Optimization of C4.5 algorithm using meta learning in diagnosing of chronic kidney diseases

To cite this article: A Nurzahputra et al 2019 J. Phys.: Conf. Ser. 1321 032022

View the article online for updates and enhancements.
Optimization of C4.5 algorithm using meta learning in diagnosing of chronic kidney diseases

A Nurzahputra*, M A Muslim and B Prasetiyo

Computer Sciences Department, Universitas Negeri Semarang, Indonesia

*Corresponding author: aldinurzah96@gmail.com

Abstract. The accuracy in diagnosing a disease is very important and crucial for the hospital institution. The big patient data can be processed into information to diagnose a disease. One of diseases can be diagnosed is chronic kidney disease. The existing data of chronic kidney disease patients can be used for data processing. Data processing is included in the process of data mining. One method that can be implemented to disease diagnosis, namely classification. The algorithm used in this research is decision tree namely C4.5 algorithm. It can be applied in classifying patient data. This study used Chronic Kidney Disease (CKD) dataset. The purpose of this study is optimizing the accuracy of C4.5 algorithm by applying meta learning in diagnosing CKD by comparing the results before and after MultiboostAB and Bagging applied. The validation used 10 fold cross validation. The accuracy is measured by confusion matrix. The combination of C4.5 and MultiboostAB obtained 99.5% accuracy which increased 0.5% from accuracy of C4.5 standalone. Then, C4.5 and Bagging obtained 100% accuracy. The result of this research is the application of bagging on C4.5 algorithm is good in optimizing accuracy.

1. Introduction

The development of big data processing is increasing rapidly. It has also been applied in various fields, one of them is in the health fields. In the health, the diagnosing a disease symptom can be conducted by processing the existing patient data [1]. Its data processing can help doctor to take decisions and to avoid complications. Therefore, the level of accuracy in diagnosing diseases is very important and crucial for hospital agencies. The processing data of patient is included in the data mining. Data mining has techniques and algorithm which able to extract information from big data automatically [2]. Data mining techniques which can be applied in predicting or diagnosing a disease is classifications [3].

Classification is a technique used to find a model or pattern by describing and distinguishing data classes or concepts. In classification there are several algorithms, one of them is the decision tree. Decision tree works by building a tree model to classify data [4]. One of the best algorithms is C.45 algorithm. The C4.5 algorithm is the development of a conventional decision tree induction algorithm, ID3 [5].

The algorithm can be applied for helping doctors and medical experts in diagnosing a disease. One of the diseases that can be diagnosed is Chronic Kidney Disease (CKD). CKD is a physiological process of pathogens can cause various symptoms which can make significant decreasing in renal function (kidney glands) then it may have effect into kidney failure [6]. CKD is generally a disease that can be detected through creating serum level, ureum plasma level and glomerulus filtration rates
which can be tested in laboratory or through several treatments to prevent development and slow the progression of the disease [7].

In this research, the accuracy of algorithm is obtained can be increased by implementing the meta learning to the algorithm. Meta learning works to learn the patterns automatically in metadata to obtain better performance than existing algorithms. Meta learning can work properly depends on the task to be solved which is improving performance of algorithm. [8] In data mining there are several techniques which can be used. In this research used MultiboostAB and Bagging as meta learning which it will be combined to work together with C4.5 algorithm. MultiBoostAB is one of techniques in meta learning which extends the AdaBoost with wagging [9]. The wagging included to the kind of bagging which works generating the training process while boosting performed in selecting of the samples. Then, as meta learning, bagging can generate a bunch of set by conducting resample which giving to that set randomly [10]. The selection of performance is conducted using majority vote. Therefore, the aim of this study is improving the accuracy of C4.5 algorithm by implementing MultiboostAB and Bagging meta learning. In this research, it also uses Area Under Curve (AUC) value to measure the performances. It can show which meta learning has the optimal performance in diagnosing CKD. The validation used is 10 fold cross validation and accuracy measurement using confusion matrix.

