real	07	Kaggle
NYS Nyserda [43]	23	223000	Categorical, integer, real	06	Kaggle
Black Friday [43]	11	234000	Categorical, integer	10	Kaggle

Precision is an effective measure to determine the cost associated with false positives. For example, detecting spam emails, a false positive indicates the number of nonspam emails which are identified as spam.

4.2.3. Recall. Recall answers what percent of positive cases is predicted correctly. Recall is also referred to as the true-positive rate given as

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{11}$$

5. Results and Discussion

The section presents results of the classifiers discussed in this study using the performance measures of accuracy, precision, and recall. Figure 1 presents the accuracy of the classifiers. NB shows more than 80% accuracy on the Adult dataset and more than 60% accuracy on NYS Nysedra, Horse Colic, CTG, and Spambase datasets. However, the accuracy results of NB are below 50% on Avila, TripAdvisor Restaurants Info, and Titanic datasets, respectively. A possible explanation of such behavior could be that Adult dataset is primarily created for binary classification whereas NYS Nysedra, Horse Colic, CTG and Spambase datasets have multiple types of attributes and these datasets are mainly designed for multiclassification. On the other hand, Avila and TripAdvisor Restaurants Info are complex datasets with several prediction classes and multiple types of attributes. Surprisingly, NB shows lower accuracy on the Titanic dataset. One of the key reasons is that the Titanic dataset has different proportions of missing values in different attributes.

The accuracy of DTs on Titanic, Spambase, and NYS Dept. of State Business Fillings datasets is above 90%. Similarly, DTs show more than 80% accuracy on the Adult dataset and more than 70% accuracy on Horse Colic and Black Friday datasets, respectively. However, the accuracy of DTs is below 60% on Avila and WHO Suicide Statistics datasets. DTs effectively analyze the statistical relationship between a given input and output. Therefore, DTs show higher accuracy overall as compared to NB on multiclass datasets and datasets with missing values.

On the other hand, ensemble-based GBDTs show more than 70% on Horse Colic and WHO Suicide Statistics datasets and above 80% accuracy on CTG and Adult datasets. Similarly, the accuracy of GBDTs is above 90% on Titanic, Spambase, Avila, and Black Friday datasets, respectively. GBDT employs bootstrap bagging to integrate weak leaners for overall improvement. Therefore, GBDT shows higher accuracy as compared to DTs, particularly on complex and multiclass datasets such as Avila.

Similarly, RF is another ensemble approach which shows more than 90% accuracy on Titanic, Spambase, and NYS Dept. of State Fillings datasets; more than 80% accuracy on Adult and Black Friday datasets; and more than 70% accuracy on the Horse Colic dataset, respectively. However, accuracy results are below 65% on Avila, CTG, and WHO Suicide Statistics. While comparing with DTs, RF shows small improvements on Avila, CTG, and WHO Suicide Statistics datasets. However, on datasets having noisy classification or regression, RF shows overfitting tendency and in the

