Predicting Lung Cancer Survivability : a Machine Learning Ensemble Method on Seer Data

Surraj Kumar P B.TECH - IT , Dr. Lilly Raamesh HOD - IT
St.Joseph's College of Engineering

Abstract

Ensemble methods are powerful techniques used in machine learning to improve the prediction accuracy of classifier learning systems. In this study, different ensemble learning methods for lung cancer survival prediction were evaluated on the Surveillance, Epidemiologyand End Results (SEER) dataset. Data were preprocessed in several steps before applying classification models. The popular ensemble methods Bagging, Adaboost and three classification algorithms, K-Nearest Neighbours, Decision Tree and Neural Networks as base classifiers were evaluated for lung cancer survival prediction. The results empirically showed that ensemble methods are able to evaluate the performance of their base classifiers and they are appropriate methods for analysis of cancer survival.

Keywords: *Machine Learning, Ensemble, K-Nearest Neighbours, Decision Tree, Neural Network*

I. INTRODUCTION

Lung cancer is the second most common cancer, and the leading cause of cancer related deaths among men and women in the USA [8]. Survival rate for lung cancer is estimated to be 15% after 5 years of diagnosis [9]. The Surveillance, Epidemiology, and End Results (SEER) Program [10] of the National Cancer Institute is an authoritative repository of cancer statistics in the United States [11]. It is a population-based cancer registry which covers about 26% of the US population across several geographic regions and is the largest publicly available domestic cancer dataset. The data includes patient demographics, cancer type and site, stage, first course of treatment, and follow-up vital status. The SEER program collects cancer data for all invasive and in situ cancers, except basal and squamous cell carcinomas of the skin and in situ carcinomas of the uterine cervix [9]. The 'SEER limited-use data' is available from the SEER website on submitting a SEER limited-use data agreement form. Ries et al. [12] presents an overview study of the cancer data at all sites combined and on selected, frequently occurring cancers from the SEER data. The SEER data attributes can be broadly classified as demographic attributes (e.g., age, gender, location), diagnosis attributes (e.g., primary site, histology, grade, tumor size), treatment attributes (e.g., surgical procedure, radiation therapy), and outcome attributes (e.g., survival time, cause of death), which makes the SEER data ideal for performing outcome analysis studies.

Applying data mining techniques to cancer data is useful to rank and link cancer attributes to the survival outcome. Further, accurate outcome prediction can be extremely useful for doctors and patients to not only estimate survivability, but also aid in decision making to determine the best course of treatment for a patient, based on patient-specific attributes, rather than relying on personal experiences, anecdotes, or population wide risk assessments. Experiments with several classifiers were conducted to find that many meta classifiers used with decision trees can give impressive results, which can be further improved by combining the resulting prediction probabilities from several classifiers using an ensemble bagging and Adaboost scheme.

The goal of ensemble learning methods as a subgroup of machine learning algorithms is to increase the performance of base classifiers by creating an ensemble of multiple classifiers and combining the results. Different methods have been proposed to create ensemble of classifiers. Two of the most common methods are: using different subsets of training data with a single learning method and using different learning methods on the same data.

II. LITERATURE REVIEW

Agrawal et al. [1] used ensemble voting method to predict lung cancer survival after 6months, 9-months, 1-year, 2-years and 5-years of diagnosis. They analyzed the data from SEER program to construct an ensemble of five decision tree algorithms and combined the results by using the average of probabilities generated by each classifier. They used 10-fold crossvalidation for training and testing and compared the performance of ensemble voting technique with individual classifiers to show the effect on ensemble data mining on increasing performance of weak classifiers.

Zolbanin et al. [2] investigated the overall survivability in comorbidity of cancer on SEER data. They joined four data files corresponding to urinary, male genital, female genital and breast cancers and sorted them by increasing case numbers and added ten attributes to each data file (nine attributes for each cancer categories and one counter attribute). They joined the data files using SQL procedures in SAS Enterprise guide 6.1. They used SAS Enterprise Miner to test Random Forest, Neural Network, Logistic Regression and Decision Tree algorithms to predict survivability in comorbidity of cancers and concluded that random forest algorithm outperforms other classification models.

