Comparative Study of Lung Disease Prediction System Using Top 10 Data Mining Algorithms with Real Clinical Medical Records

I Ketut Agung Enriko[1](✉), Teuku Muda Mahuzza[1], Sevia Indah Purnama[1], and Dadang Gunawan[2]

[1] Institut Teknologi Telkom Purwokerto, Purwokerto, Indonesia
{enriko,sevia}@ittelkom-pwt.ac.id
[2] University of Indonesia, West Java, Indonesia
guna@eng.ui.ac.id

Abstract. These years, the use of machine learning for disease prediction is blooming. Meanwhile, lung disease is one of the deadliest diseases in the world. Many researchers have been doing research on lung disease predictions using various techniques. In this study, ten machine learning algorithms are used for comparative study in lung disease prediction. The dataset is collected from a hospital in Banda Aceh, Indonesia, consisting of 300 data. The parameters included in the dataset are: symptoms, body temperature, respiration rate, oxygen saturation, blood pressure, heart rate, sex, and age. This dataset needs to be pre-processed and then analyzed using those top 10 machine learning algorithms. The prediction will be whether a patient gets a lung disease or not (binary prediction). The result shows that Naïve Bayes and k-Nearest Neighbor are the best choices among those algorithms in terms of accuracy and speed.

Keywords: machine learning · lung disease prediction · binary prediction · Naïve Bayes · k-Nearest Neighbor First Section

1 Introduction

Lung disease is one of the leading causes of deaths and disabilities. World Health Organization (WHO) reported that types of lung diseases like chronic obstructive pulmonary disease (COPD), pneumonia, tuberculosis (TB), and lung cancer cause more than 10 million deaths every year [1]. Meanwhile, Indonesia is also a country that have a large number of lung disease cases and it is worsen by the fact that the number of doctors is too few. From World Bank report, [2] the doctor/population ratio is 0.4/1000 which is the 129th rank in the world. These conditions are the main reasons that biomedical technology is blooming and widely adopted in Indonesia these days.

There are many research that have been done related to lung disease prediction. Das et al. discussed that artificial intelligence (AI) can provide a robust system to predict lung disease through the data analytics from pulmonary function tests (PFT), computed tomography (CT), forced oscillation tests (FOT), breath analysis, and lung sound analysis

I G. P. Suta Wijaya et al. (Eds.): MIMSE-I-C 2022, ACSR 102, pp. 269–281, 2022.
https://doi.org/10.2991/978-94-6463-084-8_24

[3]. Another study is a lung disease risk stratification using Riesz and Gabor transformation, using two stages process: lung delineation system and lung tissue characterization [4]. Meanwhile, [5] studied about how machine learning can help in COVID-19 disease from radiography images with k-Nearest Neighbor (kNN) algorithm. Another study is from [6] which wrote about how AI and machine learning can help to predict respiratory disease with three parameters: Thoracic imaging, histopathology or cytology, and physiological measurements and bio signals. Also, [7] have done the research to predict asthma and COPD with some machine learning algorithms which found that Random Forest gave the best result of accuracy.

This research discusses how lung disease can be predicted with machine learning algorithms. Ten popular algorithms are selected to examine the purpose. The dataset used for this research is real medical records taken from Cut Meuthia Hospital in Banda Aceh city, Indonesia. It consists of 300 records and will be analyzed in terms of accuracy and speed. The output of the prediction is a binary prediction which means whether a patient has a lung disease (or coded with "1") or not ("0").

This study is arranged as follows: Section I is for the introduction, Section II is literature review, Section III elaborates the methodology, Section IV explains the result and analysis, and Section V is for the conclusion.

2 Literature Review

Top 10 machine learning algorithms [8–10] are selected for comparison purpose in this research.

2.1 Naïve Bayes

Naïve Bayes is a simple machine learning algorithm which used in many research, mostly in classification cases. It is popular algorithm with high accuracy and speed, [11] and work well with natural language processing (NLP) use cases as well. An example of the application of the Naïve Bayes classification algorithm is the classification of new student admissions by applying and statistical approach, which gives predictions over existing data or phenomenon.

