THE EFFECT OF WEIGHT OPTIMIZATION USING GENETIC ALGORITHM ON THE CLASSIFICATION OF DHF VULNERABILITY LEVEL

Bayu Hadi Sudrajat[1], Muliadi[2], Muhammad Reza Faisal[3]
Fakultas Matematika dan Ilmu Alam Universitas Lambung Mangkuratama instansi
Jl. A. Yani Km. 36 Banjarbaru, Kalimantan Selatan, telp. (0511) 4773112
bayuhadisudrajat@gmail.com

Abstract

Dengue Hemorrhagic Fever (DHF) is a disease transmitted by the Aedes Ageypti mosquito. In South Kalimantan, especially in the city of Banjarbaru the number of cases tends to increase every year. Existing research has identified the level of dengue susceptibility by using computational methods, one of which is classification. The method used in this research is Neural Network Backpropagation with weight optimization using Genetic Algorithms for data classification of dengue disease in Banjarbaru City. The purpose of this study was to determine the performance of the classification of dengue susceptibility levels using Neural Network Backpropagation and weighting using Genetic Algorithms. The results showed that the performance obtained for the classification of the level of dengue susceptibility using the Neural Network Backpropagation Algorithm was 83.33% in the accuracy, 96.51% precision, and 84.69% recall, whereas when using the Neural Network Backpropagation Algorithm based on Genetic Algorithm for weight optimization, obtained an accuracy value of 96.29%, a precision of 98.97%, and a recall of 97%.

Keywords: *Dengue Hemorrhagic Fever, Weight Optimization , Neural Network Backpropagation, Genetic Alghorithm.*

1. INTRODUCTION

Dengue hemorrhagic fever (DHF) is a contagious disease caused by the bite of Aedes Aegypti and Aedes Albopictus mosquitoes. Sukohar explained that transmission can occur when mosquito bites and sucks the blood of someone who has been infected with the dengue virus, when the mosquito bites another person, the virus will spread [1]. The World Health Organization (WHO) noted that in Southeast Asia, Indonesia is the region with the highest DHF cases. In 2016, there were 8487 dengue cases with 108 deaths. Delays in identifying these problems can cause control delays and ultimately can lead to events or outbreaks that are out of control. Therefore we need a method that can be used to study data as learning to classify the level of vulnerability of an area in the case of DHF.

Neural Network Backpropagation (NNBP) is a good method in classifying. However, in Nawi's research, he explained that this method has two weaknesses, namely this method takes a long time to find optimal points in learning and is unstable, this is because Neural Network Backpropagation is often trapped in a local minimum[2]. To overcome these weaknesses, an optimization algorithm is needed to overcome these weaknesses.

One of the popular algorithms in solving optimization problems is the Genetic Algorithm. In Suhendra et al's research, Genetic Algorithms are used as a problem solving in determining architectural parameters and weight values with a dataset taken from the UCI repository. In this study, the Genetic Algorithm can give better results than without optimization[3].

Through this problem, it can be seen that the Neural Network Backpropagation is a suitable method for classifying the level of dengue susceptibility, but it still has shortcomings in determining the initial weight value so that the Genetic Algorithm determines the initial weight so that it can produce better performance. This research will be conducted classification of dengue susceptibility level using the Neural Network Backpropagation method with weight optimization using the Genetic Algorithm method.

2. RESEARCH METHODOLOGY

The procedures for this research are as follows:

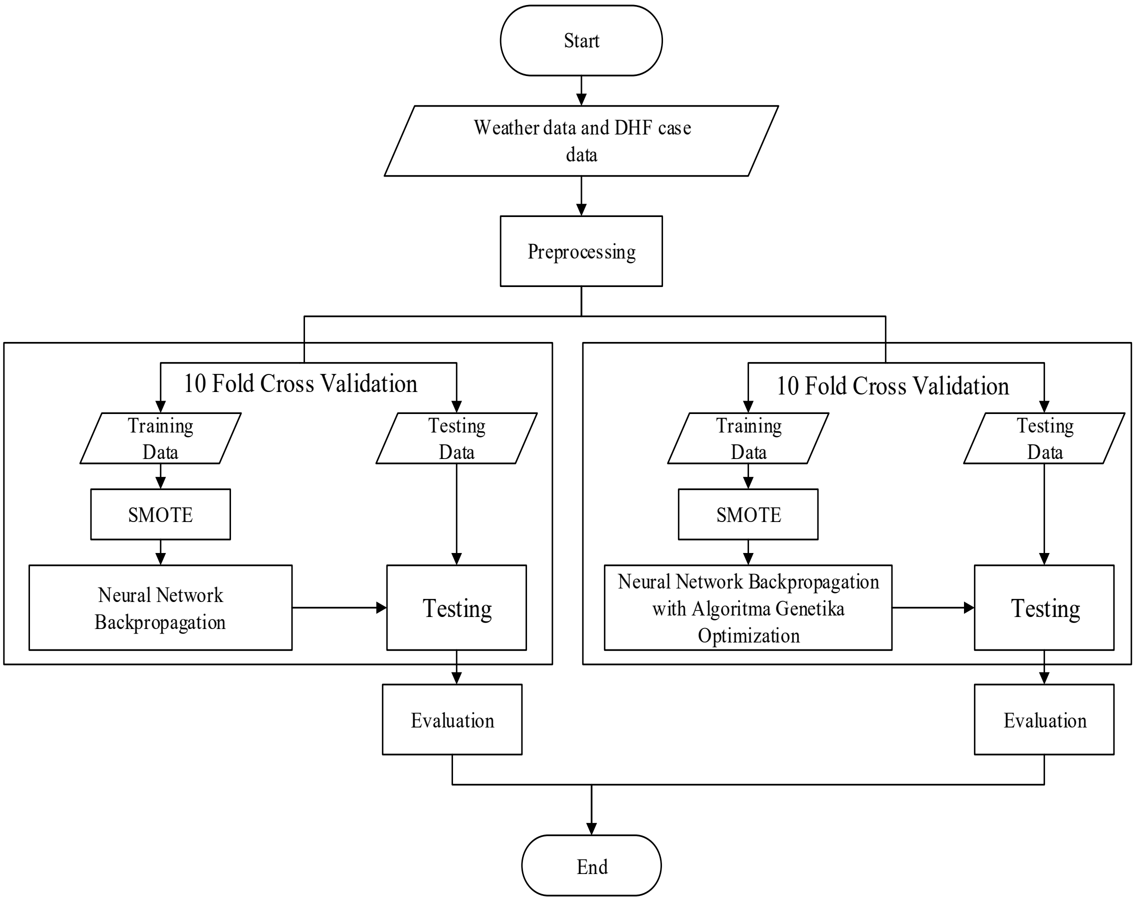

Figure 1 Procedure research

2.1 Data collection

The data used include data on the number of dengue cases obtained from the Banjarbaru City Health Office with a time span from 2017 to 2019 and from research by Zubaidah et al. [4] from 2008 to 2013, as well as population data and weather variables. consisting of temperature, humidity, rainfall, and the number of rainy days obtained from the BPS site for the city of Banjarbaru with a time span from 2008 to 2013 and 2017 to 2019.

2.2 Preprocessing

a. Merging data

This stage is the stage of combining the data that has been obtained such as the number of dengue fever heads and weather variables into excel before the data mining process is carried out.

b. Labeling

At this stage, labeling manually by calculating the value of the Incidence Rate(IR) with the following formula.

$$IR = \frac{Number\ of\ Case}{Total\ Population} \times 100.000\ Population$$

After obtaining the IR value, the data is classified as follows

Low : IR < 20

High : IR ≥ 20[5]

Examples of data that have been preprocessed and ready to be classified can be seen in table 1.