Khoshgoftaar et al. [3] compared the performance of eight boosting and Bagging methods on imbalanced and noisy data. They evaluated the results with seven different metrics for both balanced and imbalanced data and they used analysis-of variance for testing statistical significance of obtained results. They concluded that the Bagging method outperforms boosting in noisy environment with class imbalanced data.

Delen et al. [4] used two data mining algorithms (C5 and artificial neural networks) along with a statistical method (logistic regression) on SEER data to predict breast cancer survivability. They preprocessed the data before applying classification algorithms and compared the results in terms of accuracy. The results empirically showed that C5 algorithm with 93.6% accuracy outperforms artificial neural networks with 91.2% accuracy and logistic regression with 89.2% accuracy.

Zheng et al. [5] proposed a hybrid of K-means and support vector machine (K-SVM) to diagnose breast cancer based on extracted tumor features. They used K-means to discover hidden patterns of tumors and SVM algorithm to classify them. Wisconsin Diagnostic Breast Cancer (WDBC) data were used to evaluate the performance of proposed methodology. The dataset contained 569 samples and 32 features.6 features were selected by K-means algorithm. The proposed method increased the accuracy to 97.38%.

DendiGayathri Reddy, Emmidi Naga Hemanth Kumar, Desireddy Lohith Sai Charan Reddy and Monika P[6] proposed a system to predict multiple levels of lung cancer by ensemble method. The dataset was taken from Data World Source, which has 1000 data records. The paper explains about predicting various carcinoma stages using ML concepts. The paper put forth a method which uses an amalgamation of three algorithms – KNN, neural networks and decision trees with bagging. KNN is a data sensitive by nature which uses Euclidean distance. It understands the data. To correct the errors, backpropagation of neural networks is used. Next, CART algorithm was used for classification purpose. To reduce the variance of cost, bagging was used. This integrated model gave an accuracy of 98%.

JaneeAlam, Sabrina Alam and Alamgir Hossan [7] submitted their work on prediction of multistage lung cancer detection with SVM classifier. The dataset used here contained 500 lung CT images. The software tool used here was MATLAB to process the image. If the input image contains no affected cell, then probability of the disease is diagnosed. For the purpose of image enhancement, masking is done. For gaining better resolution of image, watershed transform for segmentation is applied. GLCM technique is used for feature extraction. Next, the SVM classifier is applied. This works gave 97% of detection accuracy and 87% of prediction accuracy.

III. PROPOSED METHOD

In this paper, bagging and Adaboost ensemble learning methods for lung cancer survival prediction were evaluated on the SEER dataset. A multistep data preprocessing applied before classification stage. The popular ensemble methods including Bagging and AdaBoost and three classification algorithms including Decision Tree, K-Nearest Neighbour and Neural Network as base classifiers were evaluated for lung cancer survival prediction. The proposed system architecture showed in Fig. 1. ***a) Data Gathering***

SEER data released in April 2017 were used. The data file for lung cancer survival analysis which contains 149 attributes and 1000 samples were used. Several steps for data cleaning and transformation were performed before applying classification methods. The main attributes which contributes for lung cancer prediction include smoking, gender, air pollution, chronic lung disease, chest pain, wheezing, dry cough, snoring, swallowing difficulty and clubbing.

b) Data Pre-processing

The data is interpolated for missing data and then normalized using linear transformation algorithm [8] as in (1).

$$Y = \frac{Y = (X - min(X))}{(max(X) - min(X))} \quad \ldots\ldots\ldots\ldots\ldots(1)$$

The normalized data is split into training (80%) and testing (20%) dataset. The training dataset is split into n parts in order to train n models using the concept of bagging.