Naïve Bayes prediction is written in the formula [11]

$$\mathcal{P}(\mathrm{C_k}|x) = \frac{\mathcal{P}(C_k)\mathcal{P}(x|C_k)}{\mathcal{P}(x)} \tag{1}$$

where:

p(Ck|x)	Probability of hypothesis Ck based on condition x (posteriori probability)
p(Ck)	Hypothesis probability Ck (prior probability)
p(x|Ck)	Probability of x based on the conditions on the hypothesis Ck
p(x)	Probability x.

2.2 Logistic Regression

Logistic Regression is a regression analysis that is performed when the variables have two possibilities [12]. The logistic regression model is a statistical model used to determine the effect of the predictor variable (X) on the response variable (Y) with the response variable being dichotomous data with a value of 1 which means that the response variable has the specified criteria and 0 indicates that the response variable does not have the specified criteria. An example of the application of logistic regression can predict statistical students pass the exam based on the duration of students' study in a day.

The formula of logistic regression is written as follows [12]:

$$\mathcal{P}(x) = \frac{1}{1 + e^{-(x-\mu)/s}} \tag{2}$$

where:

μ is a location parameter (where $p(\mu) = 1/2$)
s is a scale parameter.

2.3 K-Nearest Neighbor

THE k-Nearest Neighbor (kNN) algorithm is a classification method in machine learning, where kNN classifies a set of data based on learning data that has been classified or labeled. kNN is one of supervised algorithms, where to examine the classification of a datum (testing) is determined by the closest distance to the existing data (training) in kNN [13].

kNN has the principle of comparing testing data (new data) with training data (old data) one by one. kNN has several advantages, number one is its toughness against training data which have a lot of noise. The downs ide of kNN is it needs to determine the value of training parameters based on distance. The formula of kNN is written in the equation below [14]:

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{3}$$

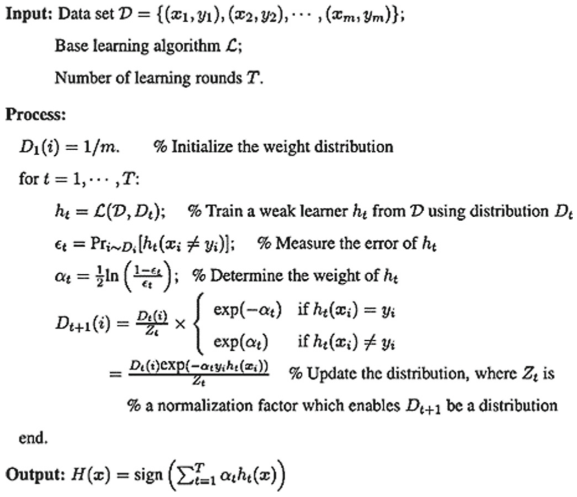
where:

d is the Euclidean distance
x is the coordinate of training data
y is the coordinate of test data.

2.4 K-Star

K-star is an example-based classifier, where the test sample classes are based on their class examples from similar exercises, as determined by some equation function [15].

K-star has the concept of entropy to define a distance metric calculated by means. K-star works by adding up the probabilities of the instances of all members of a category.



```
Input: Data set $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m)\}$;
       Base learning algorithm $\mathcal{L}$;
       Number of learning rounds $T$.
Process:
  $D_1(i) = 1/m$.   % Initialize the weight distribution
  for $t = 1, \cdots, T$:
      $h_t = \mathcal{L}(\mathcal{D}, D_t)$;   % Train a weak learner $h_t$ from $\mathcal{D}$ using distribution $D_t$
      $\epsilon_t = \Pr_{i \sim D_t}[h_t(x_i \neq y_i)]$;   % Measure the error of $h_t$
      $\alpha_t = \frac{1}{2}\ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$;   % Determine the weight of $h_t$
      $D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} \exp(-\alpha_t) & \text{if } h_t(x_i) = y_i \\ \exp(\alpha_t) & \text{if } h_t(x_i) \neq y_i \end{cases}$
                $= \frac{D_t(i)\exp(-\alpha_t y_i h_t(x_i))}{Z_t}$   % Update the distribution, where $Z_t$ is
                    % a normalization factor which enables $D_{t+1}$ be a distribution
  end.
Output: $H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$
```

Fig. 1. AdaBoost pseudo-code

This has to do with the rest of the categories to select the best possibilities, and to deal with missing values in the data set assuming that the probability of transforming it to a value type, is the average of the probabilities. The formula of K-star is written in the equation below [16]:

$$K * (b|a) = -\log_2 P * (b|a) \tag{4}$$

where P is a probability function defined as probability of all paths from instance *a* to instance *b*.