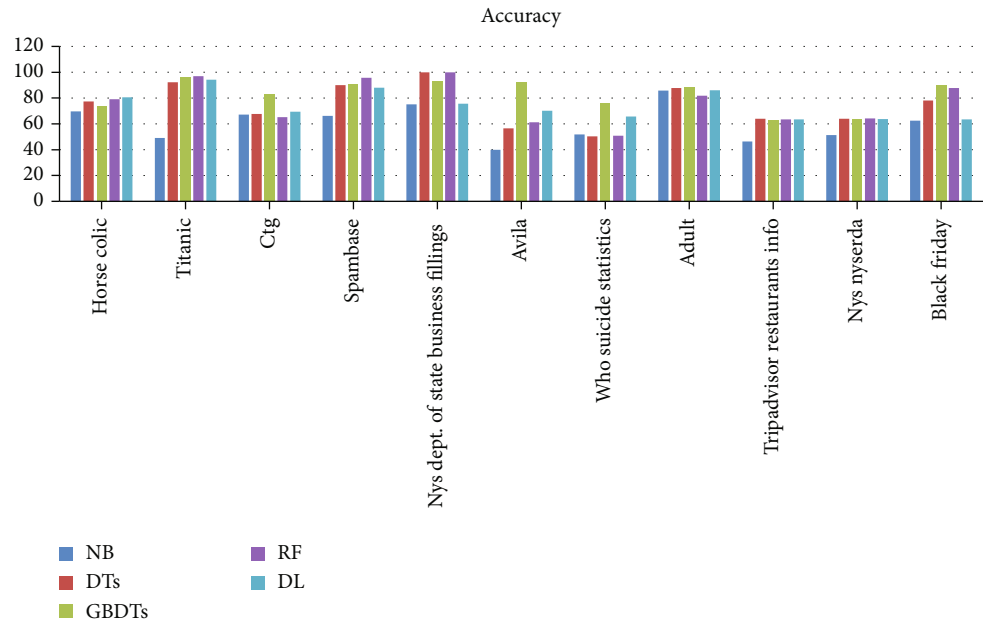

Figure 1: Accuracy of the classifiers.

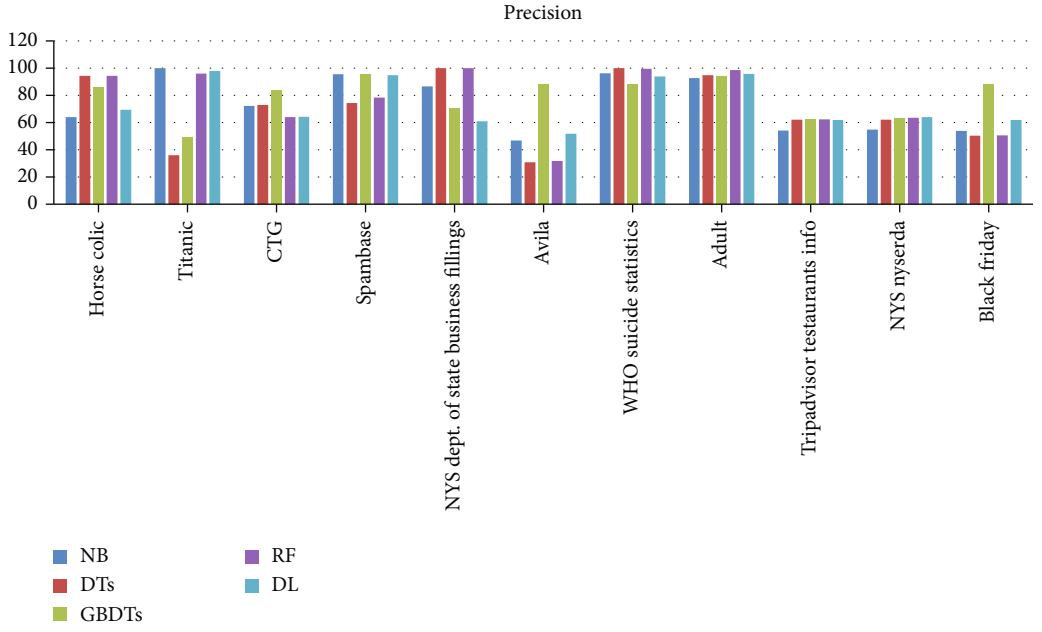

Figure 2: Precision of the classifiers.

case of categorical attributes with different numbers of levels, RF favors those attributes having more levels. This behavior is evident on Avila, CTG, and WHO Suicide Statistics datasets where GBDT shows higher accuracy as compared to RF.

Finally, convolutional neural networks (CNN) show more than 90% accuracy on the Titanic dataset; more than 80% accuracy on Horse Colic, Spambase, and Adult datasets; and more than 70% accuracy on NYS Dept. of State Fillings and Avila datasets, respectively. On small datasets, the performance of CNN suffers as DL-based classifiers are slow to train [44]. On the other hand, RF requires tuning of fewer hyperparameters which makes RF a faster algorithm. Therefore, on small datasets such as Horse Colic and NYS Dept. of State Fillings, RF outperforms CNN in terms of accuracy. One of the drawbacks of RF is that it often yields suboptimal performance on large-scale datasets using the greedy approach of tree construction [45]. Therefore, on Avila and Black Friday datasets, RF shows lower accuracy results as compared to CNN.