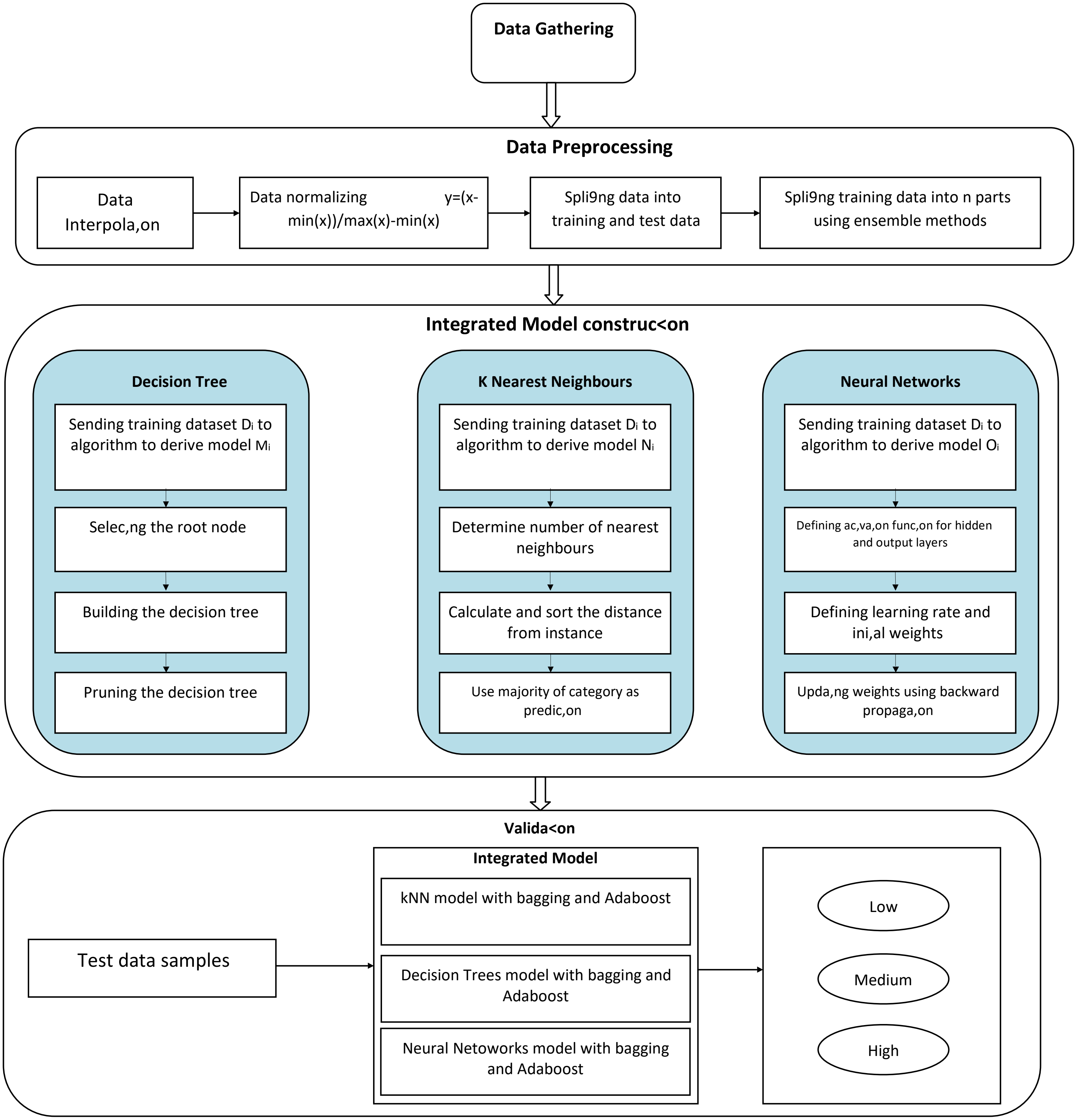

Data Gathering
Data Preprocessing
Data Interpola,on
Data normalizing y=(x-min(x))/max(x)-min(x)
Spli9ng data into training and test data
Spli9ng training data into n parts using ensemble methods
Integrated Model construc<on
Decision Tree
Sending training dataset Di to algorithm to derive model Mi
Selec,ng the root node
Building the decision tree
Pruning the decision tree
K Nearest Neighbours
Sending training dataset Di to algorithm to derive model Ni
Determine number of nearest neighbours
Calculate and sort the distance from instance
Use majority of category as predic,on
Neural Networks
Sending training dataset Di to algorithm to derive model Oi
Defining ac,va,on func,on for hidden and output layers
Defining learning rate and ini,al weights
Upda,ng weights using backward propaga,on
Valida<on
Test data samples
Integrated Model
kNN model with bagging and Adaboost
Decision Trees model with bagging and Adaboost
Neural Netoworks model with bagging and Adaboost
Low
Medium
High