2.5 AdaBoost

Adaptive Boosting algorithm or commonly called AdaBoost is one of the algorithms used for decision making. It is proposed by Freund and Schapire in 1996. AdaBoost is an ensemble method, combining multiple classifiers to improve the classifier accuracy. Its basic principle is a set or works: setting classifier weights (and combining multiple weak classifiers), training some data sample, and iteratively observing the results to ensure the prediction accuracy [17]. Figure 1 describes the pseudo-code of AdaBoost [18].

2.6 Bagging Algorithm

Bagging algorithm is a method for reducing noise data in neural networks, and it is well applied to compressive strength data sets which have numeric attributes and labels. Bagging stands for bootstrap aggregating, it uses a sub-dataset (bootstrap) to generate a training set L (learning), L trains the learning base using an unstable learning procedure, and then, during testing, takes the average. Bagging is good for classification and

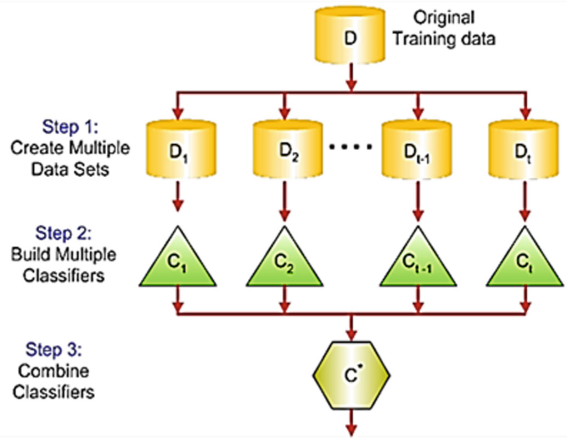
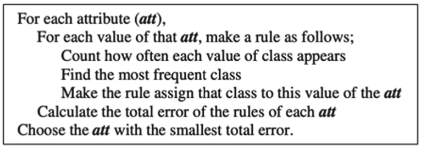

D
Original Training data
Step 1: Create Multiple Data Sets
D_1
D_2
D_{t-1}
D_t
Step 2: Build Multiple Classifiers
C_1
C_2
C_{t-1}
C_t
Step 3: Combine Classifiers
C^*

Fig. 2. Bagging algorithm combines multiple models



```
For each attribute (att),
    For each value of that att, make a rule as follows;
        Count how often each value of class appears
        Find the most frequent class
        Make the rule assign that class to this value of the att
    Calculate the total error of the rules of each att
Choose the att with the smallest total error.
```

Fig. 3. OneR pseudo-code [20]

regression. In the case of regression, to be more robust, one can take the average when combining the predictions.

Bagging is a learning algorithm that is stable at small changes in the training set causing large differences in the resulting learner, that is, the algorithm learns on data that has high variance (noise). Bagging was able to improve accuracy significantly greater than the individual model, and was stronger against noise and overfitting effects than the original training data [18]. Figure 2 depicts the idea of how Bagging algorithm works.

2.7 OneR Algorithm

The OneR algorithm stands for One Rule. The OneR algorithm will generate a rule for each attribute then select the rule with the smallest error which is then used as one rule. To create a rule for each existing attribute, it is necessary to create an occurrence table for each attribute with the target [19].

OneR is a simple classification machine learning that builds a single-level decision tree. An example of the application of OneR to detect liver disease by using one rule for each attribute in the training data then selecting the rule with the smallest error (one rule). The pseudo-code of OneR algorithm is shown in Fig. 3.

2.8 Random Forest

Random Forest is one of the methods used for regression and classification. It is the advancement of the learning method using decision trees as a base classifier that is built and combined. In Random Forest method there are three important aspects, which are: (1) bootstrapping to build a prediction tree, (2) decision trees is constructed for predictors at random, and (3) make predictions by combining the decisions from each tree by majority vote or average for classification or regression respectively [21].