Surprisingly, GBDTs outperform all other classifiers discussed in this study in terms of accuracy, particularly, on datasets with multiclassification and missing values. A

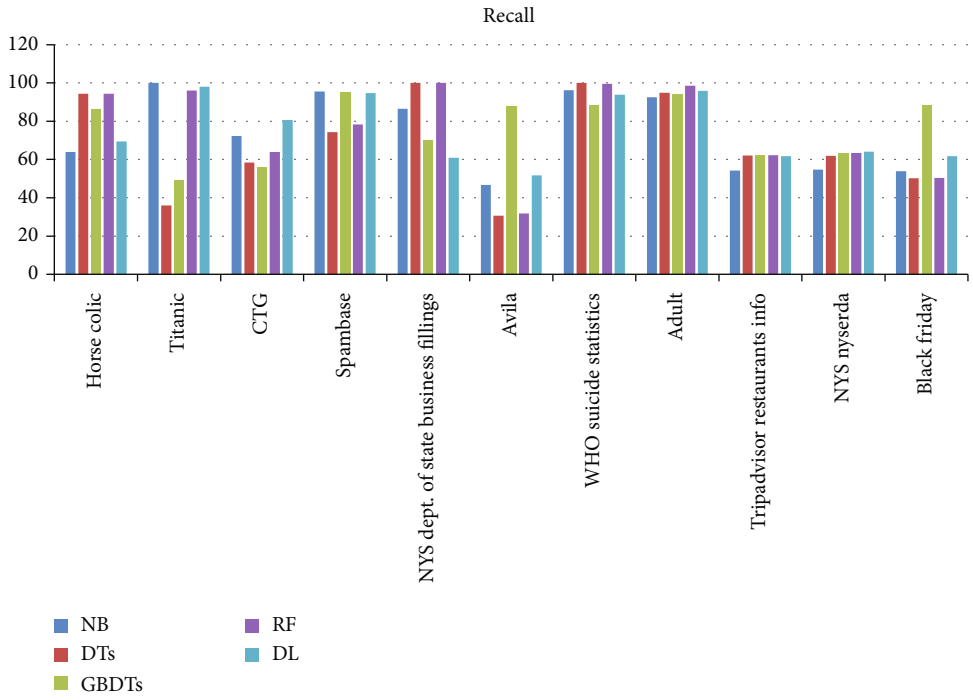

Figure 3: Recall score of the classifiers.

possible explanation of such phenomena could be that GBDTs utilize the concept of boosting to shallow the classification trees which results in model simplicity and tuning is limited to the parameters of the gradient boosting algorithms itself. GBDT performs optimization in function space that results in flexible use of custom loss function. In addition, boosting is computationally efficient as compared to deep learning [46–48].

Figure 2 presents the precision results of the classifiers. NB shows more than 90% precision on Titanic, Spambase, and WHO Suicide Statistics; more than 80% precision on NYS Dept. of State Fillings; more than 70% precision on CTG; and below 60% precision on the rest of the datasets. The precision results are different from the accuracy results as shown in Figure 1. This is because precision is independent of accuracy and is concerned with the fraction of positive predictions which are identified as positive in actual. On the other hand, accuracy is simply the fraction of total sample that is correctly identified.

The precision of DTs on Horse Colic, NYS dept. of State Fillings, WHO Suicide Statistics, and Adult datasets is above 90%. Similarly, DTs show more than 70% precision on CTG and Spambase datasets and below 50% precision on Titanic, Avila, and Black Friday datasets, respectively.

It is worth mentioning that DTs show more than 90% accuracy on the Titanic dataset. However, the precision results of DTs on the Titanic dataset are below 40%. On the other hand, NB shows more than 90% precision on the Titanic dataset. However, the accuracy of NB on the Titanic dataset is below 50%. One of the prime reasons is that NB is a simple algorithm less prone to overfit. On the other hand, DTs suffer from the inability to grasp the relationship between features of the dataset and utilize the greedy learning approach which leads to suboptimal model [38]. Overall, DTs efficiently predict the true positives; therefore, precision results on DTs are higher on most datasets as compared to NB.

On the other hand, GBDTs show more than 60% precision on NYS Dept. of State Fillings, TripAdvisor, and NYS Nysedra datasets and more than 80% precision on Horse Colic, CTG, Avila, WHO Suicide Statistics, and Black Friday datasets. Similarly, precision results are above 90% on Spambase and Adult datasets. The precision results are below 50% on the Titanic dataset. GBDTs show higher precision on complex datasets such as Avila and Black Friday and datasets with missing values such as Titanic as compared to DTs. Similarly, ensemble-based RF shows almost the same behavior as DTs, except on Titanic and Black Friday datasets where RF shows higher precision. Comparing with GBDTs, RF shows lower precision results which show that GBDT classifies the true positives efficiently as compared to RF on complex datasets and datasets with missing values.

