Journal of Applied Informatics and Computing (JAIC)
Vol.6, No.2, Desember 2022, pp. 136~140
e-ISSN: 2548-6861

Optimization of the Decision Tree Method using Pruning on Liver Disease Classification

Anindya Khrisna Wardhani[1]*, Ega Nugraha[2]*, Qonita Ulfiana [3]*
*Rekam Medis dan Informasi Kesehatan, Politeknik Rukun Abdi Luhur
anindya.khrisna@poltekun.ac.id [1], ega.nugraha@poltekun.ac.id [2], qonita.ulfiana@poltekun.ac.id [3]

Article Info

Article history:

Received 2022-08-08
Revised 2022-11-09
Accepted 2022-11-11

Keyword:

Data Mining,
Decision Tree,
Liver,
Pruning

ABSTRACT

The amount of data about liver disease can be used to become information that can be extracted using the decision tree data mining method. However, there is a weakness in the decision tree method, namely over-fitting the resulting tree can produce a good model in training data but normally cannot produce a good tree model when applied to unseen data. Based on experiments conducted using datasets taken from The UCI Machine Learning Repository database is the ILPD dataset which contains 583 clinical data with 10 attributes with a target output of 416 positive liver and 167 negative liver. The results show that the decision tree algorithm using pruning and without pruning has been tested showing an increase in accuracy. The results of the decision tree performance without pruning generated in the confusion matrix for the accuracy measure, which is 73.58 %. While the results of the system performance using the pruning method have an accuracy of 73.76%. Although the accuracy value is slightly adrift, it can prove that the decision tree method using the pruning method has much better accuracy. In addition, the models and rules generated by the decision tree can be used as the basis for developing a prototype application for liver disease classification.

I. INTRODUCTION

The liver is one of the most important organs of the human body. The function of the liver is to detoxify toxins in the human body and control cholesterol and fat in the human body. If the liver is damaged, health will be disturbed, even death. Some diseases that attack the liver one of which is Hepatitis. According to WHO data, the Hepatitis B virus attacks 350 million people in the world, especially Southeast Asia and Africa, causing 1.2 million deaths per year [1]. The amount of data regarding liver disease can be used to become information that can be extracted. One technique that is able to explore hidden information from multidimensional data sets that have been obtained is *data mining techniques*[2].

Data mining technology can be utilized on claim data. From this data, it will provide information on the factors that influence the results of claims, namely claims that are feasible and *pending* [3]. One of the *data mining methods* that can be used for data classification is a *decision tree*[4]. *Decision Tree is a flowchart* structure that has a *tree* , where each internal node indicates an attribute test, each branch represents the test result, and the leaf node represents a class or class distribution [5]. *Decision trees* tend to be simpler, easier to understand, because the structure of a *decision tree* that resembles a tree shape can be displayed graphically, and is very easy to interpret even by ordinary people. *Decision tree* can be used to predict a value [6]. The *decision tree* method algorithm used in several studies is ID3, J48, Naïve Bayes [7] and C4.5 [8]; [9]; [10]; [11]; [12].

Decision tree and C4.5 algorithm are two inseparable models, therefore to build a *decision tree*, C4.5 algorithm is needed. The C4.5 algorithm is the development of ID3. Some of the developments carried out by C4.5 are being able to overcome *missing values, continue data* and *pruning* [8]. In *decision tree* C4.5, *pruning is part* of the decision tree formation process. When forming a *decision tree*, some *nodes* are *outliers* or the result of *noise data*. The application of *pruning* to the *decision tree* can reduce *outliers* and *data noise* in the initial *decision tree* so that it can increase the accuracy of data classification. [12].

Pruning is a process carried out to cut or remove some branches (branches) that are not needed. Pruning is done to develop the generalization reliability of the Decision Tree and the accuracy of the Decision Tree predictions by moving nodes that are not needed in the Decision Tree[13]. Branches (branches) or nodes that are not needed can cause the size of the Decision Tree to be very large and this is called over-fitting[14]. For now, overfitting is a research trend among researchers. Over-fitting can produce good models in training data but normally cannot produce good tree models when applied to unseen data. Over-fitting is caused by noisy data, irrelevant features [15]. Noisy data will cause misclassification, so over-fitting will cause a poor level of accuracy in classification.