An example of applying the random forest algorithm for rainfall prediction with classification-based and regression-based methods where there is a decision tree aggregation process. This method was chosen because it produces lower errors and provides good accuracy and is effective for dealing with incomplete data.

The Random Forest formula is shown in Eq. (5) below [22]

$$A(x) = \operatorname{argmax}\left\{\sum_{i=1}^{z} Q(A(B, \theta_k) = J)\right\} \tag{5}$$

where:

A(x)	is the Random Forest model
J	represents target category variable
Q	is the characteristic function
arg $\max_J$	is the value when the function in the curly brackets gives maximum value.

2.9 Support Vector Machine (SVM)

SVM is a machine that uses vectors as supports or markers to divide data into two groups [23]. In other words, SVM is a technique that uses two points (two vectors), where these two points will form a dividing line (or a border in three dimensions or more). The boundary line or side that is formed from these two vectors is called a hyperplane. An illustration of a hyperplane is shown in Fig. 4.

It can be seen that there are two groups of data or classification. The task of SVM is to divide these two groups as best as possible. There should be two points that become the benchmark for the hyperplane, called support vectors. It should be noted that these

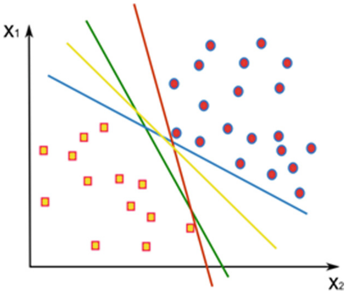


Fig. 4. A 2-D Hyperplane [23]

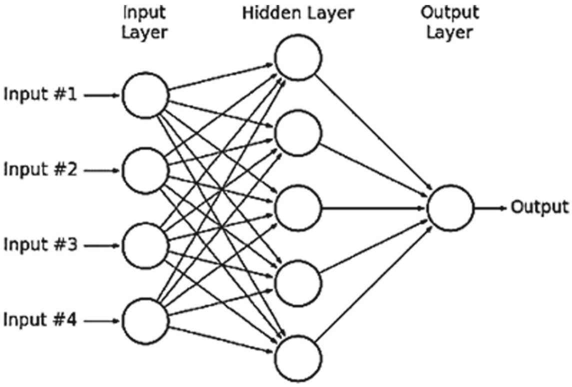


Fig. 5. Multi-Layer Perceptron [24]

support vectors are two outermost points (outermost of the group and closest to the hyperplane) that must be perpendicular to the hyperplane. Meanwhile, the other points behind these two lines do not contribute anything to the SVM results, as the SVM results are good depends on the support vectors.

2.10 Multi-layer Perceptron (MLP)

MLP is a popular algorithm from neural network (NN) family. It is a robust algorithm, works as a supervised training method, and uses data samples with known outputs [24]. MLP has neurons connected to connecting nodes, arranged with at least three layers: input layer, hidden layer(s), and output layer. Meanwhile, the newer NN algorithms like CNN and RNN emerge and outperforms MLP in terms of accuracy, but they are not available in standard WEKA software. Figure 5 shows how MLP works with multiple layers.

3 Methodology

3.1 Dataset

The dataset used in this research have been collected from Cut Meuthia Hospital (CMH), Banda Aceh, Indonesia. The period of collection is around November 2021, collecting 300 medical records as the result. The parameters of the dataset are shown in Table 1.

The first step to analyze the data is to preprocess the data, which means text formatted data should be converted to other format (binary, integer, or specific code). Referring to the pulmonary specialist doctor interview in this study, there are four most important symptoms in diagnosing lung disease: cough, phlegm cough, bleeding cough, and shortness of breath (doctors usually also check the chest X-Ray test result, but it is not in the scope of this research). These four symptoms should be separated and converted to binary formats so they can be analyzed: Cough (Yes/No), Phlegm Cough (Yes/No), Bleeding Cough (Yes/No), and Shortness of Breath (Yes/No). The diagnosis parameter

Table 1. Dataset parameters and formats

No.	Parameter	Format
1	Symptom	Text
2	Temperature (T)	Numeric
3	Respiration Rate (RR)	Integer
4	Oxygen Saturatioin (SpO_2)	Integer
5	Systolic (Sys)	Integer
6	Diastolic (Dia)	Integer
7	Heart Rate (HR)	Integer
8	Sex (S)	Binary (Male/Female)
9	Age (A)	Integer
10	Diagnosis (D)	Text

