

An extensible framework for recurrent breast cancer prognosis using deep learning techniques

Reddy Shiva Shankar, Ravi Swaroop Chigurupati, Priyadarshini Voosala, Neelima Pilli

Department of Computer Science and Engineering, Sagi Ramakrishnam Raju Engineering College, Bhimavaram, India

Article Info

Article history:

Received Jun 14, 2022

Revised Oct 5, 2022

Accepted Oct 14, 2022

Keywords:

Adaboost classifier

Breast cancer

Histogram-based gradient boosting classifier

Machine learning

Multi-layer perceptron

Wisconsin breast cancer dataset

ABSTRACT

Due to population growth, early illness detection is getting more challenging. Breast cancer is the second-deadliest malignancy. An estimated one million people are newly diagnosed with the disease annually in India. Most cases are never diagnosed because they are either ignored or not reported. Also, secondary malignancies may develop after a breast cancer recurrence, including those of the brain, lungs, and bones. Early detection and treatment of people with recurring breast cancer may help prevent secondary cancers and other disorders. By examining cell and tumour data as well as data from other diseases, this project hopes to overcome this obstacle and more accurately diagnose breast cancer. Accurate diagnosis of breast cancer may be achieved with the use of machine learning techniques. The effort focuses on recurring breast cancer and aims to efficiently identify it. In ensemble learning, decision trees filter out non-essential qualities. Cancer recurrences and non-recurrences are distinguished using voting classifiers. The soft voting classifier classifies a variety of data sets with 98.24% accuracy. The proposed model has an accuracy of 0.97, a recall of 0.97, an F1-Score of 0.969, and a Choen kappa score of 0.9655, as stated by the recommended model.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Reddy Shiva Shankar

Department of Computer Science and Engineering, Sagi Ramakrishnam Raju Engineering College

West Godavari District, Chinnaamiram, Bhimavaram, Andhra Pradesh, India

Email: shiva.cesrkr@gmail.com

1. INTRODUCTION

Cancer is caused by gene alterations, such as mutations that control cell growth. Cells will divide and multiply in an uncontrolled manner as a result of the alterations. Breast cancer (BC) is a Cancer that starts in the cells of the breast. Cancer usually develops in the lobules or ducts of the breasts. These unregulated cells move to lymph nodes under the arms and infect other breast tissue. The lymph nodes serve as a conduit for cancer cells to travel to other body parts. Both men and women are susceptible to BC. Symptoms vary depending on the type of BC. Many are similar, but a few are distinct [1].

BC that returns after therapy is known as recurrent breast cancer (RBC). Local recurrence refers to Cancer that recurs in the exact location as the original Cancer, whereas distant recurrence refers to Cancer that spreads to other body parts. The signs and symptoms of recurrent BC differ depending on where it returns [2]. BC can be staged from 0 to 4, with subcategories at each numbered step. A description of the four main stages is listed in:

- Stage 0: Ductal carcinoma in SITU (DCIS) refers to cancer cells that have only infiltrated the ducts and have not spread to the surrounding tissues.

- Stage 1: The tumour is upto 2 cm in diameter at this stage. No lymph nodes have been damaged, or there are small groupings of cancer cells in lymph nodes.
- Stage 2: The tumour is 2cm in diameter and has begun to spread to adjacent lymph nodes, or it is 2-5 cm in diameter and has not migrated to lymph nodes.
- Stage 3: The tumour has expanded to numerous lymph nodes and is upto 5cm across, or the tumour is more significant than 5cm and has spread to several lymph nodes.
- Stage 4: The Cancer has progressed to other organs, most commonly the bones, the liver, the brain, or the lungs [3]

Machine learning (ML) can be used to work with data related to tumours and learn from data to detect BC. To train and test, data is needed. Hence, the information is collected from sci-kitlearn and datahub.com sources. The dataset consists of 2 classes. They are recurrent events and non- recurrent events. As a result, two datasets were initially gathered to create the suggested system, one from unique client identifier (UCI) and the other from datahub. The two datasets need to be integrated based on standard features. After merging them into a single entity, we got our hands on a dataset with 49 features. Now this entire dataset had to get preprocessed and cleansed. Later the data was analyzed, and feature engineering was performed on the dataset. The dataset is currently ready, and various algorithms can be performed to find a suitable algorithm that gives the best accuracy without over-fitting or under-fitting the model [4]. Voting Classifier was proposed for this model; algorithm selection was crucial, so VC suited the dataset perfectly. Firstly, we apply a few assembling models to the dataset and observe their performance and results. We pick promising models and apply a voting algorithm to them. Here we used soft voting in the VC that calculates the mean accuracies of all the applied models. While applying this algorithm, we shall change values to many attributes such as activation function, random state, maximum iterations etc. The accuracy of the model varies depending on the importance of the features, and we eventually have a model that recognizes whether a patient's BC is recurrent or not [5]. As a result of this model determining whether the patient's Cancer is a recurring event, they can take safeguards before the disease spreads to another organ or infects the same organ. It may lead people to get more frequent scans and catch tumours before they spread.

2. RELATED WORK

For classification, Kim *et al.* [6] utilized the support vector machine (SVM), Cox-proportional hazard regression model, and artificial neural network (ANN) classifiers. The tool used to select the essential features from the dataset was the normalized mutual information index. The dataset contained data collected from 679 patients. The statistical numbers of the proposed system were found to be 89% which is acceptable for a medical model. In a sample size of 547 patients, Ahmad *et al.* [7] used SVM, decision tree (DT), and multilayer perceptron (MLP) ANN classifiers with feature selection. Among all the selected classifiers, SVM was the best, as it showed 96% sensitivity, 91% specificity and 94% accuracy. The national provider identifier (NPI) was proposed by Galea *et al.* [8], a predictive regression model based on tumour size, histological grade, and lymph node status. The NPI was calculated using: (tumour size 0.2) + histological grade + lymph node point. The index was based on nine factors collected from 387 patients. The index was subsequently validated in a study of 320 patients.