Fig.1 Proposed architecture

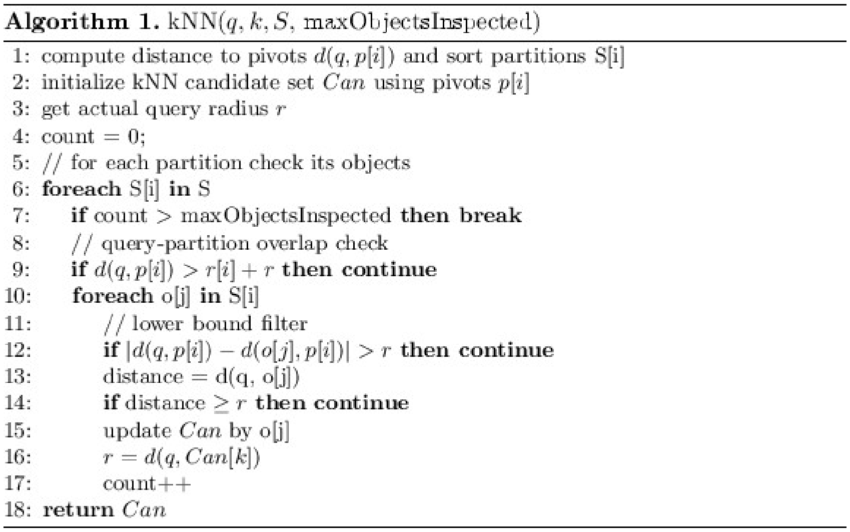
c) Model Implementation

The n-parts of training dataset are fed as input to Decision Tree, K-Nearest Neighbour and Neural Network algorithms to create n- models of each algorithm. The group of n-models of each algorithm forms a bagging model. Final bagging and Adaboost models generated from the algorithms are termed as integrated model and are used for testing and future predictions. ***i. K-Nearest Neighbour***

K-nearest neighbours' algorithm (*k*-NN) is a nonparametric technique used in regression and classification problems. *k* - closest training examples in the feature space acts as an input. The predicted class, to which an object belongs to, depends upon the class of the neighbours around it. *K-NN* is a lazy learning algorithm as it does not learn from the training data but simply memorizes the training data. The algorithm is sensitive to the local structure of the data.

Euclidean distance as in (2) is used as a distance function.

$$Euclideandistance = \sqrt{\sum_{i=1}^{k}(xi - yi)^2} \quad \ldots\ldots\ldots\ldots\ldots\ldots.. (2)$$



```
Algorithm 1. kNN($q, k, S$, maxObjectsInspected)
compute distance to pivots $d(q, p[i])$ and sort partitions S[i]
initialize kNN candidate set $Can$ using pivots $p[i]$
get actual query radius $r$
count = 0;
// for each partition check its objects
foreach S[i] in S
    if count > maxObjectsInspected then break
    // query-partition overlap check
    if $d(q, p[i]) > r[i] + r$ then continue
    foreach o[j] in S[i]
        // lower bound filter
        if $|d(q, p[i]) - d(o[j], p[i])| > r$ then continue
        distance = d(q, o[j])
        if distance $\geq r$ then continue
        update $Can$ by o[j]
        $r = d(q, Can[k])$
        count++
return $Can$
```