Table 2. Pre-processed dataset

No.	Parameter	Format
1	Cough (C)	Binary (Yes/No)
2	Phlegm Cough (PC)	Binary (Yes/No)
3	Bleeding Cough (BC)	Binary (Yes/No)
4	Shortness of Breath (SB)	Binary (Yes/No)
5	Temperature (T)	Numeric
6	Respiration Rate (RR)	Integer
7	Oxygen Saturation (SpO_2)	Integer
8	Systolic (Sys)	Integer
9	Diastolic (Dia)	Integer
10	Heart Rate (HR)	Integer
11	Sex (S)	Binary (Male/Female)
12	Age (A)	Integer
13	Diagnosis (D)	Binary (Yes/No)

should also be converted, from text to binary: Yes (the patient suffers lung disease) or No (the patient does not suffer lung disease). Meanwhile, the other parameters' formats can be left as it is. The pre-processed dataset parameters and formats are described in Table 2.

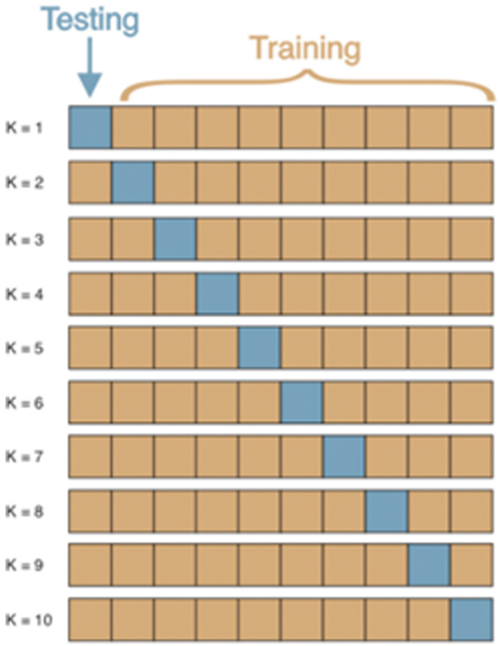


Fig. 6. 10-fold-cross-validation [26]

3.2 Machine Learning Analysis Tool

After pre-processing the dataset, then the main processing should be done. In this study we use WEKA machine learning tool. WEKA is a prevalent machine learning tool widely used by data scientists [25]. These 10 machine learning algorithms will be tested using WEKA with the dataset collected and pre-processed. The dataset will be divided to training and testing data with 10-fold-cross-validation rule. It means that 90% of the data will be used as training and 10% for testing data, and this mechanism will be looped until 10 times so all the data are used. The 10-fold-cross-validation method is described in Fig. 6.

Meanwhile, other WEKA classifier options are left with default settings. Table 3 shows classifier options of WEKA to do the analysis of the ten algorithms.

4 Result and Discussion

4.1 Prediction Accuracy

WEKA tool has been used for machine learning analysis in this study and the results is shown in Table 4.

From the table, it can be inferred that Naïve Bayes is the most accurate algorithm for the dataset to predict whether patients get lung disease or not with 78.67% accuracy. Followed by other algorithms consecutively: KNN (77.67%), SVM (75%), AdaBoost (74.67%), MLP (74.33%), Random Forest (72.67%), Logistic Regression 72.33%), K-Star (71.67%), Bagging (70%), and OneR (68.67%).