2. Methods

Decision tree is a method that exists in classification techniques in data mining. It predicts by constructing the classification in a tree structured type. Then, it disintegrates in to smaller subsets also simultaneously develops the decision nodes and leaf nodes [11]. Decision trees uses for exploring data and finding hidden relationship from the number of data or variables [2]. It is why decision tree play an important role in machine learning [12]. The methodology used in this research consists of three stages. The first stage is literature study. It was conducted by collecting theories and literature related to C4.5 algorithm and meta-learning. The second stage is data collection. The third stage is data processing and testing by applying algorithm and also the meta learning. Then, the result of accuracy was compared to get the optimal meta learning which giving increased accuracy.

2.1. C4.5 Algorithm

There are some algorithms can be used to generate the decision tree, one of them is the C4.5 algorithm. The tree of C4.5 algorithm was generated based on the gain values of each features [5]. The tree itself consist in the top is the root, and the bottom of it is called the leaf. The advantage of this algorithm is effectively analysing a large number of attributes from existing data and easily understood by end users [13].

2.2. MultiBoost

MultiBoost is one of meta learning which compared to Boosting. But, multiboost is different than boosting. Multiboost enlarges oncoming of boosting using wagging. It is a kind of bagging was used to select the samples and it generating the weight of training instances. From the past experiments and researches of implementing multiboost to some algorithm on a big and diverse collection of datasets had obtained that Multiboost obtained higher accuracy and better performance [9].

2.3. Bagging

Bagging stands for "bootstrap aggregating" and it was discovered by Breiman. Based on Han et al. [4] states that bagging is one of the techniques of meta learning which manipulated the training data. Those data was duplicated as many as n times using the randomly sampling. Then it will generate as many as the new n data. Those new data entitled the bagged classifier [5]. Because the data on bagging techniques is sampling with replacement, the size of bagging data is the same as the original data, but the distribution of data from each bagging data is different, some data from training data may appear
several times or may not appear at all [1]. This is the key to why bagging can improve accuracy because with sampling replacement can reduce the variance of the datasets.

3. Result and Discussions
This research conducted by implementing the methods using data mining tool namely Weka. The datasets used Chronic Kidney Disease well known as CKD. The dataset taken from UCI repository of machine learning. The datasets consisted 400 data records, 24 features, and a feature of class. This class feature classified into two which is ckd and notckd. The number of ckd class was 250 data records and the number of notckd class was 150 data records. Ckd class means the patient who suffered by chronic kidney disease. Then, notckd means the patient who not suffered by chronic kidney disease.

ROC curve is a way which can be used to assess the performance of classifier for binary classification. For each object is mapped to an element of the set pair of positive and negative. ROC curve can be seen in Figure 1. Moreover, according to Gorunescu, from the ROC curve, the AUC value was classified into some category. The categories shown in Table 1.

![Figure 1. ROC curve](image)

### Table 1. AUC value

| AUC Value       | Category             |
|-----------------|----------------------|
| 0.90 – 1.00     | Excellent Classification |
| 0.80 – 0.90     | Good Classification  |
| 0.70 – 0.80     | Fair Classification  |
| 0.60 – 0.70     | Poor Classification  |
| 0.50 – 0.60     | Failure Classification |

3.1. C4.5 Algorithm
The implementation of C4.5 stand alone was conducted because it’s needed to know the accuracy without implementing MultiboostAB and Bagging. This research used 10 fold cross validation as the measurement. It meant the splitting of training data and testing data was conducted automatically using 10 fold. The result of using of C4.5 algorithm produced 9 leaves and the tree size was 14. The measurement of C4.5 presented in Table 2.

### Table 2. The measurement of C4.5 classification algorithm

| Measurement Specification       | C4.5 stand alone | C4.5 and MultiboostAB | C4.5 and Bagging |
|---------------------------------|------------------|-----------------------|------------------|
| Correctly Classified Instances  | 396 or 99 %      | 398 or 99.5 %         | 400 or 100 %     |
| Classified Incorrectly Instances| 4 or 1 %         | 2 or 0.5 %            | 0 or 0 %         |
| Kappa Statistic                 | 0.9786           | 0.9893                | 1                |
| Mean Absolute Error             | 0.0225           | 0.0188                | 0.037            |
| Root Mean Squared Error         | 0.0807           | 0.0833                | 0.0874           |
| Relative Absolute Error         | 4.7995%          | 4.0108%               | 7.889%           |
| Root Relatives Squared Error    | 16.6603%         | 17.2041%              | 18.0618%         |
| Total Number of Instances       | 400              | 400                   | 400              |