Research related to the *decision tree model* [7]describes a comparison of ID3, J48 and Naïve Bayes to detect cases of health insurance fraud. The result of this research is that the *decision tree* using ID3 is the algorithm with the best level of accuracy. It takes 0.02 seconds to build the model. ID3 has the highest level of accuracy, which is 100% and the lowest accuracy is owned by J48, which is 96,7213%. In research [16], the decision tree *method* was developed for large-scale health insurance claim data. In this study, data on insurance claims used as many as 242,075 data. The *decision tree approach* is able to predict the Matthews correlation coefficient (MCC) of 0.426. This method is significantly better than the annual model, which reaches 0.375 for the group of insurance users [17]. In addition, the development of a predictive model that can predict the possibility of claims based on risk factors has been carried out. Decision tree analysis was adopted and developed with a predictive model. The error rate in the decision tree is low and indicates that the model is well validated and suitable for predicting future claims considering the data flow and risk characteristics [18]

In 2019, there was a study that discussed the classification of liver disease in the ILPD dataset using the Decision Tree C4.5 Algorithm. Based on the results of the processing carried out, it was found that the Decision Tree C4.5 Algorithm resulted in an accuracy value of 72.67% and also proved that of the 11 liver disease variables in the ILPD dataset, only 2 variables (Almine Alminotransferase) were the main factors in determining liver disease [19].

Another study [6] proved that the decision tree method is the simplest and with the most easily understood structure, and requires the shortest interpretation time compared to the random forest and convolutional neural network methods. Based on the results of testing the Naïve Bayes algorithm, k-Nearest Neighbor, Decision Tree, and Neural Network to solve the problem of classifying patients with liver disease or not using the RapidMiner studio9.1 application. The data was taken from the UCI Machine Learning Repository, namely the Indian Liver Patient Dataset (ILPD). The results show that of the four algorithms, the best and most suitable algorithm for the classification of liver patients is Decision Tree with an accuracy of 72.89%. In addition to the highest accuracy, Decision Tree is also able to classify patients who suffer from liver disease with a greater number so that it is considered accurate. So far, there are many algorithms that have high accuracy values but are unable to classify correctly, many even detect that patients do not have liver, even though the original data has liver damage. On the ROC curve, only the Decision Tree algorithm has a Y-axis graph close to 1.00 which is categorized as an "Excellent" classification [20].

From several studies that have been described above, it is explained that the decision tree algorithm is still being developed, especially in research to increase accuracy. The existence of this research is expected to find out the decision tree algorithm data mining system with *pruning method* to overcome pruning on nodes so as to improve the performance of the decision tree algorithm.

II. METHOD

The research carried out includes processing ILPD datasets using application assistance. In this case the assistance application in question is Rapidminer version 7.4. The dataset used in this study was taken from the UCI Machine Learning Repository database. The ILPD dataset contains 583 clinical data with 10 attributes with a target output of 416 positive liver and 167 negative liver.

Various modeling techniques were selected and applied to the prepared dataset to address appropriate business needs. The technique used is the *classification technique* using the *decision tree* C4.5 method [3]. The modeling stage also includes an assessment and comparative analysis of the various models built. The stages of generating a decision tree using the C4.5 algorithm are as follows:

a. The initial stage of making a decision tree is to form a tree root, then the data is differentiated according to the attributes that match to form leaves.
b. Tree pruning is the process of pruning tree branches that are not needed by an already formed tree or in other words simplifying the size of the tree because the decision tree that is formed is usually large in shape. In addition, pruning is also carried out with the aim of reducing the number of errors in the prediction results.
c. Making decision rules The tree that has been formed is made a decision rule. By tracing from root to leaf, the rule is derived from the decision tree.

The stages of the decision tree being built are as follows:

a. First is to choose attribute as Root
b. Second branch creation on each value
c. Further division of cases in branches
d. In each branch, the process is repeated until all cases in each branch have the same class to determine the attribute as the root, adjusted to the highest *gain value* of the existing attributes.