ii. Decision tree

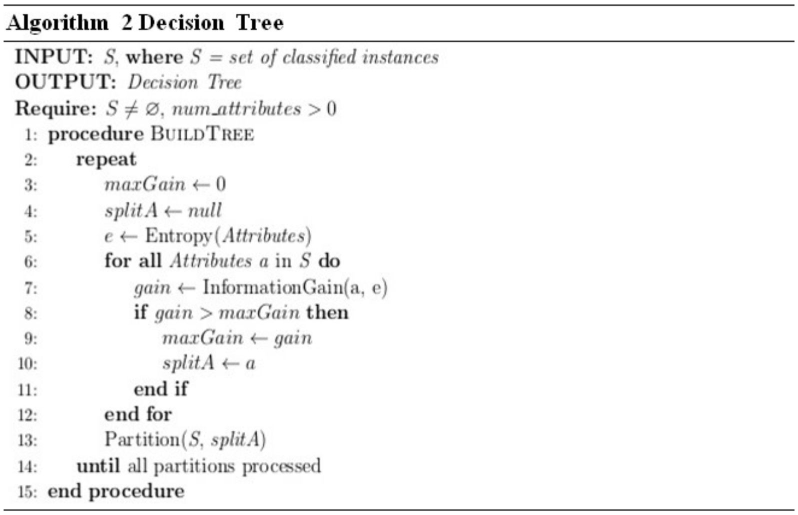
Decision Tree are forms of supervised machine learning where the information is constantly divided according to a pattern that fits. Two entities describe the tree, namely nodes and leaves of the decision. The decision nodes are the nodes where the data is to be split into child nodes and leaves. The structure of decision tree is shown below. In order to check which node will go to be used as the left node, it is important to determine the information gain value, i.e. the highest information gain value node is chosen. To calculated gain, entropy is needed given as:

$$H(S) \sum_{c\varepsilon C} -p(c)\, log_2\, p(c) \text{ ……….(3)}$$

Here H(S) is the entropy, C is the set of classes, and S is the data, p (c) is the probability of C with respect to S. Use this entropy to calculate Information gain which is given below:

$$IC(A,S) = H(S) - \sum_{t\epsilon T} p(t)H(t) \text{ ………..(4)}$$

Here, H(S) defines entropy, T is the subset on which decision to be made, p(t) gives the probability of T with respect to S, H(t) is entropy on subset T.

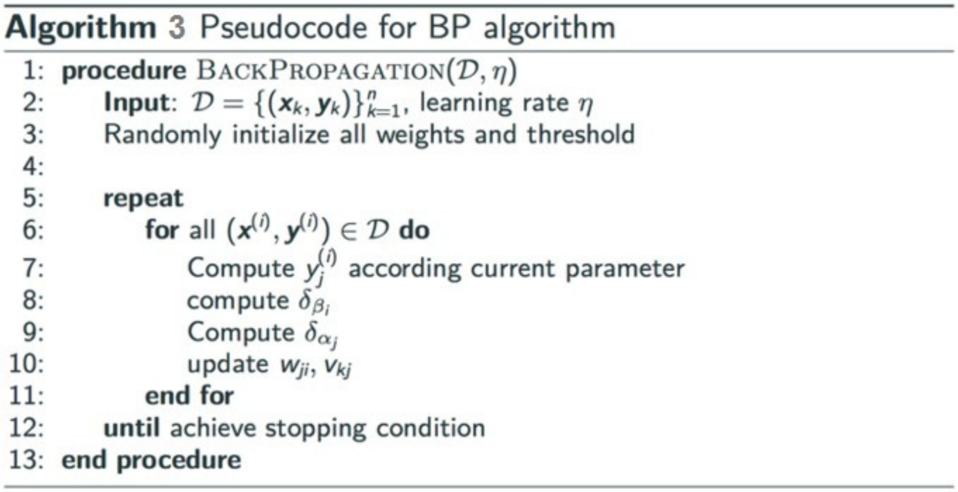


Algorithm 2 Decision Tree

```
INPUT: $S$, where $S$ = set of classified instances
OUTPUT: Decision Tree
Require: $S \neq \varnothing$, $num\_attributes > 0$
procedure BUILDTREE
    repeat
        $maxGain \leftarrow 0$
        $splitA \leftarrow null$
        $e \leftarrow$ Entropy($Attributes$)
        for all $Attributes$ $a$ in $S$ do
            $gain \leftarrow$ InformationGain(a, e)
            if $gain > maxGain$ then
                $maxGain \leftarrow gain$
                $splitA \leftarrow a$
            end if
        end for
        Partition($S$, $splitA$)
    until all partitions processed
end procedure
```