Table 1 Example dataset

No	Temperatu re (°C)	Humidity (%)	Rainfall (mm)	Number of rainy days (day)	Level of Vulnerability
1	26.1	88	466.6	28	Low
2	26.3	86	415	20	Low
3	26.4	87	237	21	Low
4	26.9	87	346	21	Low
5	27.2	86	326.1	23	Low
6	26.5	87	229.3	21	Low
7	26.3	85	154.1	18	Low
8	26.4	83	103.8	12	Low
9	27.2	80	90.5	13	Low
10	27.1	85	138.8	16	High
...	...	...	...	...	....
108	26.4	88	412.4	27	High

From the results of the preprocessing process, data consisted of classes with high dengue fever and low dengue fever levels. The amount of data from each class is shown in table 2.

Table 2 Number of dataset

Data	Total
Low	98
High	10
Total	108

In Table 2, it can be seen from a total of 108 data, 10 data with a high level of vulnerability and 98 data with a low level of vulnerability.

2.3 Split Data

Before classifying the data, the data were divided into two, namely training data and test data. Data sharing in this study uses the k-Fold Cross Validation data method. Cross Validation will divide the data into training data with a percentage of 90% of the total data or about 97 data and test data with a percentage of 10% of the total data or about 11 data, data sharing will be carried out by iterating as many as the number of k selected. Each iteration, the training data will be used as machine learning to recognize patterns from the data and the test data is used as a test of learning outcomes from the training data.

Due to data imbalance, a SMOTE (Synthetic Minority Oversampling Technique) process is carried out to balance the data. Data balancing is only done on training data as machine learning to recognize minor data. In this study, the training data, which amounted to 97 data, each iteration could have approximately 9 data with a high level of vulnerability and 88 data with a low level of vulnerability. Before entering the data classification process, oversampling will be carried out so that later data with a high level of vulnerability will be balanced with data with a low level of vulnerability. After the data is balanced, the amount of data between the classes will be the same or have the same amount of data, each class will have a total of 88 data.

2.4 Classification

After the data has been divided using Cross Validation, then the data is classified using two experiments, namely experiments with Neural Network Backpropagation without optimization and with Neural Network Backpropagation with weight optimization using Genetic Algorithms.

2.5 Evaluation

Evaluation is the performance appraisal stage of the data mining model that has been carried out. At this stage, the performance will be sought in the form of accuracy, recall, and precision using confusion matrix from the experimental results. If accuracy, recall, and precision produce the best and balanced values, the results of the classification process can be said to be correct.

Table 3. Table Confusion Matrix

Prediction	Actual	
	Low	High
Low	PP	PN
High	NP	NN

Information from the Confusion Matrix table is as follows:

a) PP (Positive - Positive) is a low amount of data that is predicted to be correct as low data.
b) PN (Positive - Negative) is the amount of high data that is wrongly predicted to be low data.
c) NP (Negative - Positive) is a low amount of data that is wrongly predicted to be high.
d) NN (Negative - Negative) is the amount of high data that is correctly predicted to be high.

The following is a formula for calculating the performance you are looking for can be seen in table 2:

Table 4 Classification performance search formula

Kinerja	Formula
Accuracy	$\frac{PP+NN}{PP+NN+PN+NP}$ x100%
Precision	$\frac{PP}{PP+NP}$ x100%
Recall	$\frac{PP}{PP+NP}$ x100%

3. RESULTS AND DISCUSSION

3.1 Results

a. NNBP Classsification without optimization

To get the best structure in NNBP and to produce the best classification performance of dengue hazard levels, it is necessary to do several tests on the existing parameters first. The test carried out in this research is testing the hidden layer parameters to find the best architecture to obtain optimal performance values. The test is to do several experiments on neurons in the hidden layer, while for other parameters the default value of Rapidminer is used, namely the learning rate with a value of 0.01, momentum with a value of 0.9, and a training cycle with a value of 200. Here are the results from testing the hidden layer parameters.