Finally, CNN shows more than 60% precision on Horse Colic, CTG, NYS Dept. of State Fillings, TripAdvisor, and Black Friday datasets. Similarly, precision results are above 90% on Titanic, Spambase, and WHO Dept. of State Fillings datasets. However, CNN shows below 50% precision on the Avila dataset. Comparing with RF, CNN shows mixed performance results. On the Horse Colic dataset, CNN shows significantly lower precision results. However, precision results of CNN are significantly higher on the Spambase dataset. Comparing with GBDTs, CNN shows significantly

lower overall precision which indicates that GBDTs can output higher precision as compared to CNN on large, multiclass datasets.

Figure 3 presents the recall of the classifiers. The recall of NB, DTs, GBDTs, RF, and CNN is almost similar to their precision results as shown in Figure 2. A possible explanation of this phenomenon is that both precision and recall are concerned with correctly identified positive predictions. The difference lies in that in precision, the correctly identified positive predictions are from the total positive predictions whereas in recall, the correctly identified positive predictions are from actual positive values. Overall, GBDTs shows higher recall as compared to other classifiers. DTs show more than 90% recall on Horse Colic, NYS Dept. of State Fillings, WHO Suicide Statistics, and Adult datasets. The recall is above 60% on Spambase, TripAdvisor, and NYS Nyserda datasets. However, the classifier shows below 40% recall on Avila and Titanic datasets, respectively. The recall results of DTs are almost similar to the precision results on the datasets analyzed in this study.

The recall of GBDTs on Spambase and Adult datasets is above 90%. The classifier shows more than 80% recall on Horse Colic, Avila, WHO Suicide Statistics, and Black Friday datasets. On NYS Dept. of State Fillings, TripAdvisor, and NYS Nyserda, the recall is above 60%. However, GBDTs shows below 50% recall on the Titanic dataset. The recall results of GBDTs are nearly similar to its precision on the datasets discussed in this study.

RF shows nearly similar recall results as precision on the datasets used for experimental setup. For instance, on Horse Colic, Titanic, NYS Dept. of State Fillings, WHO Suicide Statistics, and Adult datasets, the recall is above 90%; more than 60% recall on CTG, Spambase, Trip Advisor, and NYS Nyserda datasets; and below 40% recall on the Avila dataset, respectively.

Similarly, the recall results of CNN are almost similar to its precision results on the datasets tested. For example, on Titanic, Spambase, and WHO Dept. of State Fillings datasets, the recall is above 90%. CNN shows more than 60% recall on Horse Colic, CTG, NYS Dept. of State Fillings, TripAdvisor, and Black Friday datasets. On the other hand, recall is below 50% on the Avila dataset. As the ML-based NB and DTs, ensemble-based GBDTs and RF and DL-based CNN show nearly similar recall results; therefore, the recall of GBDTs is higher than that of the other classifiers.

6. Conclusions

The study presents a state-of-the-art comparative analysis of machine learning and deep learning-based algorithms for multiclass prediction. The study can serve as a guideline for new researchers in selecting a baseline algorithm or proposing an ensemble-based technique using any of the classifiers examined in this study. The study evaluates the performance of the classifiers using statistical measures such as accuracy, precision, and recall and shows that gradient boosting decision trees (GBDTs) outperform other classifiers discussed in this study. Similarly, decision trees (DTs) show significantly better performance as compared to classic Naïve Bayes (NB). On small datasets, random forest (RF) shows higher accuracy, precision, and recall scores as compared to convolutional neural networks (CNN). However, on large and regression-based datasets, CNN outperforms RF. The results show that DTs and RF suffer serious performance issues in the case of large and complex datasets due to the underlying greedy approach and overfitness. In the future, we plan to extend this work to include other classifiers and evaluate their performance on significantly large text datasets and image data.

In future work, we plan to apply diverse deep learning (DL) algorithms on larger datasets in addition to the datasets mentioned above. We plan to compare the performance of DL algorithms such as Long Short-Term Memory Networks, Recurrent Neural Networks, and Generative Adversarial Networks using multiple evaluation metrics.

Data Availability

Since the funding project is not closed and related patents have been evaluated, the simulation data used to support the findings of this study are currently under embargo, while the research findings are commercialized. Requests for data, upon the approval of patents after project closure, will be considered by the corresponding author.