The patients were classified as benign and malignant based on the NP index. Laghmati *et al.* [9] had an overview that early tumour detection could help implement better and quicker recovery techniques. For this, they had the idea of using computer-aided design (CAD) systems. Doctors and radiologists use these CAD systems to detect and analyze whether a patient's condition is severe or not. Ludin *et al.* [10] studied the topic of 5-year, 10-year and 15-year BC patients and their survival. They collected a total of 900 patient data, using 651 to train the model, whereas the remaining 300 were used to validate the model. They used neural networks for the model that they proposed. The database consists of 8 features that are equally important to predict. The AUCROC when neural network (NN) was used was 0.909 for 5-year patients, 0.886 for 10-year patients and 0.883 for 15-year patients.

Mishra *et al.* [11] studied and analyzed that efficient and early detection of Cancer can be aided in diagnosing Cancer properly, which gives a better survival rate. The proposed model explains that clinical decision support systems are potential utility for medical practitioners. The accuracy for BC Coimbra Dataset was 80.83%, and for the breast cancer wisconsin diagnostic (BCWD) dataset, 98.24%. Lohoura *et al.* [12] proposed an ML model trained on the datasets with the help of ANN. Particularly in this ANN, they selected an extreme learning machine (ELM) model, which performs even better. This cloud-based ELM has outperformed every other model when applied to the BC dataset. Rathore *et al.* [13] examined three supervised learning algorithms for BC prediction: DT, NB, and Association-based classification, and concluded that the ensemble technique is the best approach. Combining three categorization techniques in this suggested system is hoped to achieve more accuracy. BC is a major health problem and one of the leading causes of death among women,

according to Goayl *et al.* [14]. When Cancer returns after a few years of treatment, it is known as recurrence. Early cancer detection and prognosis are important factors in red blood cell (RBC) and can aid medical treatment. Various classifiers were utilized in the proposed model to predict the RBC. GRNN achieved 83.33%, FFBN 85.18%, SVM 77.77%, DT 70.83% and NB 72.22% in accuracy. For the prediction of BC, Shah *et al.* [15] employed three different DM categorization algorithms. Their Research had an overview of all the algorithms that modelled the data and got results in which they concluded that NB is a better algorithm when compared to DT or KNN because it had lower computation time and better accuracy than the features of superior prediction.

The Wisconsin dataset was utilized by Ojha *et al.* [16] to create an efficient predictor algorithm for predicting the recurrent or nonrecurrent character of the disease. It helped oncologists distinguish between a favourable, nonrecurrent prognosis and a lousy forecast, which means recurrent, allowing them to treat patients more efficiently. K-means, EM, PAM, and Fuzzy c-means were among the clustering algorithms, while SVM, C5.0, k-nearest neighbors (KNN), and NB were among the classification techniques. Boeri *et al.* [17] believe that the quick detection of BC and its early diagnosis has saved many lives worldwide. Even though multigene signature panels and the NPI were explored as solutions, they both attempted to provide a fresh multidisciplinary approach. Chauraisa *et al.* [18] proposed NB, J48 DT and Bagging algorithm for BC Prediction and Survivability. UC Irvine ML repository is used to predict Cancer.

WEKA tool is used for implementing the project. The advantage of implementing this technique on the holdout sample, DT (C5) is the best predictor; its prediction accuracy is higher than any previously reported in the literature, and its accuracy is up to 96.5%. The BChybrid model proposed by Sivakami *et al.* [19] combines DT and SVM algorithms. This methodology divided patients into two groups (Benign and Malignant). The dataset, which has eleven features, was gathered from the WBCD dataset, which is part of the UCI ML repository and contains 699 cases, 241 of which are malignant and 458 of which are benign. The dataset has missing values in sixteen places. Nikhilanand Arya *et al.* [20] present gated attentive DL models stacked with RF classifiers to improve BC prognosis prediction using multi-modal data and informative characteristics. It's a model with two stages: the first stage uses a sigmoid-gated attention CNN to create stacked features, and the second stage sends those features to an RF classifier. Reddy *et al.* [21] did their work on various diseases like Diabetes [22] and whether its correlated ailments are affected or not. The patient, once discharged from the hospital, are readmitted with multiple causes [23], and they predict whether the patients are diagnosed clearly or not [24] by using ML and DL models [25]. The NELM particle swarm optimization (PSO) algorithm indicated whether the DM patient was affected by BC [26]. Gopal *et al.* [27] propose a method for conducting early BC diagnostics utilizing the IoT and ML. The classifier's minimum error rate was 34.21%, 45.82%, and 64.47% in terms of MAE, RMSE, and RAE, respectively.

Islam *et al.* [28] used the logistic regression technique for BC Prediction. And the Electronic health records dataset is used for predicting BC. Using the electronic health records (ERH) dataset, they applied the Logistic regression technique for an accurate result. WEKA tool is used for implementing the project. The main advantage of these projects is 5-year survivability prediction using logistic regression. Using the Logistic regression technique with 96.4% Accuracy, they achieved and 0.33 error rate. Kumar *et al.* [29] considered BC to be the top-rated type of Cancer amongst women, and BC was the reason for the death of 627,000 odd women. The prime reason for this death rate is the late detection of BC that outlays late diagnosis, causing the Cancer to spread and become malignant. They found that the Data mining techniques were a multi-fold to detect BC and predict its malignancy 12 different algorithms were applied. Ada Boost M1, J-Rip, J48, and Lazy IBK.

