

Article - Engineering, Technology and Techniques

Design of Automatic Tool for Diagnosis of Pneumonia Using Boosting Techniques

Seda Postalcioglu^{1*}

<https://orcid.org/0000-0002-3188-8116>

¹Izmir Democracy University, Faculty of Engineering, Department of Computer Engineering, Izmir, Turkey.

Editor-in-Chief: Alexandre Rasi Aoki

Associate Editor: Raja Soosaimarian Peter Raj

Received: 17-May-2021; Accepted: 27-Aug-2021.

*Correspondence: seda.postalcioglu@idu.edu.tr; Tel.: +90232 260 1001 (S.P.).

HIGHLIGHTS

- Automatic tool is developed for diagnosis of pneumonia. Boosting techniques are used in terms of their speed and ease of use in real time applications.
- The best result in terms of simulation duration and accuracy is Catboost with 0.7 seconds running time and 83% accuracy.
- The results obtained from the model become more understandable using tool.
- A bridge is designed between the model and the user by the automatic tool.
- By using this tool, a diagnosis can be done quickly and accurately without any expert. So, treatment can be started quickly.

Abstract: Covid-19 is today's pandemic disease and can cause the hospital crowded. Additionally, It affects the lungs and may cause pneumonia. The most popular technique for diagnosis of pneumonia is the evaluation of X-ray. However, a sufficient number of radiologists are needed to interpret the X-ray images. High rates of child deaths due to pneumonia have been encountered. Using this type of system, a diagnosis can be made quickly, and then the treatment process can be started rapidly. This study aims to diagnose pneumonia using boosting techniques by the automatic tool. With this tool, the workload of the doctors/radiologists is reduced. The boosting techniques are a family of machine learning techniques. Gradient Boosting Machine (GBM), Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine (LightGBM), and Categorical Boosting (CatBoost) are used for the study. These techniques are chosen because of their simulation duration for modeling and convenience for real-time applications. L2 normalization and feature selection are applied to the data before applying the techniques. Random Forest Classifier is used for feature selection estimator. After the modeling, Categorical Boosting algorithm is observed as faster than the other techniques. Simulation duration is obtained as 0.7 seconds. By using this automatic tool, the user can be able to upload the desired X-ray image to the system and get the result easily from the screen without any radiologist/doctor.

Keywords: Categorical boosting; extreme gradient boosting; gradient boosting; light gradient boosting; machine learning; pneumonia; user interface tool.

INTRODUCTION

The respiratory system of human consist of various organ series that is responsible for inhaling oxygen and exhale carbon dioxide [1]. The primary organ for the respiratory system is the lungs by which mechanism of exchanges of gases takes place. According to the American Lung Association, the oxygen from the lungs is collected by red blood cells and delivered to the needed part of the body and in exhaling process is collected by red blood cells and delivered back to the lungs that leave out from the body as we exhale. As we breathe, the air goes through the nose or mouth to the alveoli of the lungs via air passageways which are the bronchioles or bronchiole [1]. The main division of the lungs is right and left lungs. The supply of blood to the lungs is performed by the pulmonary circulation. Improper functioning of the lungs is due to various abnormalities that lead to lung disease [1]. Pneumonia is a pathogenic infection of the lung parenchyma, which is most commonly caused by bacteria or viruses, and less commonly by other microorganisms such as fungi [2]. This disease is very serious in young children and elderly patients with a weakened immune system. Pneumonia killed 920,000 young children worldwide in 2015 [3]. According to the report of the World Health Organization, pneumonia is one of the major diseases of death [1]. Infants, children, elderly aged people with poor immune systems are mostly affected by pneumonia [4]. A report on child deaths due to infectious diseases, given in reference 5, has been prepared. In this report, child deaths are discussed and pneumonia is emphasized. This report shows how international cooperation can save 5.3 million lives by 2030 [5].

Approximately, 1.4 million children under the age of five years are killed by it which accounts for nearly 18 of all death of children under five years old worldwide [1]. Every year on 12th November World Pneumonia Day is celebrated for spreading awareness about pneumonia. The theme for World Pneumonia Day 2018 is Stop Pneumonia: Invest in Child Health [1].

Imaging plays an important role in detecting and diagnosing pneumonia [1]. CT scan, chest X-rays, and ultrasound are different imaging techniques for lung pneumonia. The most available imaging is X-Ray which is the more accurate, painless, and noninvasive type. X-ray imaging is the most preferred imaging method [1,2]. Since pneumonia is a contagious disease i.e. cough or sneeze which is exhaled by infected subjects goes in the air and as anyone inhales that contaminated air, get infected by pneumonia [1].

However, analyzing a chest X-ray image may be tedious, time-consuming, and requiring expert knowledge that might not be available in less-developed regions [2]. A careful examination of the chest X-ray is required to detect pneumonia. This requires experienced and knowledgeable radiologist professionals. This makes the pneumonia detection process a challenging task [4]. Therefore, computer-aided diagnosis systems are needed [2]. Computer-aided diagnosis has been developing rapidly in the last decade [3]. The purpose of computer-aided diagnosis is to assist radiologists in interpreting medical images using computer results. This type of diagnosis helps to improve diagnostic accuracy and reduce the workload of experts [3].

Recently, many classification systems based on deep learning have been proposed [2]. Jain et al and Yu et al used convolutional neural networks to detect pneumonia for their study [6,7]. Li et al studied deep learning for automated detection of pneumonia using X-ray images [8]. Ge et al studied predicting post-stroke pneumonia using a deep neural network [9]. Chassagnon et al examined AI-driven quantification, staging, and outcome prediction of COVID-19 pneumonia [10]. Wang and coauthors are studied COVID-19 classification using deep fusion using transfer learning [11] and in reference 12 diagnosis of Covid 19 is done using Wavelet Renyi Entropy and Three-Segment Biogeography [12]. Postalcioglu and Kesli used Naive Bayes method for pneumonia diagnosis [13]. Ramezanpour et al mentioned a collaboration between clinicians and machines at the decision stage [14]. Khan et al studied a classification method using deep learning for brain tumor type [15]. Galván-Tejada et al examined a multivariate model to classify benign and malignant tumor lesions using a computer-assisted diagnosis [16].

In this study, a normal and infected data set of chest X-ray is used for the diagnosis of pneumonia. Boosting techniques which are the machine learning techniques are used. An automatic decision making tool is designed.

The aim of this study is to diagnose of Pneumonia from X-ray images using automatic tool. Diagnosis can be done quickly and accurately by anyone using this automatic tool. Thus, it is a study that aims to reduce the hospital densities encountered especially during the covid 19 period by using this designed tool. Additionally, high rates of child deaths due to pneumonia have been encountered. Using this type of system, a diagnosis can be done quickly in case of absence of a specialist doctor. With the designed system, diagnosis can be done accurately and quickly without any radiologist. So the treatment can be started quickly.

MATERIAL AND METHODS

Supervised machine learning classifiers can be divided into multiple types. The supervised machine learning algorithms are shown in Figure 1 [17]. Ensemble method is a machine learning technique where weak learners are trained to solve the problem and combined to get better results [17]. The main hypothesis is that when weak models are correctly combined, more accurate models can be obtained [18]. Ensemble methods can be classified as bagging and boosting [19]. Bagging is a simple aggregation technique. It is combined using some model averaging techniques [19]. Boosting is considers homogeneous weak learners, learns them sequentially and combines them [18]. Doing this obtains to eliminate bias, improve model accuracy. The reasons for the success of boosting algorithms are; their ability to incorporate automated variable selection and model choice in the fitting process, their flexibility and their stability in the case of high-dimensional data [20]. The application of boosting algorithms thus offers an attractive option for biomedical researchers [20].