Disclosure

The granting agencies did not contribute in the design of the study and collection, analysis, and interpretation of data.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors thank the Natural Sciences and Engineering Research Council of Canada (NSERC) and New Brunswick Innovation Foundation (NBIF) for the financial support of the global project. These granting agencies did not contribute in the design of the study and collection, analysis, and interpretation of data.

References

[1] Z. Chen and B. Liu, "Lifelong machine learning," *Synthesis Lectures on Artificial Intelligence & Machine Learning*, vol. 10, no. 3, pp. 1–145, 2016.

[2] V. Grover, R. H. Chiang, T.-P. Liang, and D. Zhang, "Creating strategic business value from big data analytics: a research framework," *Journal of Management Information Systems*, vol. 35, no. 2, pp. 388–423, 2018.

[3] B. Zou, V. Lampos, and I. Cox, "Multi-task learning improves disease models from web search," *World Wide Web*, pp. , 201887–96, 2018.

[4] D. Geol, J. M. Khandelwal, and R. Tiwari, "Intelligent and integrated book recommendation & best price identifier system using machine learning," in *Intelligent Engineering Informatics*, pp. 397–412, Springer, Berlin/Heidelberg, Germany, 2018.

[5] L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.

[6] P. Chahuara, N. Grislain, G. Jauvion, and J.-M. Renders, "Real-time optimization of web publisher RTB revenues," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1743–1751, New York, United States, 2017.

[7] S. Bhatia, P. Sharma, R. Burman, S. Hazari, and R. Hande, "Credit scoring using machine learning techniques," *International Journal of Computer Applications*, vol. 161, no. 11, pp. 1–4, 2017.

[8] D. Nawrocka, *Machine learning for trading and portfolio management using Python*, 2018, Hochschulbibliothek HWR Berlin.

[9] B. Huang, Y. Huan, L. D. Xu, L. Zheng, and Z. Zou, "Automated trading systems statistical and machine learning methods and hardware implementation: a survey," *Enterprise Information Systems*, vol. 13, pp. 1–13, 2018.

[10] J. Mata, I. de Miguel, R. J. Duran et al., "Artificial intelligence (AI) methods in optical networks: a comprehensive survey," *Optical Switching and Networking*, vol. 28, pp. 43–57, 2018.

[11] T. O. Ayodele, "Types of machine learning algorithms, in new advances in machine learning," Portsmouth, United Kingdom, 2010.

[12] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[13] Y. Bengio, "Learning deep architectures for AI," *Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[14] S. Amiri, M. Akbarabadi, F. Abdolali, A. Nikoofar, A. J. Esfahani, and S. Cheraghi, "Radiomics analysis on CT images for prediction of radiation-induced kidney damage by machine learning models," *Computers in Biology and Medicine*, vol. 133, article 104409, 2021.

[15] M. Loreto, T. Lisboa, and V. P. Moreira, "Early prediction of ICU readmissions using classification algorithms," *Computers in Biology and Medicine*, vol. 118, article 103636, 2020.

[16] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proceedings of the 23rd international conference on Machine learning*, pp. 161–168, Pittsburgh, Pennsylvania, USA, 2006.

[17] A. Cieslak and N. V. Chawla, "Learning decision trees for unbalanced data," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 241–256, Springer, Berlin, Heidelberg, 2008.

[18] I. Brown and C. Mues, "An experimental comparison of classification algorithms for imbalanced credit scoring data sets," *Expert Systems with Applications*, vol. 39, no. 3, pp. 3446–3453, 2012.

[19] L. Thomas, J. Crook, and D. Edelman, "Credit scoring and its applications," in *Society for industrial and Applied Mathematics*, vol. 2, Siam, Philadelphia, USA, 2017.

[20] J. Howard, "The business impact of deep learning," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1135–1135, August 2013.

[21] D. W. 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.

[22] M. Sewak, S. K. Sahay, and H. Rathore, "Comparison of deep learning and the classical machine learning algorithm for the malware detection," in *International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, pp. 293–296, Busan, Korea, June 2018.

[23] J. Abellán and J. G. Castellano, "A comparative study on base classifiers in ensemble methods for credit scoring," *Expert Systems with Applications*, vol. 73, pp. 1–10, 2017.

[24] S. Lessmann, B. Baesens, H.-V. Seow, and L. C. Thomas, "Benchmarking state-of-the-art classification algorithms for credit scoring: an update of research," *European Journal of Operational Research*, vol. 247, no. 1, pp. 124–136, 2015.