Based on Table 2, the accuracy of C4.5 stand alone obtained 99%. Correctly classified instances meant the number of instances which classified and match according the actual class of datasets. The number of instances was 396 data records. It meant that there were 4 data records which classified
incorrectly. Then, it could be presented in percentages as 99 % and 1 %. Kappa statistic was obtained 0.9786. Mean absolute error was obtained 0.0225. The root mean squared error was obtained 0.0807. Relatives absolute error was obtained 4.7995%. Then, the root of it was obtained 16.6603%. For the result of the detailed accuracy was shown in Table 3.

| TP Rate | FP Rate | Precision | Recall | F-Measure | ROC Area | Class |
|---------|---------|-----------|--------|-----------|----------|-------|
| 0.996   | 0.020   | 0.988     | 0.996  | 0.992     | 0.999    | Ckd   |
| 0.980   | 0.004   | 0.993     | 0.980  | 0.987     | 0.999    | Notckd|

Weight Avg. 0.990 0.014 0.990 0.990 0.990 0.999

Based on Table 3, in ckd class True Positive was obtained 0.996 then the notckd class obtained False Positive rate was 00.4. In notckd class True Positive was obtained 0.980 then the ckd class obtained False Positive rate was 0.20. The Precision, Recall, and F-Measure had the same Weight Average which was 0.990. The ROC area was obtained 0.999. Then, it generated the confusion matrix. The confusion matrix of this C4.5 algorithm implementation was shown in Table 4.

| Classification | ckd | notckd |
|----------------|-----|--------|
| ckd            | 249 | 1      |
| notckd         | 3   | 147    |

Based on Table 4, the confusion matrix gave the number of instances after classifying the algorithm. From 250 data records of ckd, the result shown that 249 data were correctly classified and 1 data was incorrectly classified as ckd. Then, from 150 data record of notckd, the result shown that 3 data were incorrectly classified and 147 data were correctly classified as not ckd. The AUC value of C4.5 stand alone obtained 0.9994.

3.2. C4.5 Algorithm and MultiBoostAB

The implementation of MultiBoostAB worked together with C4.5 algorithm. The result of using of C4.5 algorithm and MultiBoostAB produced 8 leaves and the tree size was 12. Based on Table 2, the accuracy of C4.5 algorithm combining MultiBoostAB obtained 99.5%. For the result of the detailed accuracy was shown in Table 5.

| TP Rate | FP Rate | Precision | Recall | F-Measure | ROC Area | Class |
|---------|---------|-----------|--------|-----------|----------|-------|
| 0.996   | 0.007   | 0.996     | 0.996  | 0.996     | 1        | Ckd   |
| 0.993   | 0.004   | 0.993     | 0.993  | 0.993     | 1        | Notckd|

Weight Avg. 0.995 0.006 0.995 0.995 0.995 1

Based on Table 5, in ckd class True Positive was obtained 0.996 then the notckd class obtained False Positive rate was 00.4. In notckd class True Positive was obtained 0.993 then the ckd class obtained False Positive rate was 0.007. The Precision, Recall, and F-Measure had the same Weight Average which was 0.995. The ROC area was obtained 1. Then, it generated the confusion matrix. The confusion matrix of this C4.5 algorithm combining MultiBoostAB was shown in Table 6.

| Classification | ckd | notckd |
|----------------|-----|--------|
| ckd            | 249 | 1      |
| notckd         | 1   | 149    |

Based on Table 6, the confusion matrix gave the number of instances after classifying the algorithm. From 250 data records of ckd, the result shown that 249 data were correctly classified and 1 data was incorrectly classified as ckd. Then, from 150 data record of notckd, the result shown that 3 data were incorrectly classified and 147 data were correctly classified as not ckd. The AUC value of C4.5 stand alone obtained 0.9994.
Based on Table 6, from 250 data records of ckd, the result shown that 249 data were correctly classified and 1 data was incorrectly classified as ckd. Then, from 150 data record of notckd, the result shown that 1 data were incorrectly classified and 149 data were correctly classified as not ckd. The AUC value of C4.5 algorithm and MultiBoostAB obtained 0.9996.