iii. Neural Networks

Backpropagation technique is predominantly used in Neural Networks for making predictions. Some of the data was input as a training data, then knowledge and information were gained from the training process is used as a reference for the identification of lung cancer by using Backpropagation Neural Network. An initial stage in the training process is the input of training data. Then specify the target output of each data input. After the training process backpropagation network, then further testing. In backpropagation network testing phase is done simply by implementing a forward direction (feedforward). The data used in the test is the data that is not used during training. The weights used in the phase of the forward direction is the weight training process results. Then the calculation of the output value of each node in the hidden layer and output layer. After testing of the output of each node in the output layer. Training and adjusting of weights is done by Backpropagation. The following algorithm shows the simple implementation steps of Backpropagation.



Algorithm 3 Pseudocode for BP algorithm

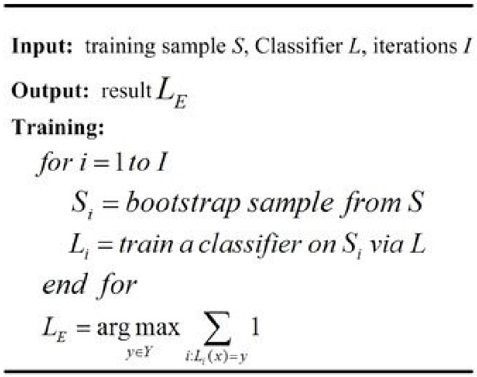
```
procedure BACKPROPAGATION($\mathcal{D}, \eta$)
    Input: $\mathcal{D} = \{(\mathbf{x}_k, \mathbf{y}_k)\}_{k=1}^{n}$, learning rate $\eta$
    Randomly initialize all weights and threshold

    repeat
        for all $(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) \in \mathcal{D}$ do
            Compute $y_j^{(i)}$ according current parameter
            compute $\delta_{\beta_i}$
            Compute $\delta_{\alpha_j}$
            update $w_{ji}$, $v_{kj}$
        end for
    until achieve stopping condition
end procedure
```

d) Validation of the integrated model

Test dataset is fed as input to the integrated model. Each model in the integrated system predicts the output. Majority poll of the above predictions is considered as the final prediction of individual test samples and accuracy scores are computed. Then proposed system has been rigorously validated on five different test data sets. For all the sets it is observed that accuracy scores are matching approximately to the maximum extent.