[25] A. C. Lorena, L. F. Jacintho, M. F. Siqueira et al., "Comparing machine learning classifiers in potential distribution modelling," *Expert Systems with Applications*, vol. 38, no. 5, pp. 5268–5275, 2011.

[26] J. I. Li and C. Beier, "Machine learning approaches for forest classification and change analysis using multi-temporal Landsat TM images over Huntington Wildlife Forest," *GIScience & Remote Sensing*, vol. 50, no. 4, pp. 361–384, 2013.

[27] N. Macià and E. Bernadó-Mansilla, "Towards UCI+: a mindful repository design," *Information Sciences*, vol. 261, pp. 237–262, 2014.

[28] S. K. Onan and H. Bulut, "Ensemble of keyword extraction methods and classifiers in text classification," *Expert Systems with Applications*, vol. 57, pp. 232–247, 2016.

[29] A. Onan, "Mining opinions from instructor evaluation reviews: a deep learning approach," *Computer Applications in Engineering Education*, vol. 28, no. 1, pp. 117–138, 2020.

[30] A. Onan, "Topic-enriched word embeddings for sarcasm identification," in *Computer Science On-line Conference*, pp. 293–304, Springer, Cham, 2019.

[31] A. Onan and M. A. Toçoğlu, "A term weighted neural language model and stacked bidirectional LSTM based framework for sarcasm identification," *Access*, vol. 9, pp. 7701–7722, 2021.

[32] A. Onan, "Hybrid supervised clustering-based ensemble scheme for text classification," *Kybernetes*, vol. 46, no. 2, pp. 330–348, 2017.

[33] F. Ma, G. Meng, H. Yan, B. C. Yan, and F. Song, "Diagnostic classification of cancers using extreme gradient boosting algorithm and multi-omics data," *Computers in Biology and Medicine*, vol. 121, article 103761, 2020.

[34] X. Wang, Y. Zhang, B. Yu et al., "Prediction of protein-protein interaction sites through eXtreme gradient boosting with kernel principal component analysis," *Computers in Biology and Medicine*, vol. 134, article 104516, 2021.

[35] A. McCallum and K. Nigam, "A comparison of event models for naive Bayes text classification," *AAAI-98 workshop on learning for text categorization*, vol. 752, pp. 41–48, 1998.

[36] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.

[37] J. H. Friedman, "Stochastic gradient boosting," *Computational Statistics & Data Analysis*, vol. 38, no. 4, pp. 367–378, 2002.

[38] J. Ye, J.-H. Chow, J. Chen, and Z. Zheng, "Stochastic gradient boosted distributed decision trees," in *Proceedings of the 18th ACM conference on Information and knowledge management*, pp. 2061–2064, November 2009.

[39] R. Genuer, J.-M. Poggi, and C. Tuleau-Malot, "Variable selection using random forests," *Pattern Recognition Letters*, vol. 31, no. 14, pp. 2225–2236, 2010.

[40] A. Liaw and M. Wiener, "Classification and regression by random forest," *R news*, vol. 2, pp. 18–22, 2002.

[41] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural Computation*, vol. 10, no. 7, pp. 1895–1923, 1998.

[42] K. T. Dheeru, *Machine Learning Repository*, School of Information and Computer Sciences, University of California Irvine, 2017.

[43] *Kaggle dataset library*, 2021, http://www.kaggle.com/).

[44] J. Schmidhuber, "Deep learning in neural networks: an overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[45] Z.-H. Zhou and J. Feng, "Deep forest: towards an alternative to deep neural networks," pp. 3553–3559, 2017, http://arxiv.org/abs/ 1702.08835.

[46] E. B. Sudakov and D. Koroteev, "Driving digital rock towards machine learning: predicting permeability with gradient boosting and deep neural networks," *Computers & Geosciences*, vol. 127, pp. 91–98, 2019.

[47] X. Wang, S. Yin, M. Shafiq et al., "A new V-net convolutional neural network based on four-dimensional hyperchaotic system for medical image encryption," *Networks*, vol. 2022, pp. 1–14, 2022.

[48] T. Shahwar, J. Zafar, A. Almogren et al., "Automated detection of Alzheimer's via hybrid classical quantum neural networks," *Electronics*, vol. 11, no. 5, p. 721, 2022.