Koh *et al.* [30] presented a paper that evaluated 3-DI radionics features of breast magnetic resonance imaging (MRI). These 3 features were considered prognostic factors to predict systemic recurrence in triple-negative BC. The results obtained from the model were validated with a different MRI scanner to perform scrutiny perfectly. The dataset consisted of 182 training data images and 49 images for validation with a Philips scanner. According to Mohebian *et al.* [31], Cancer is a group of disorders characterized by the growth of aberrant cells with the ability to invade or spread throughout the body. For comparison, SVM, DT, and MLPNN were employed. In the full cross-validation folds and the holdout test fold, the minimum sensitivity was 77%, specificity was 93%, Precision was 95%, and accuracy was 85%. Shiva *et al.* [32] used tumour data and applied AI and ML to improve diagnosis accuracy.

3. METHOD

3.1. Objectives

The objectives of the proposed work are:

- To collect suitable datasets specific to the fields related to BC diagnosis and other diseases.
- To perform data analysis and feature engineering by evaluating various ML algorithms.
- Detect whether the BC is Recurrent or Non-recurrent.

3.2. Dataset description

The dataset used for the proposed models consists of data collected from 570 patients; for each patient, there is various ground on which the features are collected. Firstly, these features are divided into three categories: i) Study of cancer cells; ii) Survey of the entire tumour; iii) Details regarding any other health complications.

The dataset is a combination of 3 datasets merged on two standard features, i.e., age and severity of Cancer. The sources approve these datasets, and few modifications are based on statistical numbers from various articles. The dataset is a combination of multiple features which helps us to detect whether the patient's Cancer currently suffering might re-occur or not. Here, feature selection is essential, which might change the accuracy of various models. The description of a few features in the dataset are shown in Table 1.

Table 1. Dataset description

| S.No | Feature name | Range | Description |
|------|-------------------|-------------------------|---|
| 1 | Age | [10 to 99] | Depicts the age of the patient |
| 2 | Menopause | [lt40, ge40, premono40] | At which stage does the patient face menopause? |
| 3 | Deg-Malig | [1 to 3] | The degree of malignancy of the tumour. |
| 4 | Node Caps | [Yes, No] | Whether node caps are present or not. |
| 5 | Brain Tumor | 1 or 0 | If the patient is affected by a brain tumour. |
| 6 | Pancreatic Cancer | 1 or 0 | If the patient has pancreatic Cancer. |
| 7 | Heart Arrhythmia | 1 or 0 | If the patient has a heart arrhythmia. |
| 8 | Radius mean | [6 to 29] | The standard of cancer cell radius. |
| 9 | Area Mean | [143 to 2502] | The mean of cancer cell area. |
| 10 | Smoothness Worst | [0 to 1] | The worst smoothness of the cancer cells. |

The first dataset is obtained from the breast cancer Wisconsin (Diagnostic) data set from the UCI Repository, which contains data regarding the tumour cells of the BC. The second dataset is collected from BC Data from datahub, an instance for study regarding the tumour's physical properties and characteristics. The third is a self-developed dataset that was created after taking statistical data and various studies on diseases caused by BC.

3.3. Data preprocessing and noise removal

The dataset has many columns with strings as their attributes, and we cannot use ML classification or clustering algorithms. Firstly, we had to find each feature's unique lines and convert them into integers or float values. For this task, we used the replace function for all the columns consisting of string values. Also, we found there were NULL values or Nan, and we had two options: remove the entire row or fill it with its mean value. We decided to fill the cells with the mean value of the column as these columns were not in the preliminary list.

3.4. Dataset analysis and feature recognition for splitting into train and test data

The data has been preprocessed, and there are no string values except for the target column. Now it was time to select essential and salient features. For this process, sklearn. feature selection library was imported to segregate the top 10 features and the least 10 features. Later, the target column selected for the model was the class, which tells whether the event will recur. Then, the entire data is divided into train and test data to make the further process easier. Then various ML and DL algorithms are applied to the split dataset to analyze how the dataset works and gives us results.

3.5. Applied models

3.5.1. Adaboostclassifier algorithm

In the AdaBoost classifier model, we first create decision trees. Before it, weights are initialized to check the incorrectly classified model; since the training dataset consists of 47 features, 47 trees are initiated to make it the stump, just as we do in the decision tree. Out of these 47 features, the model must select one base feature, and for that, it calculates Gini and Entropy for all the 47 trees created, and the element having the most negligible value will be the base feature. Now the AdaBoost classifier comes into play; the base feature sees the number of records it classified correctly, and the total error is calculated. Now the performance of the first feature of our tree is calculated, also called the significance value. Now the sample weights are updated using this significance value. The whole dataset is updated using these weights, and the tree formation occurs again. This process takes place until all the incorrect models have been updated and we get individual decision trees from every model. After all this, the model can detect whether the patient has

recurrent or non-recurrent Cancer. Adaboost classifier as shown in Algorithm 1 helps to make decision trees more efficient.