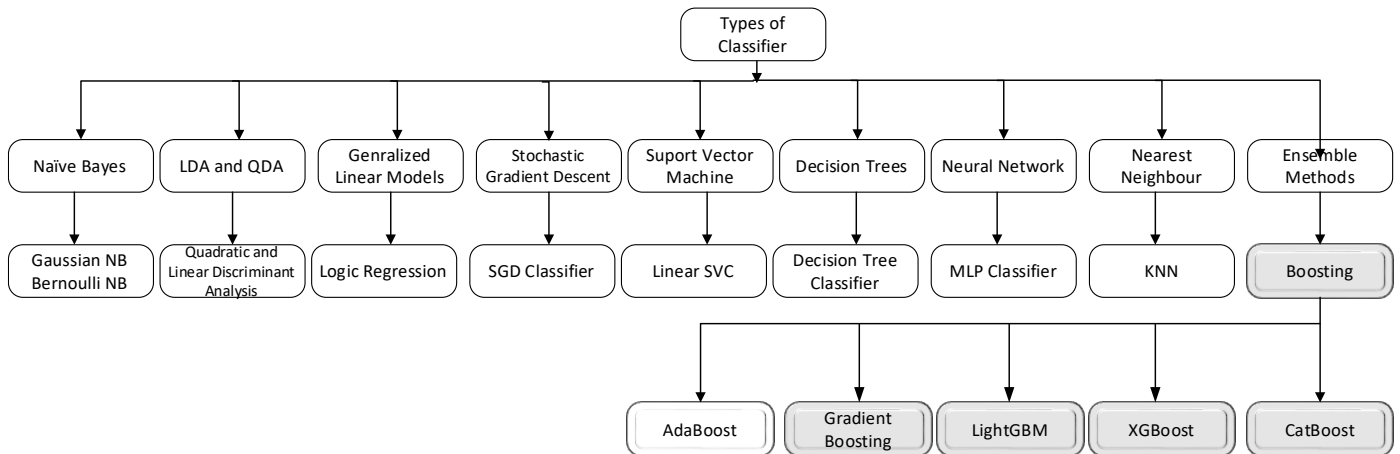


Figure 1. Types of the supervised machine learning algorithm [17]

Figure 2 shows the first stage of the diagnosis of pneumonia from X-ray images. In this study, CatBoost, LightGBM, XGBoost, and GBM are used as boosting techniques. The details are given below.

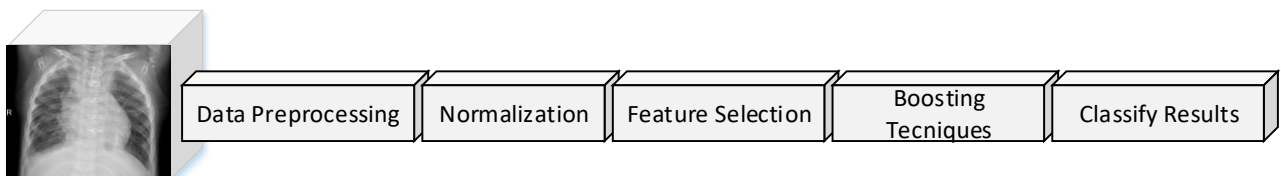


Figure 2. Stage of the diagnosis of pneumonia from X-ray images

The proposed approach consists of two steps. The first step is to create a model for the diagnosis of pneumonia, and the second step is to use the created model easily with the user interface tool. The first step is to diagnose the disease using Boosting techniques. Training is realized. The rationality and adaptability of the modeled structures are tested with the test data. Performance results are given to the figure at 10-11-12.

The second part of the study aims to enable the created model to be used by anyone who does not have any programming knowledge. The automatic tool has been designed. It is like a bridge between the user and the program. This bridge is designed with the Tkinter GUI framework. Thus, it can be able to reach the result of the diagnosis by loading the x-ray to the tool. Figure 3 shows the general system for the study.

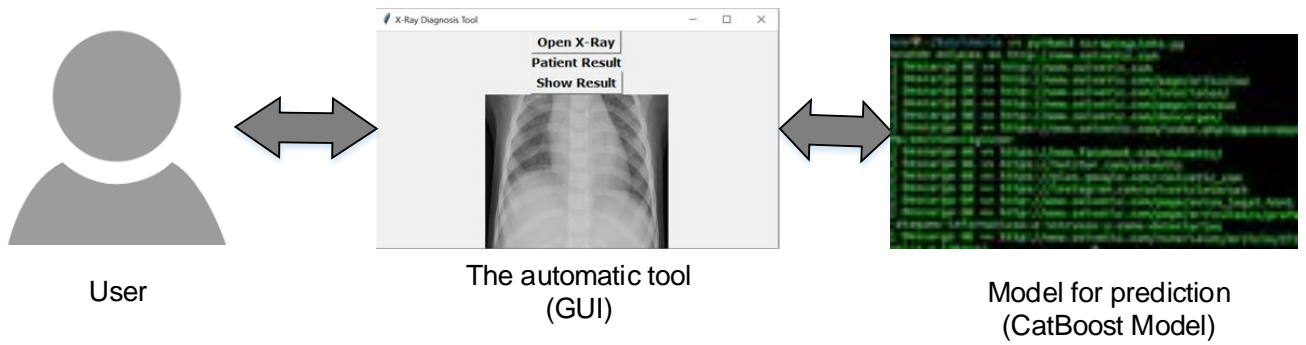


Figure 3. General System

Dataset

The dataset is obtained from the Kaggle website [21]. Samples from chest X-ray images are given in Figure 4. An unbalanced distribution is observed when the data distribution is examined. The data are distributed properly. Figure 5 shows a uniformly distributed data set. The data set is separated to 80% for training and 20% for testing with the number of 2400 training and 600 test data respectively for modeling. Firstly, X-ray images are resized to (2400, 150528) as data preprocessing. L2 norm is used. Equation (1) shows the L2-norm [22].

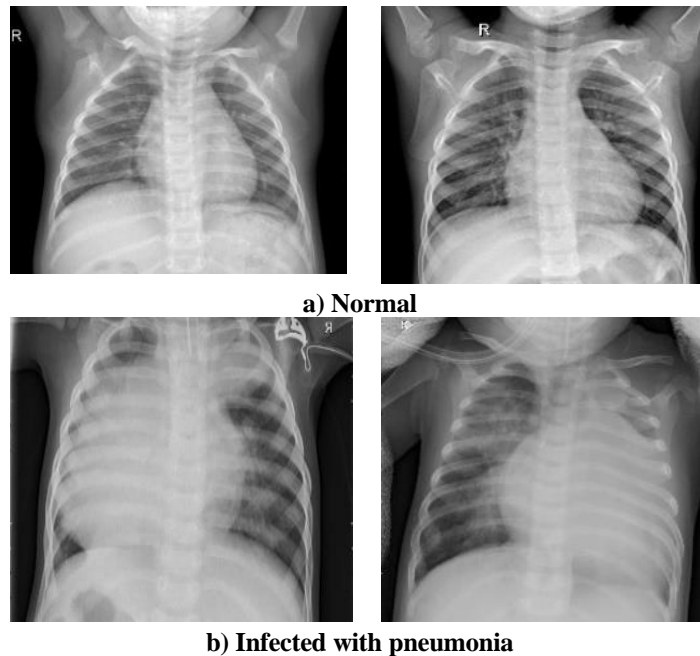


Figure 4. Samples from chest X-ray a) Normal, b) Infected with pneumonia

$$\|x\|_2 = \sqrt{\sum_i x_i^2} \tag{1}$$

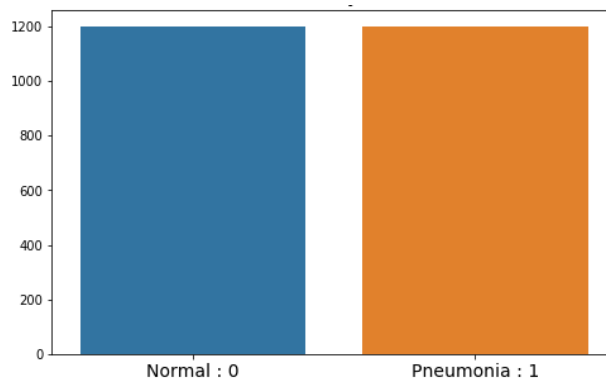


Figure 5. Dataset for training

Feature Selection

The feature selection method computes the relative importance of each attribute. These important values are used to inform a feature selection process. Random Forest Classifier estimator is used to select the best features. Random forest (RF) algorithm is a well-known tree-based ensemble learning method and the bagging-type ensemble [23]. RF differs from other standard trees, each node is split using the best among a subset of predictors randomly chosen at that node [23]. Classification of individuals is based upon aggregate voting over all trees in the forest. The out-of-bag (OOB) individuals are used to estimate the importance of particular attributes. If randomly permuting values of a particular attribute does not affect the predictive ability of trees on out-of-bag samples, that attribute is assigned a low importance score. If randomly permuting the values of a particular attribute drastically impairs the ability of trees to correctly predict the class of out-of-bag samples, then the importance score of that attribute will be high [24]. The features with the highest absolute value are considered the most important. Feature selection allows for setting the threshold. Only the features with the higher than the threshold remains. In this paper, the correlation score of 0.001 is used as a threshold, and features above this threshold are eliminated. Figure 7 and Figure 8 show the results.