Table 3. WEKA Classifier Settings

No.	Algorithm	Classifier options

1	OneR	batchSize = 100
		minBucketSize = 6
		numDecimalPlaces = 2

2	SVM (SMO)	SVMType = C-SVC
		Degree = 3
		Gamma = 0
		Epsilon = 0.001
		KernalType = radial basis function

3	AdaBoost	Classifier = DecisionStump
		numIterations = 0
		seed = 1
		weightThreshold = 100

4	KNN	k = 9
		distanceWeighting = no
		NNSearchAlgorithm = LinearNNSearch

5	Naïve-Bayes	Cross-validation folds = 10

6	K-Star	batchSize = 100
		globalBlend = 20
		missingMode = Avergae column entropy curves
		numDecimalPlaces = 2

7	Random forest	numTrees = 100
		maxDepth = 0
		seed = 1

8	Bagging	bagSizePercent = 100
		classifier = REPTree
		numIterations = 10
		seed = 1

9	Logistic regression	maxIts = 1
		ridge = 1.0E-8

10	MLP	learningRate = 0.3
		momentum = 0.2
		seed = 0
		tariningTime = 500

4.2 Algorithm Speed

Besides accuracy, one important measurement of the algorithm performance is speed. This study uses a personal computer with Apple M1 CPU and 8 GB RAM to do the simulation. Table 5 shows the speed (refers to the time taken to build models) of each algorithm‘s to build models in WEKA simulation.

It can be inferred from the table that KNN, K-Star, and OneR are the fastest algorithms with 0 s (too fast to be measured by WEKA) time taken to build the model. Followed by

Table 4. Prediction Accuracy

Rank	Algorithm	Accuracy (%)
1	Naïve Bayes	78.67
2	KNN	77.67
3	SVM	75
4	AdaBoost	74.67
5	MLP	74.33
6	Random Forest	72.67
7	Logistic Regression	72.33
8	K-Star	71.67
9	Bagging	70
10	OneR	68.67

Table 5. Algorithm speed

Rank	Algorithm	Time to Build Model (s)
1	KNN	0 (too fast to be measured)
2	K-Star	0 (too fast to be measured)
3	OneR	0 (too fast to be measured)
4	Logistic Regression	0.01
5	Naïve Bayes	0.01
6	AdaBoost	0.04
7	Bagging	0.06
8	SVM	0.07
9	Random Forest	0.13
10	MLP	2.75

other algorithms consecutively: Logistic Regression and Naïve Bayes (0.01 s), AdaBoost (0.04 s), Bagging (0.06 s), SVM (0,07 s), Random Forest (0.13 s), and the last is MLP (2.75 s).

4.3 Discussion

From the WEKA simulations using CMH data, Naïve Bayes algorithm is superior with its best accuracy (78.67%) and and the speed (0.1 s). KNN also performs well with second best accuracy (77.67%) and powerful speed (0 s or almost instantaneous). SVM provides third best accuracy (75%) but the speed is third worst (0.07 s). AdaBoost gives good accuracy (74.67%) and speed (0.04 s) in average. Then, MLP has a good accuracy

(74.33%) but the speed is the worst (2.75 s) while Random Forest has a similar characteristic with 72.67% accuracy and quite slow speed (0.13 s). The next three algorithms have fair performances: Logistic Regression (72.33% accuracy and 0.01 s speed), K-Star (71.67 accuracy and 0 s speed), and Bagging (70% accuracy and 0.06 s speed). The last is OneR which has the worst accuracy (68.67%) but has an excellent speed (0 s).

5 Conclusion

Machine learning algorithms have been used for many purposes. This study's goal is to predict whether a patient gets lung disease or not, with top ten machine learning algorithms (KNN, K-Star, One-R, Logistic Regression, Naïve Bayes, AdaBoost, Bagging Algorithm, SVM, Random Forest, and MLP). Dataset used in this research was collected from Cut Meuthia Hospital, city of Banda Aceh, Indonesia. There are 300 patient medical records with 13 parameters, analyzed with WEKA software.

The WEKA simulation result shows that Naïve Bayes, KNN, and SVM provide the best accuracy with 78.67%, 77.67%, and 75% correct predictions respectively. Meanwhile from the speed aspect, KNN, K-Star, and OneR are the fastest algorithms with nearly zero seconds needed to build the model.

Acknowledgment. This research is funded by Indonesia Telecommunication and Digital Research Institute (ITDRI), PT Telkom Indonesia.