Algorithm 1. AC (Adaboost Classifier) describes as an ensemble learning which is used to classify the model. It will predict whether the patient is having cancer or not. Here it will use Decision Tree

- Step1: Adjust the weights of the samples.
- Step 2: Classify the data, build a decision tree, and assess the output.
- Step 3: Determine the tree's bearing on the overall categorization. Where significance is

$$Significance = \frac{1}{2} \log \left(\frac{1 - totalerror}{totalerror} \right)$$

where total error = Sum of all errors for sample weights,
significance = performance of the stump

- Step 4: the sample weights so that the subsequent decision tree may consider the mistakes made by the decision tree that came before it.

$$NewWeight = OldWeight * e^{Significance}$$

- Step 5: Form a new dataset

$$D_{t+i}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$$

where h_t = hypothesis/classifier, ϵ = minimum misclassification error for the model,
 α =weight for the classifier, \exp = Euler's e: 2.71828
 Z_t =normalization factor used to ensure that weights represent a true distribution
 X_i =current feature value, Y_i = current output value

- Step 6: Repeat steps 2-5 until the number of iterations matches the hyperparameter number of estimators
- Step 7: Make predictions based on data not part of the training set using the forest of decision trees

3.5.2. Multi-layer perceptron

In MLP, Algorithm 2 describes a fully connected feed forward neural network. It generates a set of outputs from set of inputs. The model will take each row as an input with all the 47 features in the training dataset, and it calculates the input values' dot product with the weights between the input layer and the hidden layer. Now the model doesn't push forward; instead, they utilize the activation function at each calculated layer. The activation function that was used in the proposed model is tanh.

$$a_i = \tanh(\text{net}_i) = \frac{e_i^{\text{net}} - e^{-\text{net}_i}}{e_i^{\text{net}} + e^{-\text{net}_i}}$$

where a_i = activation, n_i = network input

The calculated values are pushed through the activation functions then the deals are made into the next layer by taking the weights corresponding to the next hidden layer. This process takes place until the model reaches the output layer. In training, when the model comes to the output layer, it correlates to the output with the value it acquired when it reaches the output layer. It maps the calculated value of the Cancer as Recurrent or Non-Recurrent. In the testing phase, the model does the entire process and acquires the value, and it must predict if the Cancer is recurrent or not and checks with the actual data and calculates its accuracy. The summary of the entire process can be represented in the form of:

$$y = \varphi(\sum_{i=1}^n w_i x_i + b) = \varphi(w^T x + b)$$

where: w =vector of weights; x =vector of inputs; b =bias; φ =non-linear activation function

3.5.3. Histogram-based gradient boosting (HGB) Classifier

The model HGB Classifier takes all 47 features of data as input, and it starts working as a gradient boost machine (GBM). Here GBM is created by iteratively building 47 weak learners, where each soft learner weights greater the mispredictions of the last learner and so on. It implements vulnerable base learners as decision trees, one reason for this algorithm's efficiency. It is as follows, Sum of multiple weak learners F_m built on a stage-wise fashion, where each F is a decision tree. If the model is trained with m trees, then the final gradient boost Machine will be constructed as:

$$F_M(x) = F_0(x) + \sum_{m=1}^M F_m(x)$$

where, $F_M(x)$ = Mth weak learner, $F_0(x)$ = first weak learner. Here, $M=47$.

The model works with decision trees,

- The first tree learns how to fit the dataset's target variable, 'Class'. It calculates the average of target label values.

$$(0 + 1)/2 = 0.5$$

- The second tree learns how to fit the residuals between the predictions of the first tree and the ground truth.

$$\text{Residual value} = \text{Actual value} - \text{Predicted value}$$

- The third tree learns how to fit the residuals of the second tree, and so on.

A DT is constructed, and it is built to predict the residuals. Now the histogram comes into the picture. The algorithm creates a histogram of feature values with equal bins. Here there are 2 values, so it will have 2 bins 0 and 1 and are mapped against the bin index. Each value maps to the index in the histogram. During training, the tree learns whether samples with missing data should be sent to the left or right child at each split point. Since we have no missing values left in the data set, it sets the values for a given feature. Whichever child has the most value here is 1, i.e., recurrent. It predicts the target by multiplying each residual with the learning rate. Here, the default learning rate is 0.1.

$$\text{Predicted value} = \text{residual} \times 0.1 (\text{learning rate})$$

The entire process takes place until all the iterations are completed. Here all the weak learners are built as strong learners and thus gradually improve the model's performance. Though it starts slowly, it gradually boosts up and performs well to predict the RBC by classification. As an ensemble model, it generates multiple decision trees on the dataset by selecting random features. The bins initialized with 0 weights are skipped making the algorithm work faster against the dataset. Algorithm 3, describes about HGB classification tree and it is a technique for training faster decision trees used in the gradient boosting ensemble model. The output of a decision tree is given as input to another tree so that the error can be reduced, and the model can predict the RBC efficiently.

Algorithm 2. Multi layer perceptron (MLP) is a fully connected Feed Forward Neural Network. It generates a set of outputs from set of inputs

- Step 1: Forward Pass
In this model training stage, we send the input to the model, multiply it with weights, and apply bias at every layer.
- Step 2: Calculate error or loss
Now, after we forward pass, we get an output from the model, and now we must compare it with the actual output or expected output. And based upon the actual output and expected output, we must find the loss and to calculate it; we must backpropagate. There are many formulas to calculate the error or loss. The choice should match our requirements.
- Step 3: Backward Pass
After the loss is computed, it is back-propagated and used to inform a gradient-based update to the model's weights. Here we focus on the essential part of training the model. As the gradient moves in that direction, the consequences will shift accordingly.