Table 5 Results testing hidden layer parameters

Hidden Layer		Accuracy	Precision	Recall
Number of hidden layers	*Neuron*			
1	5	79.63%	96.34%	80.61%
1	10	82.41%	96.47%	83.67%
1	15	79.63%	96.34%	80.61%
1	20	82.41%	96.47%	83.67%
1	25	82.41%	96.47%	83.67%
1	30	81.48%	95.35%	83.67%
1	35	83.33%	96.51%	84.69%
1	40	79.63%	96.34%	80.61%
1	45	79.63%	96.34%	80.61%
1	50	82.41%	96.47%	83.67%

From the experimental results of hidden layer parameters and hidden layer neurons, the best performance results were obtained with 1 hidden layer and the number of neurons in the hidden layer as many as 35. And the following results were obtained.

Table 6 Performance Testing without Optimization

Testing	Accuracy	Precision	Recall
Without Optimization	83.33%	96.51%	84.69%

b. NNBP classification with weight optimization using a Genetic Algorithm

Then, in the second experiment, the classification of Neural Network Backpropagation was carried out with weight optimization using Genetic Algorithms. Before classifying with the Neural Network Backpropagation method, first, the optimal weight is searched using a Genetic Algorithm. Because at the Cross Validation stage, the data balancing is carried out on the training data, so the weight search is also carried out at each iteration to get the weight value based on the learning data that has been balanced with the training data. The weight results obtained can be seen in table 7.

Table 7 Results of weights given by the Genetic Algorithm

Iterasi	Variabel	Bobot
1	Temperatur(°C)	0.843
	Humidity (%)	1.0
	Rainfall (mm)	0.0
	Number of rainy days (day)	0.403
2	Temperatur(°C)	1.0
	Humidity (%)	1.0
	Rainfall (mm)	0.0
	Number of rainy days (day)	0.0
3	Temperatur(°C)	0.473
	Humidity (%)	0.094
	Rainfall (mm)	1.0
	Number of rainy days (day)	0.0
…	…	…
	Temperatur(°C)	1.0
	Humidity (%)	0.0

10	Rainfall (mm)	1.0
	Number of rainy days (day)	0.0

The next step is to calculate the input value in the DBD dataset using the attribute weight values obtained. After that, the data is classified using the Neural Network Backpropagation method. The Neural Network Backpropagation parameters used in this second experiment are the same parameters as the first experiment, namely 1 hidden layer, 35 hidden layer neurons, a learning rate of 0.01, a momentum of 0.9, and a training cycle of 200. Meanwhile, the Genetic Algorithm parameter uses the default value from Rapidminer. The prediction results from the classification of Neural Network Backpropagation with weight optimization using Genetic Algorithms can be seen from the Confusion Matrix table in table 6.

Table 8 Performance Testing With Optimization

Testing	Akurasi	Presisi	Recall
With Optimization	96.29%	98.97%	97%

c. Performance comparasion

The next process is a comparison of the performance of the two experiments to determine the effect of weight optimization using Genetic Algorithms on the Neural Network Backpropagation. The results of the performance comparison can be seen in table 5 and figure 2.

Table 9 Comparations Performance

No	Method	Accuracy	Precision	Recall
1	NNBP	83.33%	96.51%	84.69%
2	NNBP + GA	96.29%	98.97%	97%