Gradient Boosting Machine

The family of boosting techniques is a machine learning technique for classification problems [19]. Gradient boosting machine sequentially adds new models from a group of weak models. At each iteration, a new weak, base-learner model is trained concerning the error of the whole ensemble learned [25]. The idea that each new model can minimize the loss function [17]. The overall accuracy is improved by using the loss function. However, boosting must eventually be stopped; otherwise, the model may tend to be overfit. The stopping criteria can be a threshold or the maximum number of models created [17]. Estimation of the regression function $f(\cdot)$ of a model, the predictor variables X with the outcome Y , can be seen in equation 2 [20].

$$\hat{f}(\cdot) = \operatorname{argmin} \{E_{Y,X}[\rho(Y, f(X))]\} \quad (2)$$

Where $\rho(\cdot)$ denotes a loss function. Loss function is used as logistic regression. Loss function is given in equation 3 [26].

$$\text{Loss} = \sum_{(x,y) \in D} -y \log(y') - (1 - y) \log(1 - y') \quad (3)$$

$(x, y) \in D$ is the dataset. y' is the predicted value given the set of features in x . Gradient boosting machines are a family of machine-learning techniques. They are so successful for practical applications [25]. The number of boosting stages to perform is used as a hundred. The fraction of samples to be used for fitting the individual base learners is determined as 1. MSE (mean squared error) is used for measuring the quality of a split. The learning rate is used as 0.1. Logistic regression is used for loss function. The maximum depth is used as 3. It limits the number of nodes in the tree. The minimum number of samples required to split is used as 2. The number of boosting stages to perform is used as 100.

Light Gradient Boosting Machine

LightGBM is a histogram-based algorithm. It reduces calculation cost by making variables with continuous value discrete. The training time of decision trees is directly proportional to the number of calculations and divisions made. This method reduces training time and resource use [27]. Depth-wise or leaf-wise can be used in learning in decision trees. In a level-oriented strategy, the balance of the tree is maintained while the tree grows [27]. In the leaf-oriented strategy, the division process from the leaves, which reduces the loss, continues. LightGBM differs from other boosting algorithms with this feature [27]. The model has less error rate and learns faster with a leaf-oriented strategy. However, the leaf-focused growth strategy causes the model to be prone to over-learning when the data is low. Therefore, the algorithm is more suitable for use in big data. Also, parameters such as depth, number of leaves can be optimized to prevent over learning [27].

The advantages of these algorithms are; Fast training speed, Low memory consumption. It can produce much more complex trees by following a leaf-wise split approach, which is the main reason for achieving higher accuracy [28]. Boosting type gradient boosting decision tree is used. The learning rate is used as 0.1 for LightGBM, Number of boosted trees is used as 100. The number of leaves per tree is used as 31. No limit is used for maximum tree depth. The minimum sum of instance weight needed in a leaf and is used as 0.001. The minimum number of data needed in a leaf is determined as 20.

Extreme Gradient Boosting

XGBoost stands for Extreme Gradient Boosting [29]. XGBoost is a machine learning technique based on decision-tree and gradient-boosting. Studies have shown that XGBoost model has low computational complexity, fast running speed, and high accuracy. The boosting algorithm is to integrate many weak classifiers into a strong classifier. The booster parameter sets the type of learners. Gbtree is used as a booster in this study. Gbtree booster uses a version of regression tree as a weak learner. As XGBoost is a lifting tree model, it integrates many tree models to create a powerful classifier [30]. There are two types of boosted trees in XGBoost. These are regression trees and classification trees. Given the n -labeled samples with features M , K additive functions are used to predict labels by the tree ensemble technique [31]. It is shown in equation (4) [31].

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i), f_k \in F \quad (4)$$

Where $F = f(x) = w_{q(x)}(q: R^m \rightarrow T, w \in R^T)$ is the regression tree's space. q represents the independent structure of each T-leaf tree. Each f_k corresponds to a tree that has independent leaf weight w . XGBoost minimizes the regularized objective to learn the set of functions. It is seen in Equation (5) [31].

$$L = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \quad (5)$$

$$\Omega(f_k) = \gamma T + \frac{1}{2} \lambda \|w\|^2$$

Where l is the loss function and Ω is the regularized term [31]. Where y_i is the prediction of the i -th instance. The XGBoost algorithm implements the weak learner by optimizing the structured loss function [32]. Gamma is the regularization hyperparameter, lambda is regularization term on weights. The values are 0 and 1 respectively. The number of boosting stages used as 100. Logistic regression is used for loss function. The maximum depth of a tree is used as 6. XGBoost make splits upto the max depth used and then start pruning the tree backwards and remove splits beyond which there is no positive gain.

Categorical Boosting

CatBoost is a decision tree based gradient boosting algorithm. One of the main differences between CatBoost and other boosting algorithms is that CatBoost generates trees symmetrically. In this way, it causes a serious reduction in training time. Additionally, it catches a high estimation rate without building very deep trees and overcomes the problem of over learning [33,34].

CatBoost is a gradient boosting application that uses binary decision trees. For example, Supposing a data with samples $D = \{(X_j, y_i)\}_{j=1, \dots, m}$, where $X_i = x_j^1, x_j^2, \dots, x_j^n$ is a vector of n features and response feature $y_j \in R$ is observed. The goal of the learning task is to train a function $H: R^n \rightarrow R$. It is given in equation (6) [35].

$$L(H) := EL(y, H(X)) \quad (6)$$

Where $L()$ is a smooth loss function and (X, y) is a testing data sampled from the training data D . The maximum depth is used as 5. It limits the number of nodes in the tree. The number of boosting stages is used as 100. Log is used for loss function. The maximum number of trees as iterations is used as 20. Learning rate is used as 0.02.

RESULTS

Pneumonia has been investigated in some studies [5]. With international cooperation, it is aimed to save 5.3 million lives by 2030 [5]. Every year, 12 November World Pneumonia Day is celebrated to raise awareness about pneumonia. The 2018 World Pneumonia Day theme is "Stop Pneumonia: Invest in Child Health" [1]. Additionally, There is currently a covid 19 pandemic. Covid -19 causes pneumonia and crowds in hospitals have become inevitable. In the face of hospital crowd, the doctors are getting tired. In this case, it can lead to misdiagnosis. On the other hand, radiologists may not be sufficient in some regions. This kind of tool will be helpful to diagnosis of the disease. As a result of all these reasons, this system is designed. Using this type of system, a diagnosis can be done quickly and accurately. So the treatment process can be started rapidly.

This study consists of two stages. In the first stage, diagnosis of pneumonia are done using boosting techniques. In the second stage, the automatic tool are designed. With this designed tool, the desired X-ray is loaded into the system and the user can see the result on the screen in a short time. Firstly, feature selection is applied to the dataset as mentioned above. Figure 6 shows the relation between the value of estimator and accuracy for Random Forest Classifier.