How to find the *gain value* is with the equation below:

$$Gain\ (S, A) = Entropy(S) - \sum_{i=1}^{n} - \frac{|Si|}{|S|} * Entropi\ Si \quad (1)$$

Information:
S : Set of cases

A : Attributes
N : Number of partition attribute A
| Si | : Number of cases on partition i
| S | : Number of cases in S

while to produce the value of *Entropy* is by the following formula:

$$Entropy\ (S) = \sum_{i=1}^{k} - P_i * log_2 P_i \quad (2)$$

Information:
S : Set of cases
A : Attribute N : Number of partitions S
Pi : Proportion of Si to S

In this study, the tests used were *Confusion matrix* . The confusion *matrix is* used to measure the performance of the model, because the *confusion matrix* is a useful tool to analyze how well the classifier can recognize tuples/features from different classes [4]. *The confusion matrix* can help show the details of the classifier's performance by providing information on the number of features of a class that are classified correctly and incorrectly [4]. *The confusion matrix* provides an assessment of the performance of the classification model based on the number of objects that are predicted correctly and incorrectly [6]. The *confusion matrix* is a 2-dimensional matrix that describes the comparison between prediction results and reality, shown in Table 2.2. If the predicted value is correct and the actual value is correct, it is called *True Positive* (TP).

If the predicted value is correct and the actual value is false, it is called a *False Positive* (FP) [2]. If the predicted value is wrong and the actual value is correct, it is called a *False Negative* (FN). If the predicted value is wrong and the actual value is wrong, it is called *True Negative* (TN). Good results are seen from the high diagonal values from the top left (TP) to the bottom right (TN), and the low diagonal from the bottom left (FP) to the top right (FN).

TABEL I
CONFUSION MATRIX MODEL

Class		**True value**	
		Right	**Wrong**
Predicted Value	**Right**	TP (*True Positive*)	FN (*False Negatives*)
	Wrong	FP (*False Positive*)	TN (*True Negatives*)

Description:
TP : positive prediction results and the actual value is also positive (*true positive)*
TN: the prediction result is negative and the actual value is also negative (*true negative)*
FN: the predicted result is negative while the actual value is positive *(false negative)*
FP: the predicted result is positive while the actual value is negative (*false positive)*

After the *confusion matrix has been created* , the accuracy, sensitivity, or called *recall* or *True Positive Rate* (TPrate), *specificity* (firmness) or called *True Negative Rate* (TNrate), *False Positive Rate* (FPrate), *False Negative Rate* (FNrate), *precision are calculated.* or called *Positive Predictive Value* (PPV), *Negative Predictive Value* (NPV), *F-Measure, Geometric Mean (G-Mean)* , and *Area Under the ROC Curve* (AUC).

The formulas used to perform the calculations are:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

$$\text{Sensitivity (recall)} = \frac{TP}{TP+FN} \quad (4)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (5)$$

III. RESULTS AND DISCUSSION

Based on the results of decision tree modeling using the pruning method on the dataset, here are the results of calculating the model performance by creating a confusion matrix as shown in Figure 1.

accuracy: 73.76%

	true Liver	true NonLiver	class precision
pred. Liver	416	153	73.11%
pred. NonLiver	0	14	100.00%
class recall	100.00%	8.38%	

Figure 1. Confusion Matrix

Based on the confusion matrix table data, it can be concluded that:

a. True Positive (TP) = 416 data from one current class that can be predicted correctly in the current class.
b. true negative (TN)=153 data from one crash class that can be predicted correctly in the crash class
c. false positive (FP) = 0 data from conditions where the current class has a wrong prediction on the stuck class, while
d. false negative (FN) = 14 data from conditions in the crash class that were predicted to be wrong in the current class.
e. *Accuracy* = 73.76%

Knowledge generated by the decision tree algorithm can be presented in two forms, namely decision trees and rules using IF-THEN.