References

1. S. M. Levine and D. D. Marciniuk.: Global Impact of Respiratory Disease. (2022).
2. W. Bank. Physicians (Per 1,000 People). World Bank Report, https://data.worldbank.org/indicator/SH.MED.PHYS.ZS?most_recent_value_desc=true, last accessed 2021/02/15.
3. N. Das, M. Topalovic, and W. Janssens.: Artificial intelligence in diagnosis of obstructive lung disease: Current status and future potential. Current Opinion in Pulmonary Medicine 24(2), 117–123 (2018).
4. J. C. M. Than et al.: lung disease stratification using amalgamation of Riesz and Gabor trans-forms in machine learning framework. Comput. Biol. Med. 89, 197–211 (2017).
5. L. Brunese, F. Martinelli, F. Mercaldo, and A. Santone.: Machine learning for coronavirus covid-19 detection from chest x-rays. Procedia Comput. Sci. 176, 2212–2221 (2020).
6. S. Gonem, W. Janssens, N. Das, and M. Topalovic.: Applications of artificial intelligence and machine learning in respiratory medicine. Thorax 75(8), 695–701 (2020).
7. D. Spathis and P. Vlamos.: Diagnosing asthma and chronic obstructive pulmonary disease with machine learning. Health Informatics J. 25(3), 811–827 (2019).
8. X. Wu et al.: Top 10 Algorithms in Data Mining. Knowledge and Information Systems. Knowl. Inf. Syst. 14(1), 1–37 (2008).
9. W. C. Lin, S. W. Ke, and C. F. Tsai.: Top 10 data mining techniques in business applications: a brief survey. Kybernetes 46(7), 1158–1170 (2017).
10. I. K. A. Enriko, M. Suryanegara, and D. Gunawan.: Comparative Study of Heart Disease Diagnosis Using Top Ten Data Mining Classification Algorithms. J. Telecommun. Electron. Comput. Eng, (2019).

11. N. Y. Septian.: Data Mining Menggunakan Algoritma Naïve Bayes Untuk Klasifikasi Kelulusan Mahasiswa Universitas Dian Nuswantoro. J. Semant, 1–11 (2009).
12. D. Bertsimas and A. King.: Logistic regression: From art to science. Stat. Sci. 32(3), 367–384 (2017).
13. D. Enriko, I. K. A., Suryanegara, M., & Gunawan.: Heart Disease Prediction System using k-Nearest Neighbor Algorithm with Simplified Patient's Health Parameters. J. Telecommun. Electron. Comput. Eng. 8(12), 59–65 (2016).
14. I. K. A. Enriko, M. Suryanegara, and D. Gunawan.: heart disease diagnosis system with k-nearest neighbors method using real clinical medical records. (2018).
15. S. Ravikumar, H. Kanagasabapathy, and V. Muralidharan.: Fault diagnosis of self-aligning troughing rollers in belt conveyor system using k-star algorithm. Meas. J. Int. Meas. Confed. 133, 341–349 (2019).
16. R. Satishkumar and V. Sugumaran.: Remaining life time prediction of bearings using K-star algorithm – a statistical approach. J. Eng. Sci. Technol. 12(1), 168–181 (2017).
17. C. Tu, H. Liu, and B. Xu.: AdaBoost typical Algorithm and its application research. MATEC Web Conf. 139 (2017).
18. X. Wu et al.: Top 10 algorithms in data mining. Knowl. Inf. Syst. (2008).
19. F. Nidaul Khasanah.: Klasifikasi Proses Penjurusan Siswa Tingkat SMA Menggunakan Data Mining. Informatics Educ. Prof. 1(1), 65–69 (2016).
20. M. Jamjoom.: The pertinent single-attribute-based classifier for small datasets classification. Int. J. Electr. Comput. Eng. 10(3), 3227–3234 (2020).
21. S. Mohan, C. Thirumalai, and G. Srivastava.: Effective heart disease prediction using hybrid machine learning techniques. IEEE Access 7 (2019).
22. X. Li, Z. Wang, L. Wang, R. Hu, and Q. Zhu.: A multi-dimensional context-aware recommendation approach based on improved random forest algorithm. IEEE Access 6, 45071–45085 (2018).
23. J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez.: A comprehensive survey on support vector machine classification: Applications, challenges and trends. Neurocomputing (2020)
24. H. Taud and J. F. Mas.: Multilayer Perceptron (MLP). 451–455 (2018).
25. I. H. W., Eibe Frank, Mark A. Hall.: The WEKA Workbench. Online Appendix for Data Mining: Practical Machine Learning Tools and Techniques. (2016).
26. C. Neale.: Cross Validation: A Beginner's Guide. Towards Data Science Tutorial, https://towardsdatascience.com/cross-validation-a-beginners-guide-5b8ca04962cd, last accessed 2022/03/30.