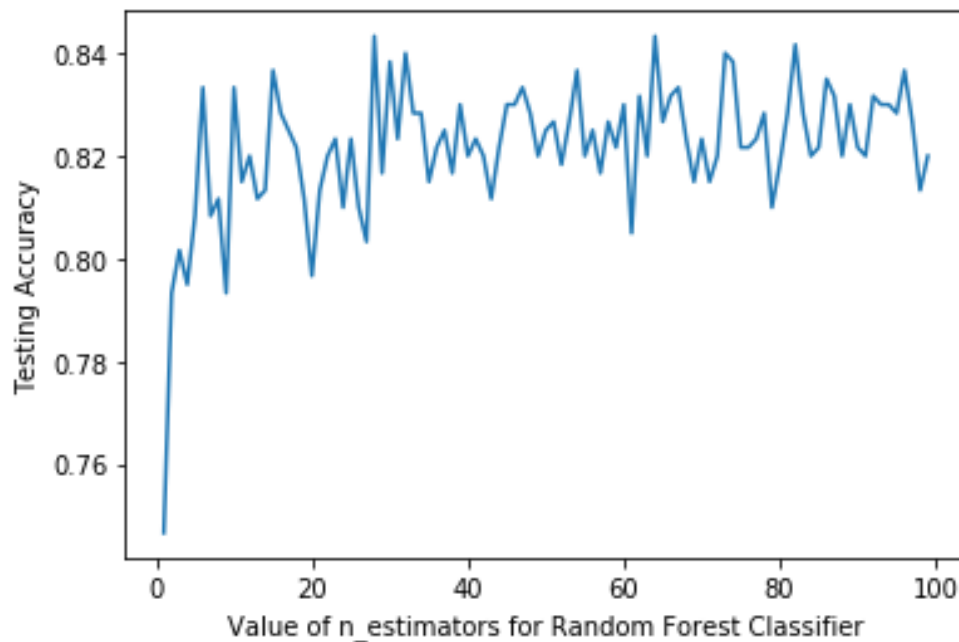


Figure 6. The relation between the value of estimator and accuracy for Random Forest Classifier

The function to measure the quality of a split Gini criteria is used for feature selection. The Gini Index calculates the probability of a particular feature being misclassified when randomly selected. Based on the Gini index of node impurity and classification accuracy of OOB data, are usually used. Given a node t and estimated class probabilities $p(k|t)$ $k = 1, \dots, Q$, the Gini index is shown in equation (7). Where Q is the number of classes [36].

$$G(t) = 1 - \sum_{k=1}^Q p^2(k|t) \quad (7)$$

Figure 7 shows the score distribution of the features. It is clear from Figure 7 that mostly feature scores are between 0 and 0.05. The threshold is randomly determined as 0.001 for this study. After the selection of the threshold, the training data dimension is 2400x172. Figure 8 shows after the feature selection of the data. The classifiers are XGBoost, LightGBM, GBM, and CatBoost for this study. All experiments were run on Intel Core i7 CPU 2.80 Ghz computer. After the feature selection, classifiers are used.

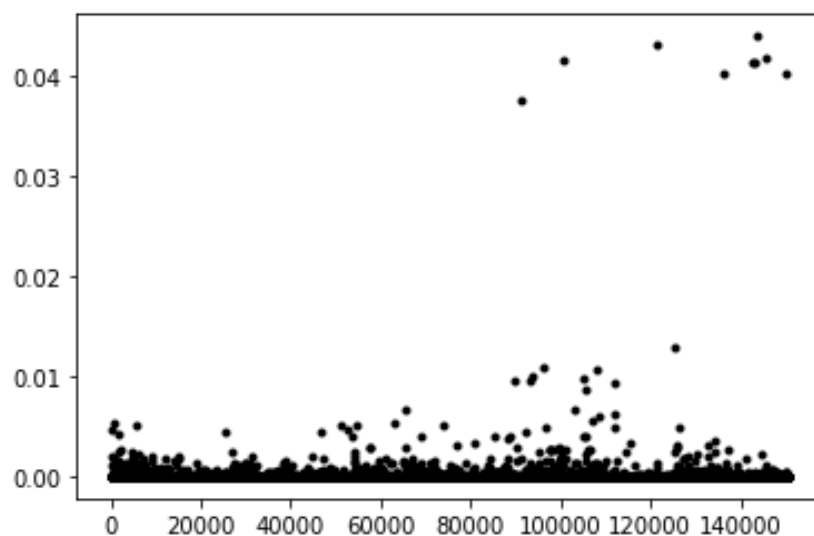


Figure 7. Score distribution

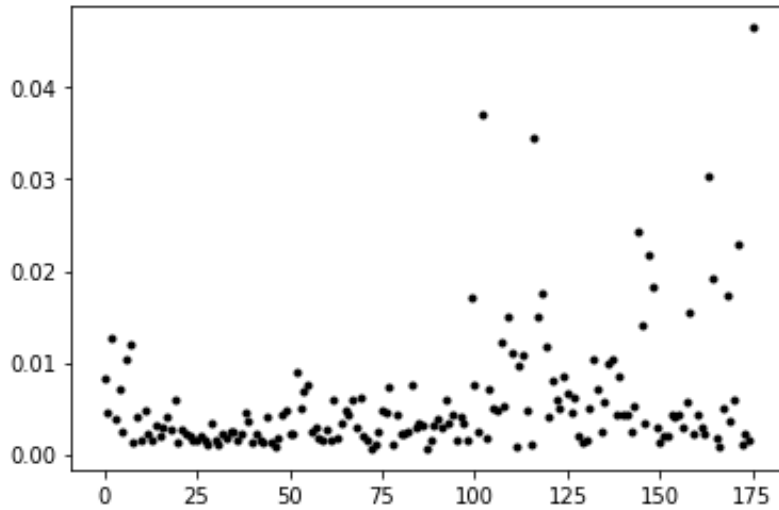


Figure 8. After the feature selection of the data.

Figure 9 shows the correlation of all columns in the data. This matrix shows the relationship between all features in a color palette. Pearson correlation is used. Correlation is a measure of how two variables change together. In the correlation matrix, the values are positioned between the dark color (black) and the light color (yellow). The values close to dark colors are interpreted as negative correlation, and the values close to light colors are interpreted as positive correlations. The values of two positively correlated variables increase or decrease together. As the value of one of the two negatively correlated variables increases, the other decreases. Correlation between x and y for Pearson's Correlation coefficient can be seen in Equation 8 [37].

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{[\sum_{i=1}^n (x_i - \bar{x})^2][\sum_{i=1}^n (y_i - \bar{y})^2]}} \tag{8}$$

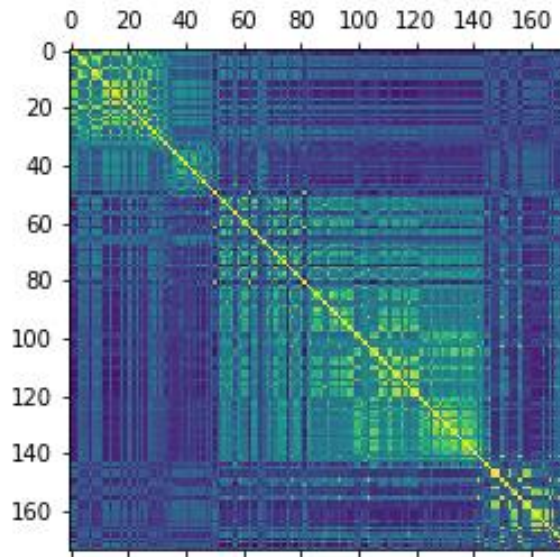
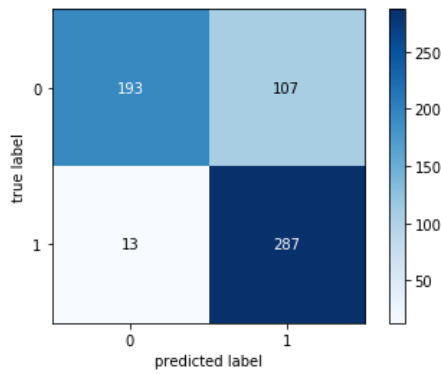
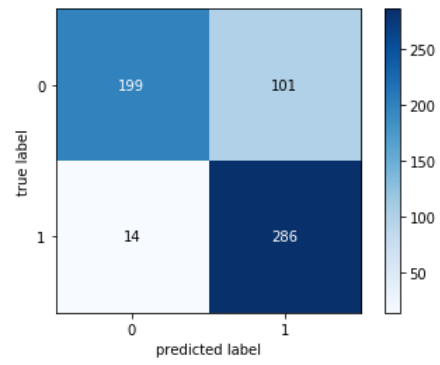


Figure 9. Correlation of the data after feature selection.