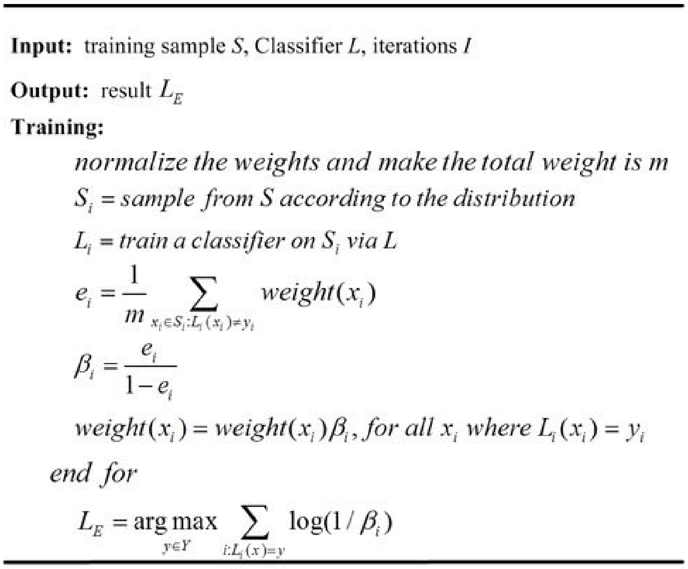
Bagging and AdaBoost

Unlike statistical voting theory which is based on the assumption of independent data sources and uses all training samples only one time, Boosting and Bagging are exerted by manipulating training samples. Bagging is the abbreviation of bootstrap aggregating. In this algorithm, n samples are selected at random from a training set with k samples, and instructive iteration is exerted to create some different bags, and each bag is classified by vote to predict its class. Boosting can process data with weights, and the weights of misclassified samples are increased to concentrate the learning algorithm on specific samples. Bagging has been shown to reduce the variance of the classification, while Boosting reduces both the variance and the bias of the classification. So in most cases, Boosting can produce more accurate classification results than Bagging. However, the computation time of Boosting is more than Bagging, and Boosting is sensitive to noise. AdaBoost is the popular used approach of Boosting algorithms. The pseudocode of Bagging and AdaBoost can be seen in below algorithm.



```
Input: training sample $S$, Classifier $L$, iterations $I$
Output: result $L_E$
Training:
  $for\ i = 1\ to\ I$
      $S_i = bootstrap\ sample\ from\ S$
      $L_i = train\ a\ classifier\ on\ S_i\ via\ L$
  $end\ for$
  $L_E = \arg\max_{y \in Y} \sum_{i:L_i(x)=y} 1$
```

(a) Bagging



```
Input: training sample $S$, Classifier $L$, iterations $I$
Output: result $L_E$
Training:
      $normalize\ the\ weights\ and\ make\ the\ total\ weight\ is\ m$
      $S_i = sample\ from\ S\ according\ to\ the\ distribution$
      $L_i = train\ a\ classifier\ on\ S_i\ via\ L$
      $e_i = \frac{1}{m} \sum_{x_i \in S_i : L_i(x_i) \neq y_i} weight(x_i)$
      $\beta_i = \frac{e_i}{1-e_i}$
      $weight(x_i) = weight(x_i)\beta_i, for\ all\ x_i\ where\ L_i(x_i) = y_i$
    $end\ for$
      $L_E = \arg\max_{y \in Y} \sum_{i:L_i(x)=y} \log(1/\beta_i)$
```

(b) AdaBoost

$$Final\_prediction = Max\{KNNb(t), NNb(t), DTb(t)\} \text{........(5)}$$

Where, KNNb is bagging model of KNN,
NNb is bagging model of Neural Networks,
DTb is bagging model of Decision Trees, t
is the test tuple

$$Final\_prediction = Max\{KNNa(t), NNa(t), DTa(t)\} \text{........(6)}$$

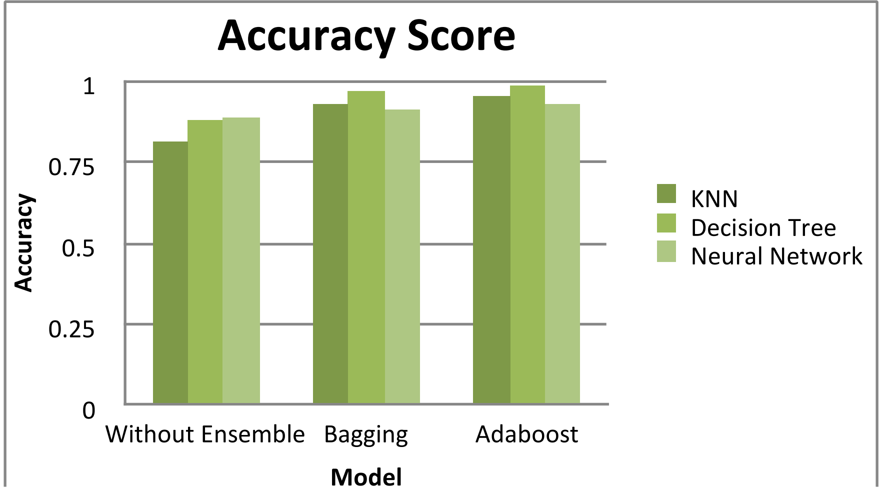
Where, KNNa is Adaboost model of KNN,
NNa is Adaboost model of Neural Networks,
DTa is Adaboost model of Decision Trees, t
is the test tuple

IV. PERFORMANCE ANALYSIS

The accuracy score is considered as the performance metric and the observed values are tabulated in table 1. The recorded accuracy scores for each of the algorithms with bagging, Adaboost and without bagging depicts that bagging and Adaboost technique enhances the performance of the individual models with the accuracy decision tree readings bagging (0.972) and Adboost(0.982), regarding kNN bagging(0.932) and Adaboost (0.951), Neural Network bagging (0.912) and Adaboost(0.931). The integrated model accuracy scores to 0.983, which is better than the individual algorithmic scores with and without ensemble method in chart.