3.5.4. Proposed model (voting classifier)

As VC is an ensemble technique, it enhances RBC detection classification. Three Ensemble Classification Models have been given to the VC. They are Adaboost, XGboost and Histogram Gradient Boosting Classifier Models. Here we have applied both hard and soft voting. A "hard voting" method chooses an outcome most consistent with the classifications predicted by the various classifiers. After assigning each classifier a weight, the class label is determined by averaging the classifiers' predictions and returning the class label as the maximum of those predictions. After that, the class label that has the most

significant average probability is the one that is used to produce the final class label. By voting lightly, we got the highest possible VC score.

The voting ensemble classifier has two hyperparameters, namely estimators and voting. The estimator hyperparameter will create a list of objects for all the ensemble models used for the vote. The voting hyperparameter can be either soft or hard. So, we created objects for the three ensemble models: Adaboost, XGboost and Histogram Gradient Boosting Classifiers. These three objects are passed as parameters for the VC along with the voting hyperparameter, i.e. either soft or hard. Later, an object is created for VC as well. When we set the voting hyperparameter as hard, all three classifiers passed as estimator hyperparameters will predict the target class. Out of all predicted target classes for a given record, the VC will select the most predicted target classes.

Predicted Class=MODE (Adaboost predicted class, XGboost predicted type, HGboost predicted class)

When we set the voting hyperparameter as soft, all three classifiers passed as the estimator's hyperparameters will predict the probability of the target classes. The probabilities predicted by all three classifiers are averaged for every target class of a record. Out of all the predicted target classes, the target class with maximum likelihood will be selected as a predicted class. The Entire work process is shown in Figure 1. Here algorithm 4 describes about VC machine learning model which trains on ensemble of numerous models and predicts the output on their highest value.

$$P_1(\text{Votingclassifierclass1}) = \frac{P_1(\text{XGboost})+P_1(\text{Adaboost})+P_1(\text{HGboost})}{3}$$

$$P_2(\text{Votingclassifierclass2}) = \frac{P_2(\text{XGboost})+P_2(\text{Adaboost})+P_2(\text{HGboost})}{3}$$

$$\text{Predictedclass} = \text{Max} (P_1 , P_2)$$

Where P_1 = Average probability of class 1 to be predicted,

- P_2 = Average probability of class 1 to be predicted,
- $P_1(\text{XGboost})$ = probability of class 1 to be predicted by XGboost classifier,
- $P_1(\text{Adaboost})$ = probability of class 1 to be predicted by Adaboost classifier,
- $P_1(\text{HGboost})$ = probability of class 1 to be predicted by HGboost classifier,
- $P_1(\text{XGboost})$ = probability of class 1 to be predicted by XGboost classifier,
- $P_1(\text{Adaboost})$ = probability of class 1 to be predicted by Adaboost classifier,
- $P_1(\text{HGboost})$ = probability of class 1 to be predicted by HGboost classifier.

Algorithm 3. HGB is a classification tree and it is a technique for training faster decision trees used in the gradient boosting ensemble model

Step 1: Calculate the average of target label values. The values in the target variable are 0 and 1.

So the average will be 0.5.

Step 2: The model generates decision trees for the target label. Calculate the residual of each and every tree. So that the error will be calculated and passed through the next tree in the next iteration.

Residual value = Actual value - Predicted value

Step 3: Constructs a decision tree. A new decision tree is built in every iteration, i.e. the number of iterations here is 50, so 50 decision trees are created.

Step 4: The model predicts the target value using residual values and learning rate. Here the learning rate is 0.1.

Predicted value = Residual x Learning rate.

Step 5: Plots Histogram with equal bins of feature values which is the goal state. It plots the histogram of predicted values, i.e. 0 and 1, against the values count.

Step 6: Calculates the gain function of a node. The left and right child gradients are enough to calculate the gain function.

$$\sum g_i = \sum_{k \in \text{left}} g_k + \sum_{j \in \text{right}} g_j$$

where, g_i = total gain, g_k = left child gradient, g_j = Right child gradient.

Step 7: Compute the new residuals

Step 8: Repeat steps 2 to 7 until the specified number of maximum iterations is reached.

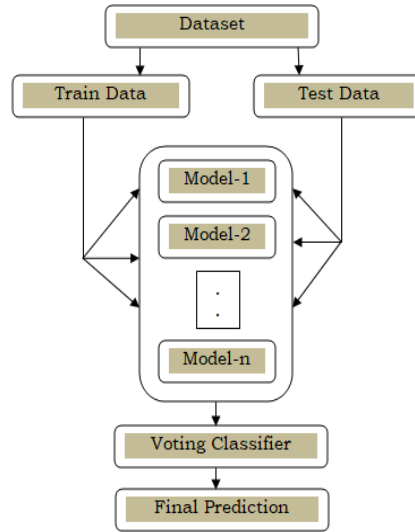


Figure 1. Workflow of voting classifier

Algorithm 4. VC is a machine learning model which trains on ensemble of numerous models and predicts the output on their highest value

Step 1: Create the objects for all the classifiers, i.e. for Adaboost, XGboost and Histogram Gradient Boosting Classifier.

Step 2: Create the object for the VC by taking the estimators hyperparameter as a list of things for classifiers Adaboost, XGboost and Histogram Gradient Boosting Classifier and set the voting hyperparameter as soft or hard.

Step 3: Train the VC with the training data; this implies training all three classifiers: Adaboost, XGboost and Histogram Gradient Boosting Classifier.