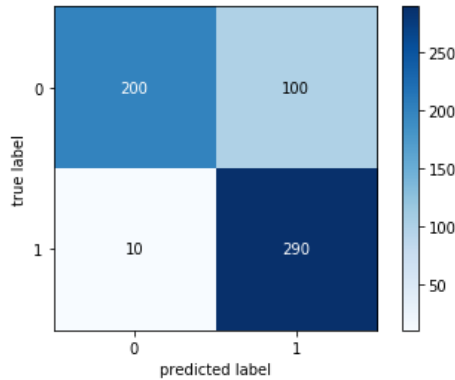
For performance evaluation of the classifiers, confusion matrices are obtained. Figure 10 shows the confusion matrix results for the classifiers. A confusion matrix is a table and is used to describe the performance of a classification model.



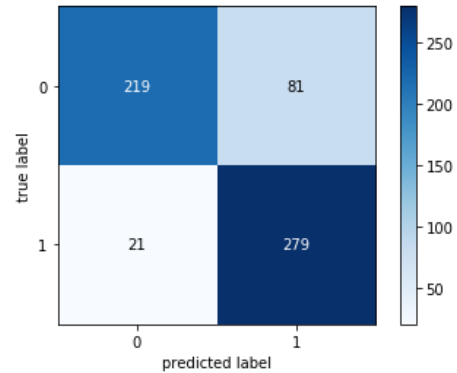
a) Confusion matrix for GBM



b) Confusion matrix for XGBoost



c) Confusion matrix for LightGBM



d) Confusion matrix for Catboost

Figure 10. Confusion matrix results a) Confusion matrix for GBM b) Confusion matrix for XGBoost, c) Confusion matrix for LightGBM, d) Confusion matrix for CatBoost

The accuracy value is calculated by the ratio of correctly predicted areas in the model to the total data set. TP, TN, FP, FN represent true positive, true negative, false positive, false negative, respectively. True positive shows the correctly predicted event values. False positive means incorrectly predicted event values. True negative is used for correctly predicted no-event values. False negative demonstrates incorrectly predicted no-event values. Equality of expressions used in confusion matrices based performance evaluation are given in Table 1 [38]. Accuracy, precision, F1-Score, sensitivity, specificity, matthews correlation coefficient (MCC), error rate, log-loss, are used as performance evaluations. Performance evaluation results are given in Figure 11 and Figure 12.

Table 1. Confusion matrices based performance evaluation

Accuracy	$\frac{TP + TN}{TP + FP + TN + FN}$
Precision	$\frac{TP}{TP + FP}$
F1-Score	$2 * \frac{\text{precision} * \text{sensitivity}}{\text{precision} + \text{sensitivity}}$
Sensitivity	$\frac{TP}{TP + FN}$
Specificity	$\frac{TN}{TN + FP}$
Matthews Correlation Coefficient (MCC)	$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$
Error Rate	$\frac{FP + FN}{TP + TN + FN + FP}$

When the performance results are analyzed, if MCC approaching 1, the relationship can be considered as strong. Sensitivity is the ability of a test to correctly identify patients with a disease. Specificity is the ability of a test to correctly identify people without the disease. For the diagnosis to be successful, it is important that it is both sensitive enough and specific enough. Sensitivity results are slightly closer to 1 than precision results. We can say that F1-Score, which is the harmonic average of sensitivity and precision values, gave more objective results. Error rate is calculated as the number of all incorrect predictions divided by the total number of the dataset. If the error rate is approaching zero, it is considered the best score.

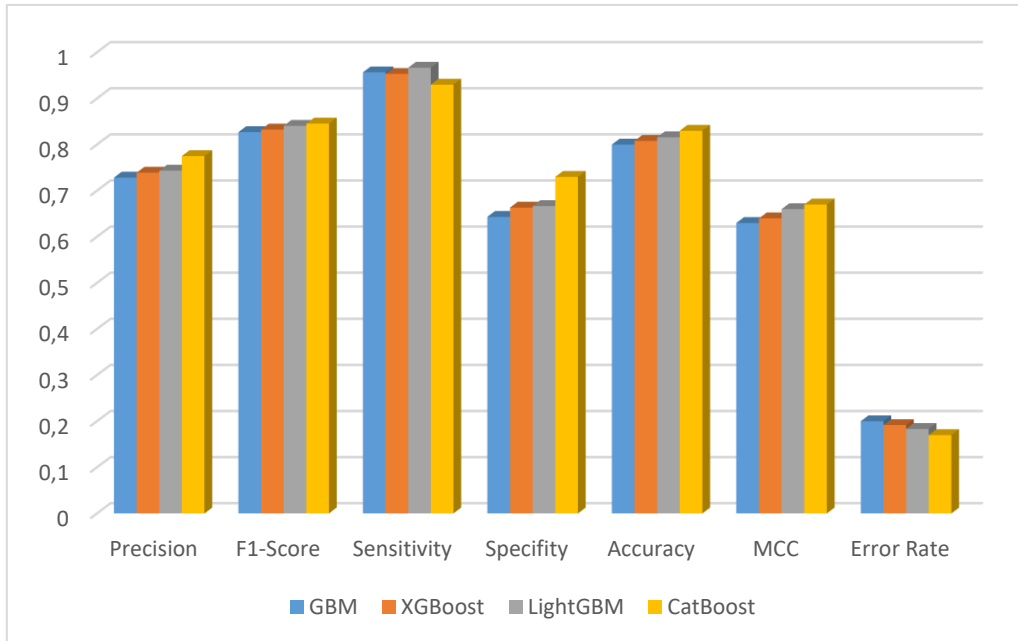


Figure 11. Performance evaluation results.

The log loss is defined for two or more labels. For a single sample with true label $y \in \{0,1\}$ and a probability estimate p , the log loss is defined as equation 9 [39].

$$\text{logloss} = (y \log(p) + (1 - y) \log(1 - p)) \tag{9}$$

The lower the log-loss value means the higher the model performance. Figure 12 shows that CatBoost has the smallest log loss score.

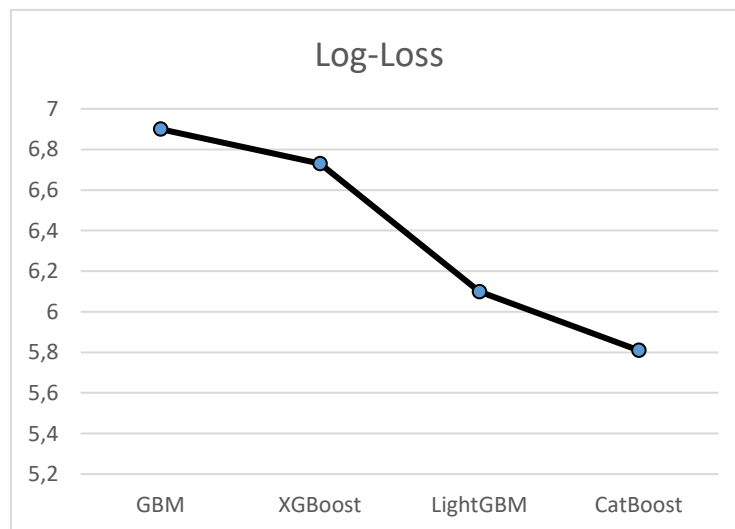


Figure 12. Log-Loss results.

Figure 13 shows the accuracy and running time results. The redline shows the accuracy result as the correct prediction rate for the techniques. Accuracy increases for the techniques as GBM (80%), XGBoost (81%), LightGBM (82%), CatBoost (83%), respectively. CatBoost is better than the other techniques for the

correct prediction rate. When the techniques are compared in terms of running times, it is seen that CatBoost has the shortest running time as 0,7 second. Running time shows the simulation duration. A fast algorithm is important for real-time applications and decision-making systems. When gradient boosting algorithms are examined in terms of time complexity, it has training time complexity as $O(npn_{trees})$, prediction time complexity as $O(pn_{trees})$. n is the number of training sample, p is the number of features, n_{trees} is the number of trees. [40]. The complexity of an algorithm/model is expressed using the Big O Notation. Time complexity shows the time for completing the task of the algorithm. It is clear that boosting techniques is effective about time complexity.