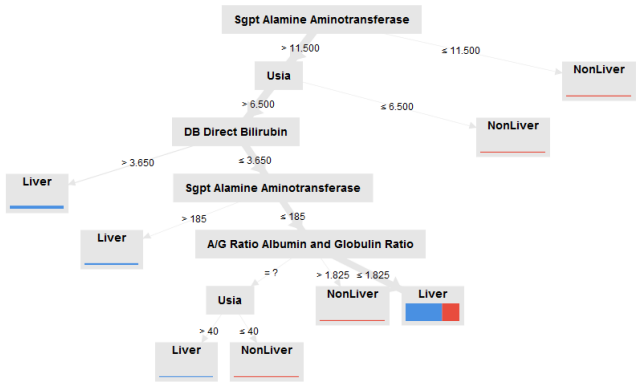

Figure 2. Decision Tree using Pruning

The decision tree is read from top to bottom or from the root (the first top node) to the leaf (the outermost node that no longer has branches). As for the rules for each class and the population records that meet these rules can be described as in the rule of the following tree model:

Tree

```
Sgpt Alamine Aminotransferase > 11.500
|   Usia > 6.500
|   |   DB Direct Bilirubin > 3.650: Liver (Liver=65, NonLiver=0)
|   |   DB Direct Bilirubin ≤ 3.650
|   |   |   Sgpt Alamine Aminotransferase > 185: Liver (Liver=26, NonLiver=0)
|   |   |   Sgpt Alamine Aminotransferase ≤ 185
|   |   |   |   A/G Ratio Albumin and Globulin Ratio = ?
|   |   |   |   |   Usia > 40: Liver (Liver=2, NonLiver=0)
|   |   |   |   |   Usia ≤ 40: NonLiver (Liver=0, NonLiver=2)
|   |   |   |   A/G Ratio Albumin and Globulin Ratio > 1.825: NonLiver (Liver=0, NonLiver=3)
|   |   |   |   A/G Ratio Albumin and Globulin Ratio ≤ 1.825: Liver (Liver=323, NonLiver=153)
|   Usia ≤ 6.500: NonLiver (Liver=0, NonLiver=3)
Sgpt Alamine Aminotransferase ≤ 11.500: NonLiver (Liver=0, NonLiver=6)
```

Figure 3. Rule Decision Tree using pruning

As for the attribute weights that have an important role in determining the label on the classification of liver disease, it can be seen in Figure 4.

attribute	weight
DB Direc...	0.064
A/G Rati...	0.051
Sgpt Ala...	0.187
Usia	0.697

Figure 4. Attribute Weight

Based on the results of decision tree modeling without using the pruning method on the dataset, here are the results of calculating the model performance by creating a confusion matrix as shown in Figure 5.

accuracy: 73.58%

	true Liver	true NonLiver	class precision
pred. Liver	416	154	72.98%
pred. NonLiver	0	13	100.00%
class recall	100.00%	7.78%	

Figure 5. Confusion Matrix Without Pruning

Based on the confusion matrix table data, it can be concluded that:

a. True Positive (TP) = 416 data from one current class that can be predicted correctly in the current class.
b. true negative (TN)=154 data from one crash class that can be predicted correctly in the crash class
c. false positive (FP) = 0 data from conditions where the current class has a wrong prediction on the stuck class, while
d. false negative (FN) = 13 data from conditions in the crash class that were predicted to be wrong in the current class.
e. *Accuracy* = 73.58%

Knowledge generated by the decision tree algorithm can be presented in two forms, namely decision trees and rules using IF-THEN.

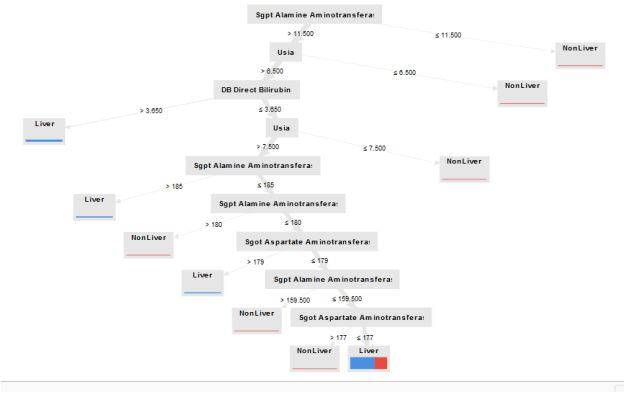

Figure 6. Decision Tree Without Using Pruning

The decision tree is read from top to bottom or from the root (the first top node) to the leaf (the outermost node that no longer has branches). As for the rules for each class and the population records that meet these rules can be described as in the rule of the following tree model.