Step 4: The VC predicts the target class based on the voting hyperparameter

4. RESULTS

Metrics help analyze and measure ML models' performance quality in terms of efficiency and error proneness by using Accuracy, Precision, Recall, F1 score and Specificity values. Every metric of Table 2 is evaluated by using the Confusion Matrix. The entire proposed work is implemented using Python.

Table 2. Metrics evaluation

| S.NO | Metric | Expression |
|------|-------------------|---|
| 1 | Accuracy | $\frac{TP + TN}{TP + FP + TN + FN}$ |
| 2 | Specificity | $\frac{TN}{TN + FP}$ |
| 3 | Precision | $\frac{TP}{TP + FP}$ |
| 4 | Recall | $\frac{TP}{TP + FN}$ |
| 5 | F1 – Score | $2 \times \frac{Precision \times Recall}{Precision + Recall}$ |
| 6 | Cohen Kappa Score | $\frac{P_o - P_e}{1 - P_e}$ |

From Table 2, metrics performance was evaluated and compared with metrics using the models Adaboost, MLP, HGBM and voting classifier. These values are shown in Table 3. By using this Table 3, various graphs were drawn. Among them, Figure 2 was shown for Accuracy metrics with a different model; Figure 3 was shown for comparison on numerous metrics like Accuracy, Precision, Recall, F1 Score and Specificity. Figure 4 was shown the Cohen_Kappa metric compared with various models.

Table 3. Metrics comparison obtained by all the models

| Model/Metric | Accuracy | Precision | Recall | F1 Score | Specificity | Cohen_Kappa |
|-------------------|---------------|---------------|--------------|--------------|---------------|---------------|
| Adaboost | 0.956 | 0.9767 | 0.913 | 0.943 | 0.913 | 0.9079 |
| MLP | 0.96 | 0.96 | 0.923 | 0.957 | 0.923 | 0.9201 |
| HGBM | 0.972 | 0.9687 | 0.947 | 0.959 | 0.941 | 0.9635 |
| Voting Classifier | 0.9824 | 0.9782 | 0.962 | 0.969 | 0.9782 | 0.9635 |

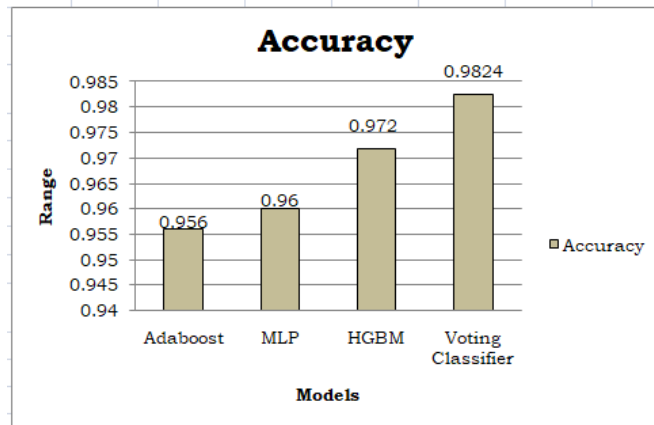


Figure 2. Accuracy for various models obtained

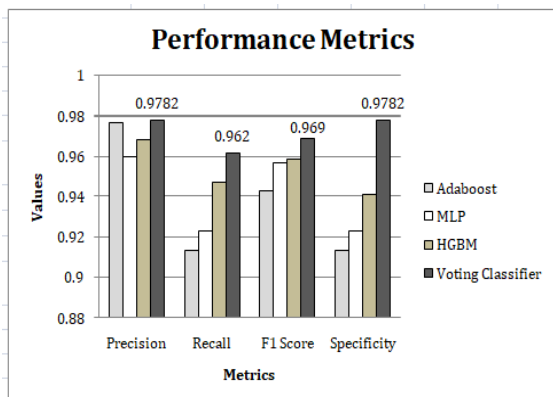


Figure 3. Comparison with various metrics with various models

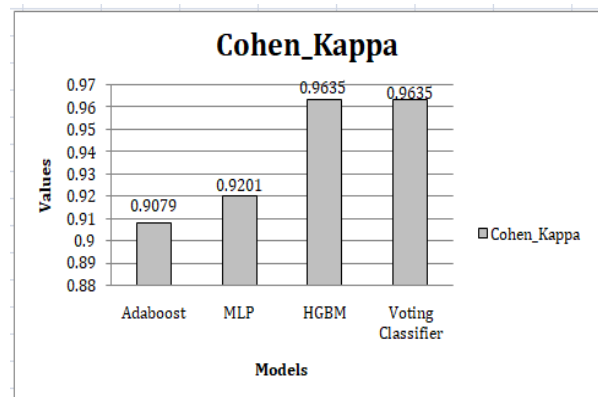


Figure 4. Comparison with Cohen_Kappametric with various models

5. CONCLUSION




The RBC is detected concerning the tumour data and the diseases a patient is already suffering from. The RBC is seen with the help of Adaboost, XGboost and Histogram Gradient Boosting Classifier, and the Voting Ensemble Classifier is used for better efficiency. The model performed well with 98.24% accuracy. Also, other metrics scored by the model are Precision of 0.9782, Recall of 0.962, F1-Score of 0.969, Specificity of 0.9782 and Cohen Kappa Score of 0.9635. So, when patient data is given as input, it provides the best output. In the future, we can extend this project further to predict other BC-related diseases so that it can help predict diseases in the medical field. Future Research should consider the potential effects of RBC on Heredity. Future Research on gene mutations might extend the explanations of RBC. This is desirable for future work.