If the Accuracy score approaches 1, the model is considered as successful. As seen in Figure 13, the accuracy value of the Catboost model that intersects with the red line is the highest. On the left axis, the simulation time of the algorithms is seen. Catboost has the fastest operating performance with 0.7sec. As a result, Performance for CatBoost seems more sensible from figures 11-12-13.

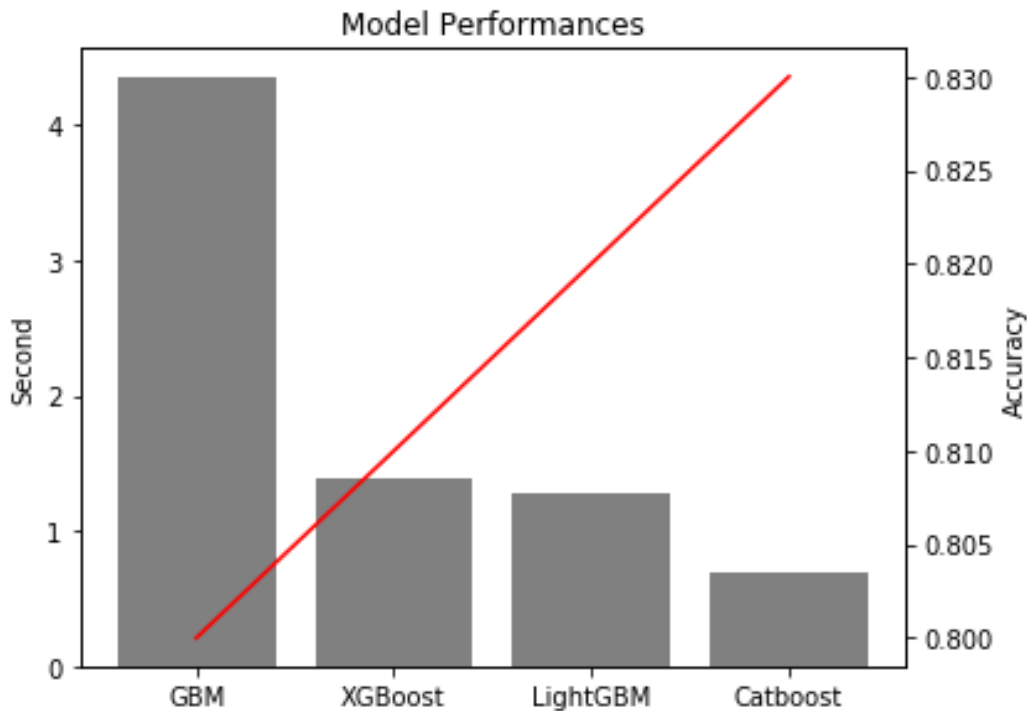


Figure 13. Accuracy and running time (sec) results.

The second stage for this study is to design the automatic tool. After the modeling phase is completed, the interface is designed to use the trained model effectively. Categorical Boosting model is used in this interface. Because it has the shortest prediction time. If the prediction time is short, automatic tool gives a result quickly. Figure 14 shows the general diagram for the designed system.

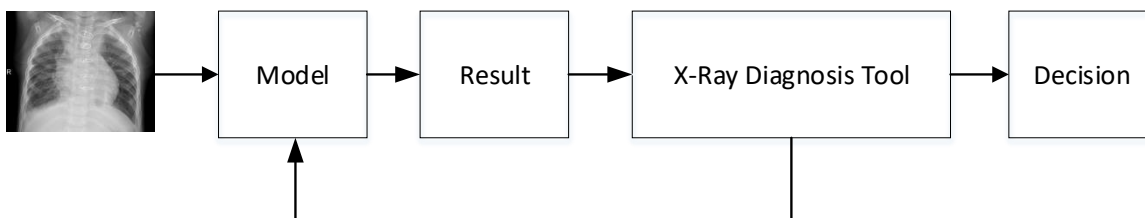


Figure 14. General diagram for the designed system

To accomplish a goal like diagnosis of pneumonia, code is written and executed in the IDE, and it produces an output either in the terminal or in the IDE. Graphical User Interface (GUI) helps to interact with the computer. With the GUI design, the results obtained from the code become more understandable and the code can be used more easily. Thus, a user who does not have an information about the subject, can plan the next process by looking at the result provided by the interface. In this study Tkinter is used for automatic tool. Tkinter is Python's standard GUI framework [41]. Figure 15 shows the design for X-Ray Diagnosis Tool.

Window manager helps in controlling the size, position, and other attributes of the window. Tkinter Label is a widget that is used to implement display boxes. The text showed by this widget can be changed by the developer. To use a label, you just have to specify what to display in it. Geometry management is used to arrange widgets on a window.

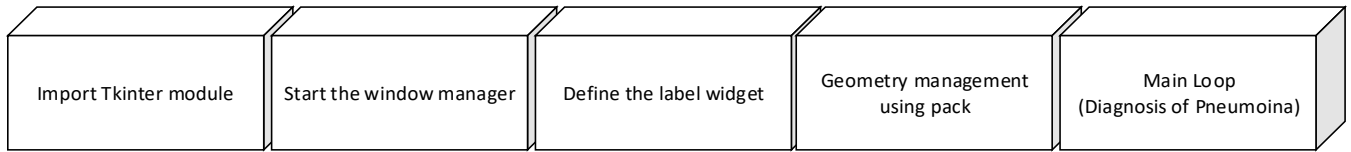


Figure 15. Design for X-Ray Diagnosis Tool.

Figure 16 shows the tool for the user. There are two buttons in the tool. *Open X-Ray* button and *Show Result* button. When the *Open X-Ray* button is clicked, it enables the X-ray to be loaded from the relevant folder.

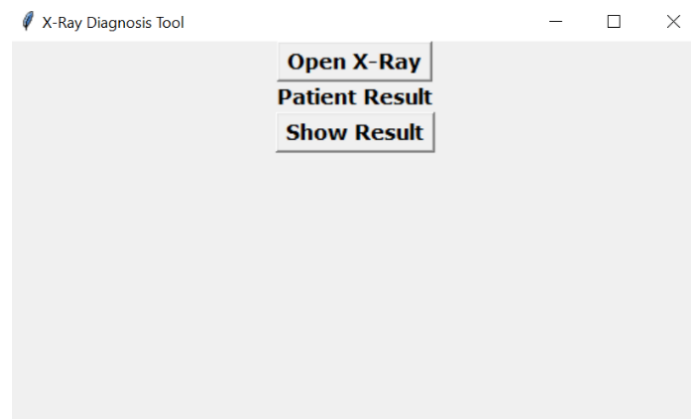


Figure 16. Tool for the user

X-ray loaded from the computer is sent to the model with this designed interface. As a result of the code, the outcome of the X-ray is displayed on the interface. Thus, the output of the model is transferred to real life by using the interface easily. Figure 17 shows the screen that when the *Open X-Ray* button is clicked.

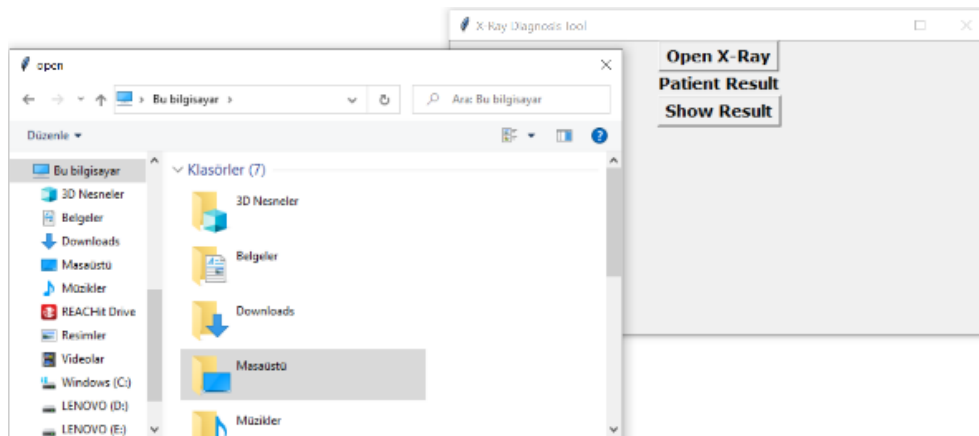


Figure 17. Opening of X-Ray using button.