Tree

```
Sgpt Alamine Aminotransferase > 11.500
|   Usia > 6.500
|   |   DB Direct Bilirubin > 3.650: Liver (Liver=65, NonLiver=0)
|   |   DB Direct Bilirubin ≤ 3.650
|   |   |   Usia > 7.500
|   |   |   |   Sgpt Alamine Aminotransferase > 185: Liver (Liver=26, NonLiver=0)
|   |   |   |   Sgpt Alamine Aminotransferase ≤ 185
|   |   |   |   |   Sgpt Alamine Aminotransferase > 180: NonLiver (Liver=0, NonLiver=1)
|   |   |   |   |   Sgpt Alamine Aminotransferase ≤ 180
|   |   |   |   |   |   Sgot Aspartate Aminotransferase > 179: Liver (Liver=19, NonLiver=0)
|   |   |   |   |   |   Sgot Aspartate Aminotransferase ≤ 179
|   |   |   |   |   |   |   Sgpt Alamine Aminotransferase > 159.500: NonLiver (Liver=0, NonLiver=1)
|   |   |   |   |   |   |   Sgpt Alamine Aminotransferase ≤ 159.500
|   |   |   |   |   |   |   |   Sgot Aspartate Aminotransferase > 177: NonLiver (Liver=0, NonLiver=1)
|   |   |   |   |   |   |   |   Sgot Aspartate Aminotransferase ≤ 177: Liver (Liver=306, NonLiver=154)
|   |   |   Usia ≤ 7.500: NonLiver (Liver=0, NonLiver=1)
|   Usia ≤ 6.500: NonLiver (Liver=0, NonLiver=3)
Sgpt Alamine Aminotransferase ≤ 11.500: NonLiver (Liver=0, NonLiver=6)
```

Figure 7. Rule Decision Tree using pruning

Meanwhile, the attribute weights that have an important role in determining the label on the classification of liver disease can be seen in Figure 8.

attribute	weight
DB Direc...	0.082
Sgpt Ala...	0.465
Usia	0.270
Sgot Asp...	0.183

Figure 8. Attribute Weight

V. CONCLUSION

Based on the experiment, it was found that the Decision Tree algorithm using pruning and without pruning that was tested showed an increase in accuracy . The results of the decision tree performance without pruning generated in the confusion matrix for the accuracy measure, which is 73.58 %. While the results of the system performance using the pruning method have an accuracy of 73.76%. Although the accuracy value is slightly adrift, it can prove that the decision tree method using the pruning method has much better accuracy. In addition, the models and rules generated by the decision tree can be used as the basis for developing a prototype application for liver disease classification.

REFERENCES

[1] T. Assegie, "Support Vector Machine And K-Nearest Neighbor Based Liver Disease Classification Model", Indonesian Journal of Electronics, Electromedical Engineering, and Medical Informatics, vol. 3, no. 1, pp. 9-14, Feb. 2021.

[2] A. Wardhani, "Implementasi Algoritma K-Means Untuk Pengelompokkan Penyakit Pasien Pada Puskesmas Kajen Pekalongan," *Jurnal Transformatika*, vol. Volume 14 No 1, pp. 30–37, 2016.

[3] T. Bimantoro and A. K. Wardhani, "Implementasi Algoritma Partitioning Around Medoids Dalam Pengelompokan Restoran," *Indonesian Journal of Technology, Informatics and Science (IJTIS)*, vol. 2, no. 1, pp. 33–36, 2020, doi: 10.24176/ijtisv2i1.5651.

[4] A. K. Wardhani, "Penerapan Algoritma Partitioning Around Medoids Untuk Menentukan Kelompok Penyakit Pasien (Studi Kasus: Puskesmas Kajen Pekalongan)," 2017.

[5] A. K. Wardhani, C. E. Widodo, and J. E. Suseno, "Information System for Culinary Product Selection Using Clustering K-Means and Weighted Product Method," vol. 165, no. ICCSR, pp. 18–22, 2018, doi: 10.2991/iccsr-18.2018.5.