REFERENCES




[1] F. Selchick, "A comprehensive guide to breast cancer," *Healthline*, 2022. <https://www.healthline.com/health/breast-cancer> (accessed Mar. 15, 2022).
 [2] MayoClinic, "Recurrent breast cancer," *MayoClinic*, 2021. <https://www.mayoclinic.org/diseases-conditions/recurrent-breast-cancer/symptoms-causes/syc-20377135> (accessed Mar. 12, 2022).

- [3] F. Selchick, "What happens at each stage of breast cancer?," *Medical News Today*, 2021. <https://www.medicalnewstoday.com/articles/322760#how-is-the-stage-determined> (accessed Mar. 25, 2022).
- [4] O. L. M. William H. Wolberg, and W. Nick Street, "Breast cancer Wisconsin (Diagnostic) data set," *UCI Machine Learning Repository*, 1995. <https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29>.
- [5] "sklearn.ensemble.VotingClassifier," Scikit learn, 2007. <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.VotingClassifier.html>(accessed Sep. 25, 2022)
- [6] W. Kim *et al.*, "Development of novel breast cancer recurrence prediction model using support vector machine," *Journal of Breast Cancer*, vol. 15, no. 2, pp. 230–238, 2012, doi: 10.4048/jbc.2012.15.2.230.
- [7] L. Ahmad and A. Eshlaghy, "Using Three Machine Learning Techniques for Predicting Breast Cancer Recurrence," *Journal of Health & Medical Informatics*, vol. 04, no. 02, p. 2018, 2013, doi: 10.4172/2157-7420.1000124.
- [8] M. H. Galea, R. W. Blamey, C. E. Elston, and I. O. Ellis, "The Nottingham prognostic index in primary breast cancer," *Breast Cancer Research and Treatment*, vol. 22, no. 3, pp. 207–219, 1992, doi: 10.1007/BF01840834.
- [9] S. Laghmati, A. Tmiri, and B. Cherradi, "Machine learning based system for prediction of breast cancer severity," in *Proceedings - 2019 International Conference on Wireless Networks and Mobile Communications, WINCOM 2019*, 2019, doi: 10.1109/WINCOM47513.2019.8942575.
- [10] M. Lundin, J. Lundin, H. B. Burke, S. Toikkanen, L. Pylkkänen, and H. Joensuu, "Artificial neural networks applied to survival prediction in breast cancer," *Oncology*, vol. 57, no. 4, pp. 281–286, 1999.
- [11] A. K. Mishra, P. Roy, and S. Bandyopadhyay, "Binary particle swarm optimization based feature selection (BPSO-FS) for improving breast cancer prediction," *Advances in Intelligent Systems and Computing*, vol. 1164, pp. 373–384, 2021, doi: 10.1007/978-981-15-4992-2_35.
- [12] V. Lahouraet *al.*, "Cloud computing-based framework for breast cancer diagnosis using extreme learning machine," *Diagnostics*, vol. 11, no. 2, 2021, doi: 10.3390/diagnostics11020241.
- [13] N. Rathore, Divya, and S. Agarwal, "Predicting the survivability of breast cancer patients using ensemble approach," in *Proceedings of the 2014 International Conference on Issues and Challenges in Intelligent Computing Techniques, ICICT 2014*, Feb. 2014, pp. 459–464, doi: 10.1109/ICICT.2014.6781326.
- [14] K. Goyal, P. Aggarwal, and M. Kumar, "Prediction of breast cancer recurrence: a machine learning approach," in *Advances in Intelligent Systems and Computing*, vol. 990, 2020, pp. 101–113, doi: 10.1007/978-981-13-8676-3_10.
- [15] C. Shah and A. G. Jivani, "Comparison of data mining classification algorithms for breast cancer prediction," in *2013 4th International Conference on Computing, Communications and Networking Technologies, ICCCNT 2013*, Jul. 2013, pp. 1–4, doi: 10.1109/ICCCNT.2013.6726477.
- [16] U. Ojha and S. Goel, "A study on prediction of breast cancer recurrence using data mining techniques," in *Proceedings of the 7th International Conference Confluence 2017 on Cloud Computing, Data Science and Engineering*, Jan. 2017, pp. 527–530, doi: 10.1109/CONFLUENCE.2017.7943207.
- [17] C. Boeri *et al.*, "Machine Learning techniques in breast cancer prognosis prediction: A primary evaluation," *Cancer Medicine*, vol. 9, no. 9, pp. 3234–3243, 2020, doi: 10.1002/cam4.2811.
- [18] V. Chaurasia and V. Chaurasia, "Data mining techniques: to predict and resolve breast cancer survivability," *International Journal of Computer Science and Mobile Computing (IJCSMC)*, vol. 3, no. 1, pp. 10–22, 2014.
- [19] K. Sivakami, "Mining big data: breast cancer prediction using DT - SVM Hybrid model," *International Journal of Scientific Engineering and Applied Science*, vol. 1, no. 5, pp. 418–429, 2015.
- [20] N. Arya and S. Saha, "Multi-modal advanced deep learning architectures for breast cancer survival prediction," *Knowledge-Based Systems*, vol. 221, 2021, doi: 10.1016/j.knosys.2021.106965.
- [21] K. Karunasri, G. Mahesh, and R. S. Shankar, "Medical images security in cloud computing using Cp-Abe algorithm," *ARPN Journal of Engineering and Applied Sciences*, vol. 17, no. 7, pp. 759–766, 2022.
- [22] S. S. Reddy, R. Rajender, and N. Sethi, "A data mining scheme for detection and classification of diabetes mellitus using voting expert strategy," *International Journal of Knowledge-Based and Intelligent Engineering Systems*, vol. 23, no. 2, pp. 103–108, 2019, doi: 10.3233/KES-190403.
- [23] N. Sethi, Shiva, S. Reddy, and R. Rajender, "A comprehensive analysis of machine learning techniques for incessant prediction of diabetes mellitus," *International Journal of Grid and Distributed Computing*, vol. 13, no. 1, pp. 1–22, 2020.
- [24] S. S. Reddy, N. Sethi, and R. Rajender, "A review of data mining schemes for prediction of diabetes mellitus and correlated ailments," *Proceedings - 2019 5th International Conference on Computing, Communication Control and Automation, ICCUBEA 2019*, 2019, pp. 1-5, doi: 10.1109/ICCUBEA47591.2019.9128880.
- [25] S. S. Reddy, N. Sethi, and R. Rajender, "Evaluation of Deep Belief Network to Predict Hospital Readmission of Diabetic Patients," in *Proceedings of the 2nd International Conference on Inventive Research in Computing Applications, ICIRCA 2020*, Jul. 2020, pp. 5–9, doi: 10.1109/ICIRCA48905.2020.9182800.
- [26] S. S. Reddy and G. Mahesh, "Risk assessment of type 2 diabetes mellitus prediction using an improved combination of NELM-PSO," *EAI Endorsed Transactions on Scalable Information Systems*, vol. 8, no. 32, pp. 1–26, 2021, doi: 10.4108/eai.3-5-2021.169579.
- [27] V. N. Gopal, F. Al-Turjman, R. Kumar, L. Anand, and M. Rajesh, "Feature selection and classification in breast cancer prediction using IoT and machine learning," *Measurement: Journal of the International Measurement Confederation*, vol. 178, p. 109442, 2021, doi: 10.1016/j.measurement.2021.109442.
- [28] M. Islam, M. Haque, H. Iqbal, H. Hasan, M. Hasan M, and M.N. Kabir, "Breast cancer prediction: a comparative study using machine learning techniques." *SN Computer Science*. vol. 1, no. 5, pp. 1-4, Sep. 2020.
- [29] V. Kumar, B. K. Mishra, M. Mazzara, D. N. H. Thanh, and A. Verma, "Prediction of malignant and benign breast cancer: a data mining approach in healthcare applications," *Lecture Notes on Data Engineering and Communications Technologies*, vol. 37, pp. 435–442, 2020, doi: 10.1007/978-981-15-0978-0_43.
- [30] J. Koh *et al.*, "Three-dimensional radiomics of triple-negative breast cancer: Prediction of systemic recurrence," *Scientific Reports*, vol. 10, no. 1, 2020, doi: 10.1038/s41598-020-59923-2.
- [31] M. R. Mohebian, H. R. Marateb, M. Mansourian, M. A. Mañanas, and F. Mokarian, "A hybrid computer-aided-diagnosis system for prediction of breast cancer recurrence (HPBCR) using optimized ensemble learning," *Computational and Structural Biotechnology Journal*, vol. 15, pp. 75–85, 2017, doi: 10.1016/j.csbj.2016.11.004.
- [32] S. S. Reddy, N. Pilli, P. Voosala, and S. R. Chigurupati, "A comparative study to predict breast cancer using machine learning techniques," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 27, no. 1, pp. 171–180, 2022.