TABLE1 Accuracy Scores

Machine Learning	Without Ensemble	Bagging	Adaboost
KNN	0.813	0.932	0.951
Decision Tree	0.882	0.973	0.982
Neural Network	0.922	0.912	0.931
Integrated Model	0.983		



V. CONCLUSION

In this paper, lung cancer as one of the most spreading cancer type was studied. Dataset from SEER program for lung cancer were analyzed survival prediction. SEER data originally with 149 attributes and 1000 samples after initial preprocessing, reduced to 24 attributes. Bagging and

Adaboost ensemble methods with three base learners(K-Nearest Neighbour, Decision Tree and Neural Network) were evaluated for lung cancer survival prediction. AdaBoost accuracy score is high comparatively with bagging. It is observed that bootstrap aggregating technique enhances the performance of the individual models with the accuracy scores decision tree readings Adboost(0.982), kNN and Adaboost (0.951), Neural Network bagging Adaboost(0.931). The obtained results from this study can be applied to increase performance of real patient survival prediction systems in future.

Reference

1. A. Agrawal, S. Misra, R. Narayanan, L. Polepeddi, and A. Choudhary, "Lung cancer survival prediction using ensemble data mining on SEER data," Sci. Program, vol. 20, no. 1, pp. 29–42, 2012.

2. H. M. Zolbanin, D. Delen, and A. Hassan Zadeh, "Predicting overall survivability in comorbidity of cancers: A data mining approach," Decis. Support Syst., vol. 74, pp. 150–161, 2015.

3. T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "Comparing boosting and bagging techniques with noisy and imbalanced data," IEEE Trans. Syst. Man, Cybern. Part ASystems Humans, vol. 41, no. 3, pp. 552–568, 2011.

4. D. Delen, G. Walker, and A. Kadam, "Predicting breast cancer survivability : a comparison of three data mining methods," Artif. Intell. Med., vol. 34 No.2, pp.113-27, 2005.

5. B. Zheng, S. W. Yoon, and S. S. Lam, "Expert Systems with Applications Breast cancer diagnosis based on feature extraction using a hybrid of K-means and support vector machine algorithms," Expert Syst. Appl., 2013.
6. DendiGayathri Reddy, Emmidi Naga Hemanth Kumar, Desireddy Lohith Sai Charan Reddy and Monika, " Integrated Machine Learning Model for Prediction of Lung Cancer Stages from Textual data Using Ensemble Method ",Proceedings of ICAIT, 2019.

7. JaneeAlam, Sabrina Alam and Alamgir Hossan, " Multi-Stage Lung Cancer Detection and Prediction Using Multi-Class SVM Classifier", Proceedings of IEEE.

8. "Lung Cancer Statistics," Centers for Disease Control and Prevention, Available: http:// www.cdc.gov/cancer/lung. [Accessed: 18-Jun-2016].

9. R. LAG, J. L. Young, G. E. Keel, M. P. Eisner, Y. D. Lin, and M. J. Horner, "SEER Survival Monograph, Cancer Survival Among Adults: US SEER Program, 1988-2001, Patient and Tumor Characteristics," National Cancer Institute NIH Pub., 2007.

10. National Cancer Institute, Surveillance, epidemiology and end results (seer) program (www.seer.cancer.gov) limited-use data (1973–2006). DCCPS, Surveillance Research Program, Cancer Statistics Branch, 2008. released April 2009, based on the November 2008 submission.

11. Overview of the seer program, Surveillance Epidemiology and End Results, available at: http://seer.cancer.gov/about/, accessed: April 29, 2010.

12. L.A.G. Ries, M.E. Reichman, D.R. Lewis, B.F. Hankey and B.K. Edwards, Cancer survival and incidence from the surveillance, epidemiology, and end results (SEER) program, *Oncologist* **8**(6) (2003), 541–552.