[6] T. Lan, H. Hu, C. Jiang, G. Yang, and Z. Zhao, "A comparative study of decision tree, random forest, and convolutional neural network for spread-F identification," *Advances in Space Research*, vol. 65, no. 8, pp. 2052–2061, 2020, doi: 10.1016/j.asr.2020.01.036.

[7] R. Pal and S. Pal, "Application of Data Mining Techniques in Health Fraud Detection," *International Journal of Engineering Research and General Science*, vol. 3, no. 5, pp. 129–137, 2015.

[8] Rismayanti, "Decision Tree Penentuan Masa Studi Mahasiswa Prodi Teknik Informatika (Studi Kasus: Fakultas Teknik dan Komputer Universitas Harapan Medan)," *Jurnal Sistem Informasi*, vol. 02, no. 01, pp. 16–24, 2018.

[9] C. A. Sugianto, "Penerapan Teknik Data Mining Untuk Menentukan Hasil Seleksi Masuk Sman 1 Gibeber Untuk Siswa Baru Menggunakan Decision Tree," pp. 39–43, 2017, doi: 10.31227/osf.io/vedu7.

[10] E. P. Cynthia and E. Ismanto, "Metode Decision Tree Algoritma C.45 Dalam Mengklasifikasi Data Penjualan Bisnis Gerai Makanan Cepat Saji," *Jurasik (Jurnal Riset Sistem Informasi dan Teknik Informatika)*, vol. 3, no. July, p. 1, 2018, doi: 10.30645/jurasikv3i0.60.

[11] P. Bimo, N. Setio, D. Retno, S. Saputro, and B. Winarno, "Klasifikasi dengan Pohon Keputusan Berbasis Algoritme C4.5," *PRISMA, Prosiding Seminar Nasional Matematika*, vol. 3, pp. 64–71, 2020.

[12] D. Rosdiana and A. H. Rismayana, "Prediksi waktu tanam cabai menggunakan algoritma c4.5," pp. 436–442, 2018.

[13] C. C. Chern, Y. J. Chen, and B. Hsiao, "Decision tree-based classifier in providing telehealth service," *BMC Med Inform Decis Mak*, vol. 19, no. 1, pp. 1–15, 2019, doi: 10.1186/s12911-019-0825-9.

[14] A. Brunello, E. Marzano, A. Montanari, and G. Sciavicco, "Decision tree pruning via multi-objective evolutionary computation," *Int J Mach Learn Comput*, vol. 7, no. 6, pp. 167–175, Dec. 2017, doi: 10.18178/ijmlc.2017.7.6.641.

[15] Y. Rokhayati, N. Z. Jannah, S. Irawan, and D. E. Kurniawan, "Decision Determination of Hinterland Selection Using Analytical Network Process," in 2019 2nd International Conference on Applied Engineering (ICAE), Oct. 2019, pp. 1–5. doi: 10.1109/ICAE47758.2019.9221825.

[16] Y. Xie *et al.*, "Predicting Days in Hospital Using Health Insurance Claims," *IEEE J Biomed Health Inform*, vol. 19, no. 4, pp. 1224–1233, 2015, doi: 10.1109/JBHI.2015.2402692.

[17] Y. Xie *et al.*, "Analyzing health insurance claims on different timescales to predict days in hospital," *J Biomed Inform*, vol. 60, pp. 187–196, 2016, doi: 10.1016/j.jbi.2016.01.002.

[18] N. K. Frempong, N. Nicholas, and M. A. Boateng, "Decision Tree as a Predictive Modeling Tool for Auto Insurance Claims," *Int J Stat Appl*, vol. 7, no. 2, pp. 117–120, 2017, doi: 10.5923/j.statistics.20170702.07.

[19] I. Setiawati, A. P. Wibowo, A. Hermawan, M. Teknologi, I. Universitas, and T. Yogyakarta, "Implementasi Decision Tree Untuk Mendiagnosis Penyakit Liver," 2019.

[20] A. P. Ayudhitama and U. Pujianto, "Analisa 4 Algoritma Dalam Klasifikasi Penyakit Liver Menggunakan Rapidminer," *JIP (Jurnal Informatika Polinema)*, 2020.