BIOGRAPHIES OF AUTHORS

Shiva Shankar Reddy    is an Assistant Professor at the Department of Computer Science and Engineering in Sagi RamaKrishnam Raju Engineering College, Bhimavaram, Andhrapradesh, INDIA. He is pursuing PhD degree in Computer Science and Engineering with a specialization in Medical Mining and Machine Learning. His research areas are image processing, medical mining, machine learning, deep learning and pattern recognition. He published 30+ papers in International Journals and Conferences. S.S. Reddy has filed 05 patents. His research interests include image processing, medical mining, machine learning, deep learning and pattern recognition. He can be contacted at email: shiva.csesrkr@gmail.com.






Ravi Swaroop Chigurupati    is Assistant Professor at Sagi Ramakrishnam Raju Engineering College, Department of Computer Science and Engineering, India. He received B.Tech. degree in Sagi Ramakrishnam Raju Engineering College, Department of Information Technology in 2012. He holds an M.Tech. degree in Sagi Ramakrishnam Raju Engineering College, Department of Information Technology, in 2018. His research areas are image processing, bioinformatics, machine learning, deep learning, and data mining. He can be contacted at email: raviswaroop.chigurupati@gmail.com.



Priyadarshini Voosala    is Assistant Professor at Sagi Ramakrishnam Raju Engineering College, Department of Computer Science and Engineering, India. She Received a B.Tech. degree from Sri Vishnu Engineering College for Women, Department of Computer Science and Engineering, in 2005. She holds an M.Tech. degree in Sagi Ramakrishnam Raju Engineering College, Department of Computer Science and Engineering, in 2010. Her research areas are cloud computing, fog computing, edge computing, machine learning and image processing. She can be contacted at email: priyavoosala@gmail.com.



Neelima Pili    is Assistant Professor at Sagi Ramakrishnam Raju Engineering College, Department of Computer Science and Engineering, India. She Received a B.Tech. degree in Sagi Ramakrishnam Raju Engineering College, Department of Computer Science and Engineering, in 2006. She holds an M.Tech. degree in Sagi Ramakrishnam Raju Engineering College, Department of Computer Science and Engineering, in 2010. Her research areas are cloud computing, fog computing, edge computing, machine learning and IoT. She can be contacted at email: neelima.p47@gmail.com.