After the selection of the X-ray from the folder, it is loaded. Figure 18 shows the after selection of the X-Ray. X-Ray is seen on the tool.

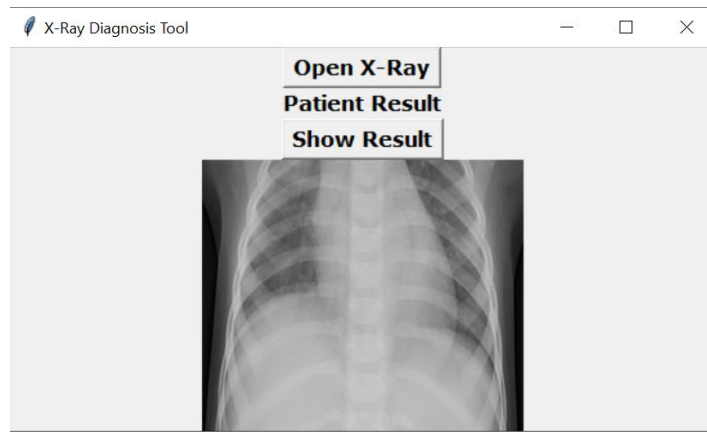


Figure 18. Loaded X-Ray from computer.

Figure 19 shows the piece of code for the automatic tool. When the *Show Result* button is clicked to see the patient result, *degistir* function is called by the button. In this function, *catboost* model is used for prediction. the X-ray is applied to the *CatBoost* model and the patient result is seen on the screen.

```
def degistir():
    p=cat_model.predict(imgx)
    if p == 1:
        s="No detection"
    elif p == 0:
        s="Pneumonia Detected"
    yazi.config(text=s,font =('Verdana', 12,"bold"))
    yazi=Label(root)
    yazi.config(text="Patient Result",font =('Verdana', 12,"bold"))
    yazi.pack()
    dugme=Button(root)
    dugme.config(text="Show Result",font =('Verdana', 12,"bold"))
    dugme.config(command=degistir)
    dugme.pack()
```

Figure 19. The piece of code for the automatic tool

This situation is shown in Figure 20. In this example, pneumonia is detected for this X-Ray. The result is shown as *Pneumonia Detected* on the tool.

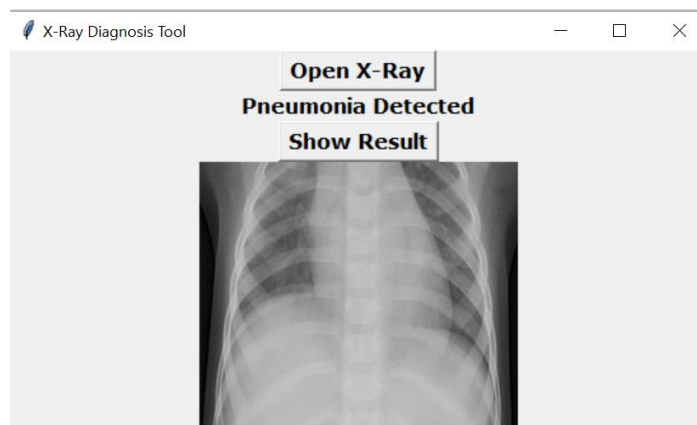


Figure 20. Result for x-ray with the tool.

As a result, this study contains the design of an automatic tool for the diagnosis of pneumonia using Boosting techniques. It has two stages. At the first stage, Boosting techniques have been analyzed. The main aim of this study to reach the result quickly using the interface. It is important that the model is fast and has a high accuracy rate. Among these techniques, the best result in terms of simulation duration and accuracy is *Catboost* with 0.7 seconds prediction time and 83% accuracy.

The second stage of this study includes a user interface tool. It has been designed for the diagnosis of pneumonia using Boosting methods. The automatic tool has been developed that everyone can use and

interpret the classification results easily. By using this automatic tool, the user can be able to upload the desired X-ray image to the system and get the result from the model easily. Thus, the diagnosis result is seen from the screen without any radiologists/doctors. The diagnosis of the disease can be made with a tool not needing any specialist.

Using this tool, the appointment can set according to the diagnosis status by the secretary. Thus, patients whose conditions are urgent will be directed to the doctor first. Besides, with preliminary evaluation by this tool, the hospital crowd will be prevented. This diagnostic tool is very important in terms of making a pre-diagnosis automatically and quickly.

CONCLUSION

This study includes the automatic tool design that enables Pneumonia diagnosis to be done quickly and accurately by anyone using X-ray. In fact, a bridge has been created between the trained model and the user. Thus, diagnosis can be done using the tool without the need for an expert. Today, the main problems we face due to Covid 19 are the crowd of patients in hospitals and the insufficient number of radiologists/doctors. Long-term diagnoses cause increased hospital crowded as well as a severe course of diseases. One of the diseases caused by Covid 19 is pneumonia. The most common method for the diagnosis of pneumonia is X-rays. In this study, X-ray images obtained from the patient are evaluated with the developed software tool. Boosting techniques are used for decision. Among these techniques, the best result in terms of speed and accuracy is Catboost with 0.7 seconds running time and 83% accuracy. Later, The automatic tool is designed. The Automatic tool is a graphical user interface that makes the results obtained from the model more understandable and helps the user to interact with the code. But there is a limitation of the proposed method. If the dataset increases over time, it needs to be retrained to diagnose with higher accuracy.

In the face of a global epidemic like Covid 19, the hospital crowd is inevitable. In the face of hospital crowd, the doctors are getting tired. In this case, it can lead to misdiagnosis. On the other hand, radiologists may not be sufficient in some regions. This kind of tool will be helpful to diagnose the disease for doctor/radiologists.

In addition, there is no guarantee that there will be no other pandemic situations in the coming years. As a result of all these reasons, a system has been designed that can automatically diagnose x-ray results. Based on this study, studies can be carried out to diagnose different diseases with different models. The system can be developed and appointment dates can be automatically assigned according to the severity of the diagnosed disease.

REFERENCES

1. Singh N, Sharma R, Kukker A. Wavelet Transform Based Pneumonia Classification of Chest X-Ray Images. International Conference on Computing, Power and Communication Technologies (GUCON); 2019; New Delhi, India, p. 540-545.
2. Irfan A, Adivishnu AL, Sze-To A, Dehkharghanian T, Rahnamayan S, Tizhoosh HR. Classifying Pneumonia among Chest X-Rays Using Transfer Learning. 42nd International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC); 2020; Montreal, Canada, p. 2186-2189.
3. Mubarak AF, Dominique JAM, This AH. Pneumonia Detection with Deep Convolutional Architecture. International Conference of Artificial Intelligence and Information Technology (ICAIT); 2019; Yogyakarta, Indonesia, p. 486-489.
4. Sharma H, Jain JS, Bansal P, Gupta S. Feature Extraction and Classification of Chest X-Ray Images Using CNN to Detect Pneumonia. 10th International Conference on Cloud Computing, Data Science & Engineering; 2020; Noida, India, p. 227-231.
5. Save the children fighting for breath- A call to action on childhood pneumonia: Save the Children 1st John's Lane; 2017 [cited 05.06.2021]. 83p. Available from: <https://www.savethechildren.org.uk/content/dam/global/reports/health-and-nutrition/fighting-for-breath-low-res.pdf>
6. Jain R, Nagrath P, Kataria G, Kaushik VS, Hemanth DJ. Pneumonia detection in chest X-ray images using convolutional neural networks and transfer learning. Measurement. 2020; 165:1-10.
7. Yu X, Wang SH, Zhang YD. CGNet: A graph-knowledge embedded convolutional neural network for detection of pneumonia. Inf Process Manag. 2021; 58(1):1-25.
8. Li Y, Zhang Z, Dai C, Dong Q, Badrigilan S. Accuracy of deep learning for automated detection of pneumonia using chest X-Ray images: A systematic review and meta-analysis. Comput. Biol. Med. 2020; 123:1-8.
9. Ge Y, Wang Q, Wang L, Wu H, Peng C, Wang J, Xu Y, Xiong G, Zhang Y, Yi Y. Predicting post-stroke pneumonia using deep neural network approaches. Int.J. Med. Inform. 2019; 132:1-8.