3.3. C4.5 Algorithm and Bagging

The implementation of Bagging also worked together with C4.5 algorithm. The result of using of C4.5 algorithm and Bagging produced 9 leaves and the tree size was 14. Based on Table 2, the accuracy of C4.5 algorithm combining Bagging obtained 100%. For the result of the detailed accuracy was shown in Table 7.

| TP Rate | FP Rate | Precision | Recall | F-Measure | ROC Area | Class  |
|---------|---------|-----------|--------|-----------|----------|--------|
| 1       | 0       | 1         | 1      | 1         | 1        | ckd    |
| 1       | 0       | 1         | 1      | 1         | 1        | notckd |

Weight Avg. 1 0 1 1 1

Based on Table 7, in ckd class True Positive was obtained 1 then the notckd class obtained False Positive rate was 0. In notckd class True Positive was obtained 1 then the ckd class obtained False Positive rate was 0. The Precision, Recall, and F-Measure had the same Weight Average which was 1. The ROC area was obtained 1. Then, it generated the confusion matrix. The confusion matrix of this C4.5 algorithm combining Bagging was shown in Table 8.

| Classification | ckd | notckd |
|----------------|-----|--------|
| ckd            | 250 | 0      |
| notckd         | 0   | 150    |

Based on Table 8, from 250 data records of ckd, the result shown that 250 data were correctly classified and no data was incorrectly classified as ckd. Then, from 150 data record of notckd, the result shown that no data were incorrectly classified and all 150 data were correctly classified as not ckd. It was way this accuracy obtained 100%. The AUC value of C4.5 algorithm and Bagging obtained 1.

3.4. The Comparison of Accuracy and AUC Value

Based on the result, the accuracy of C4.5 algorithm was increased by implementing MultiBoostAB and Bagging. It meant that the meta learning could increase the accuracy of algorithm. The AUC value was also increased. From the result, the best accuracy was obtained by combining C4.5 algorithm and Bagging. The comparison of the result shown in Table 9.

| Algorithm                  | Accuracy (%) | AUC Values |
|----------------------------|--------------|------------|
| C4.5 Algorithm stand alone | 99%          | 0.9994     |
| C4.5 Algorithm and MultiBoostAB | 99.5%       | 0.9996     |
| C4.5 Algorithm and Bagging | 100%         | 1          |

4. Conclusion

The result shows that the implementing of meta learning could optimize the performance and increase the accuracy. The accuracy of MultiBoostAB and C4.5 algorithm is obtained 99.5%. It is increasing 0.5% compared to C4.5 algorithm stands alone accuracy. Then, the accuracy of Bagging and C4.5 algorithm is obtained 100%. It is increasing 1% compared to C4.5 algorithm stands alone accuracy.
So, it can be concluded that the optimal meta learning implemented with C4.5 algorithm was Bagging and meta learning can optimize the C4.5 algorithm.

References
[1] Muslim M A, Anis J H, Endang S and Budi P 2018 J. Phys.: Conf. Ser. 983 012062
[2] Rokach L and Oded M 2015 Data Mining with Decision Trees Theory and Applications 2nd Edition (Singapore: World Scientific)
[3] Witten I H, Eibe F, Mark A H and Christopher J P 2016 Data Mining: Practical Machine Learning Tools and Techniques (San Francisco: Morgan Kaufmann)
[4] Han J, Jian P and Micheline K 2011 Data Mining: Concepts and Techniques (Waltham: Morgan Kaufmann Publishers)
[5] Muslim M A, Aldi N and Budi P 2018 Proc. Int Conf. Inf Comm Technol 2018 142
[6] Muslim M A, Iin K and Endang S 2015 J. Theor. Appl. Inf. Technol. 78 70
[7] Tikariha P and Prashant R 2018 Proc. Int Conf. Recent Advnc Comput Comm 195
[8] Lemke C, Marcin B and Bogdan G 2015 Artif Intellgnc Rev 44 117
[9] Webb G 2000 Mach Learn 40 159
[10] Jelinek H F, Jemal H A, Andrei V K, Morshed U C and Andrew S 2014 Aus J. Medic Sci 1 10
[11] Padmanaban K R A and G Parthiban 2016 India J. Sci Technol 9 29
[12] Mantas C J and Joaquin A 2014 Expert Syst Appl 41 4628
[13] Cheng L C, Yan H H, and Shr H C 2017 J. Medic Systm 41 85