10. Chassagnon G, Vakalopoulou M, Battistella E, Christodoulidis S, Hoang-Thi TN, Dangeard S, et al. AI-driven quantification, staging and outcome prediction of COVID-19 pneumonia. *Med. Image Anal.*2021; 67:1-16.
11. Wang SH, Nayak DR, Guttery DS, Zhang X, Zhang YD. COVID-19 classification by CSHNet with deep fusion using transfer learning and discriminant correlation analysis. *Inf Fusion.* 2021 Apr; 68:131-148.
12. Shuihua W, Xiaosheng W, Yu-Dong Z, Chaosheng T, Xin Z. Diagnosis of COVID-19 by Wavelet Renyi Entropy and Three-Segment Biogeography-Based Optimization. *Int. J. Comput. Intell.* 2020;13(1):1332-44.
13. Postalcioglu S, Keşli A. Diagnosis of Pneumonia by Naive Bayes Method; 3rd International Conference on Data Science and Applications (ICONDATA'20); 2020 June 25-28; Istanbul, TURKEY, p. 208-211.
14. Ramezanpour A, Beam AL, Chen JH, Mashaghi A. Statistical Physics for Medical Diagnostics: Learning, Inference, and Optimization Algorithms. *Diagnostics.* 2020; 10(11): 1-16.
15. Khan MA, Ashraf I, Alhaisoni M, Damaševičius R, Scherer R, Rehman A, Bukhari SAC. Multimodal Brain Tumor Classification Using Deep Learning and Robust Feature Selection: A Machine Learning Application for Radiologists. *Diagnostics.* 2020; 10(8):1-19.
16. Galván-Tejada CE, Zanella-Calzada LA, Galván-Tejada JI, Celaya-Padilla JM, Gamboa-Rosales H, Garza-Veloz I, Martínez-Fierro ML. Multivariate Feature Selection of Image Descriptors Data for Breast Cancer with Computer-Assisted Diagnosis. *Diagnostics.* 2017; 7(1):1-17.
17. Rahman S, Irfan M, Raza M, Moyeezullah KG, Yaqoob S, Awais M. Performance Analysis of Boosting Classifiers in Recognizing Activities of Daily Living. *Int J Environ Res Public Health.*2020 Feb;17(3):1-15.
18. Rocca J, Ensemble methods: bagging, boosting and stacking, [Internet]. [cited 05.06.2021]. Available from: <https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205>
19. Devhunter. Gradient Boosting. [Internet]. [cited 01.11.2020]. Available from: <https://devhunteryz.wordpress.com/2018/07/11/gradyan-artirmagradient-boosting/>
20. Binder H, Gefeller O, Schmid M, Mayr A. The Evolution of Boosting Algorithms. *Methods of Information in Medicine.* 2014; 53(6): 419–427.
21. Chest x-ray pneumonia, [Internet]. [cited 14.06.2020]. Available from: <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>
22. Rorasa. l0 norm, l1 norm, l2 norm, l infinity norm, [Internet]. [cited 01.11.2020] Available from: <https://rorasa.wordpress.com/2012/05/13/l0-norm-l1norm-l2-norm-l-infinity-norm>
23. Abdullahi A, Raheem L, Muhammed M, Rabiat OM, Saheed AG. Comparison of the CatBoost Classifier with other Machine Learning Methods. *Int. J. Adv. Comput. (IJACSA).*2020;11(11):738-748.
24. Reif D, Alison M, Mckinney B, Crowe J, Moore J. Feature Selection using a Random Forests Classifier for the Integrated Analysis of Multiple Data Types. *EEE Symposium on Computational Intelligence and Bioinformatics and Computational Biology*; 28-29 Sept. 2006; Canada;p. 1-8. 10.1109/CIBCB.2006.330987.
25. Alexey N, Knoll A. Gradient Boosting Machines, A Tutorial. *Frontiers in Neurobotics.* 2013; 7:1-21.
26. Logistic Regression: Loss and Regularization, [Internet]. [cited 05.06.2021] Available from: <https://developers.google.com/machine-learning/crash-course/logistic-regression/model-training>
27. Muratlar ER. LightGBM. [Internet]. [cited 06.12.2020]. Available from: <https://www.veribilimiokulu.com/lightgbm/>
28. Minastireanu E, Mesnita G. Light GBM Machine Learning Algorithm to Online Click Fraud Detection. *Journal of Information Assurance & Cybersecurity.* 2019; 2019:1-12.
29. Gumus M, Kiran MS. Crude oil price forecasting using XGBoost. *International Conference on Computer Science and Engineering (UBMK)*; 2017. Antalya, p. 1100-1103. doi: 10.1109/UBMK.2017.8093500.
30. Wang Y, Guo Y. Forecasting method of stock market volatility in time series data based on mixed model of ARIMA and XGBoost. *China Communications.* March 2020; 17(3):205-221.
31. Long J, Yan Z, Shen Y, Liu W, Wei Q. Detection of Epilepsy Using MFCC-Based Feature and XGBoost. 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI);2018; Beijing, China; p. 1-4. doi: 10.1109/CISP-BMEI.2018.8633051.
32. Liao X, Cao N, Li M, Kang X. Research on Short-Term Load Forecasting Using XGBoost Based on Similar Days. *International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS)*;2019; Changsha, China; p. 675-678. doi: 10.1109/ICITBS.2019.00167.
33. Muratlar ER. CatBoost Nedir? Diğer Boosting Algoritmalarından Farkı Nelerdir?. [Internet] [cited 01.12.2020]. Available from: <https://www.veribilimiokulu.com/catboost-nedir-diger-boosting-algoritmalarindan-farki-nelerdir/>
34. Dorogush A, Ershov V, Gulin A. CatBoost: Gradient boosting with categorical features support. *Proc. Workshop ML Syst. Neural Inf. Process. Syst. (NIPS)*;2017 [cited 01.12.2020]; pp. 1-7. Available from: <https://arxiv.org/pdf/1810.11363.pdf>
35. Abdullahi AI, Raheem L, Muhammed M, Muhammed RO, Ganiyu AS. Comparison of the CatBoost Classifier with other Machine Learning Methods. *Int. J. Adv. Comput. (IJACSA).* 2020;11(11): 738-48.
36. Nguyen C, Wang Y, Nguyen N. Random forest classifier combined with feature selection for breast cancer diagnosis and prognostic . *Journal of Biomedical Science and Engineering.* 2013;6(5):551-60.
37. Mukaka M. Statistics Corner: A guide to appropriate use of Correlation coefficient in medical research. *Malawi Med. J.* 2012; 24(3): 69-71 .
38. Powers DA. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation. *J. Mach. Learn. Technol.* 2011; 2: 2229-3981. doi: 10.9735/2229-3981.
39. Khanna R. MachineX: Evaluation Metrics for Classification Models. December 2, 2019. [Internet]. [cited 05.12.2020]. Available from: <https://blog.knoldus.com/machinex-evaluation-metrics-for-classification-models/>

40. Computational complexity of machine learning algorithms. [Internet]. [cited 05.06.2021]. Available from: <https://www.thekerneltrip.com/machine/learning/computational-complexity-learning-algorithms/>
41. Sharma A. Introduction to GUI With Tkinter in Python. [Internet]. December 10 2019. [cited 16.06.2021] Available from: <https://www.datacamp.com/community/tutorials/gui-tkinter-python>



© 2022 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY NC) license (<https://creativecommons.org/licenses/by-nc/4.0/>).