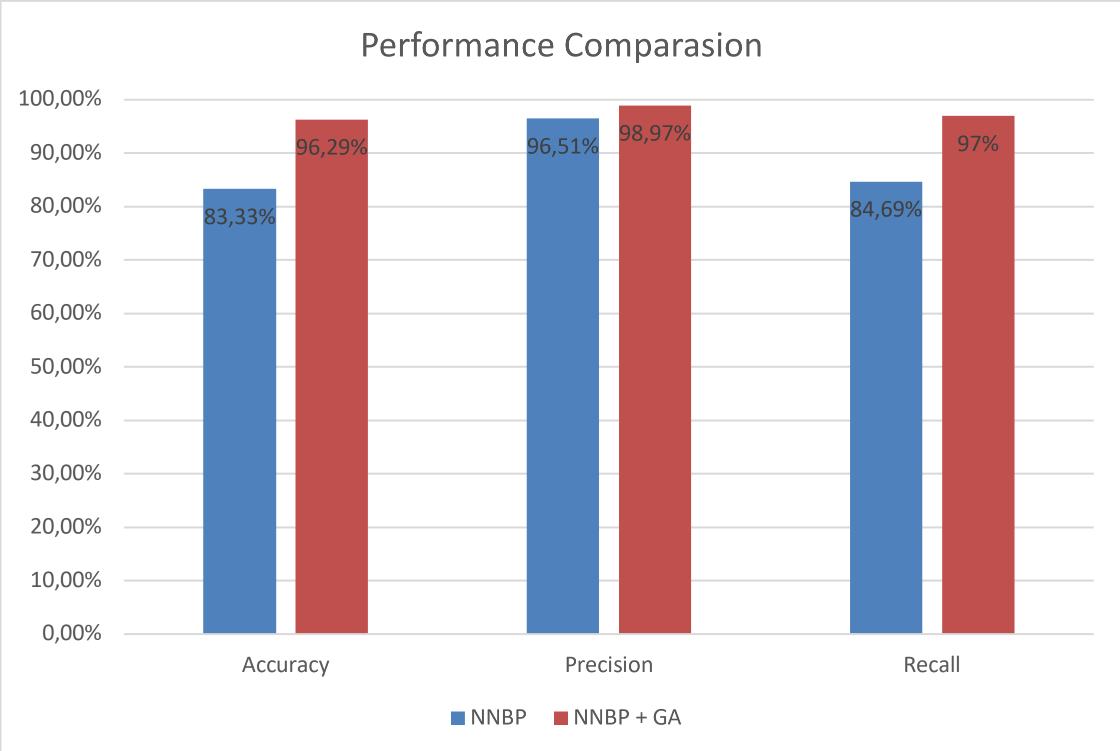

Figure 2 Comparassion Performance

3.2 Discussion

The results of the Neural Network Backpropagation performance without using weight optimization resulted in an accuracy performance of 83.33%, 96.51% precision, and 84.69% recall. By dividing the training data and test data using K-Fold Cross Validation with a k value of 10. In Cross Validation, each iteration is carried out by balancing the data on the training data only to make data with a high level of vulnerability become balanced with data with a low level of vulnerability. Before classification, testing is first carried out on the hidden layer to get the best architecture, and the best architecture is obtained, namely with one hidden layer and 35 neurons. Furthermore, the data is classified using Neural Network Backpropagation with a learning rate parameter with a value of 0.01, momentum with a value of 0.9, and a training cycle with a value of 200.

Then the next experiment is to add a Genetic Algorithm to find the best initial weight on the Neural Network Backpropagation. This experiment resulted in an accuracy performance of 96.29%, a precision of 98.97%, a recall of 97%. The distribution of training data and test data is still the same as the first experiment, namely using Cross Validation with a value of K = 10 and balancing the training data. The difference in this second experiment is that before the data is classified using the Neural Network Backpropagation, the initial weight is first searched using a Genetic Algorithm. The Neural Network Backpropagation parameter in this experiment is the same as the previous experiment, namely one hidden layer, 35 hidden layer neurons, a learning rate with a value of 0.01, a momentum with a value of 0.9, and a tryning cycle with a value of 200. the default of rapidminer.

4. CONCLUSION

This study concludes that after the Neural Network Backpropagation experiment was carried out without optimization it resulted in an accuracy performance of 83.33%, 96.51% precision, and 84.69% recall. Then the Neural Network Backpropagation experiment was carried out with weight optimization using the Genetic Algorithm resulting in an accuracy performance of 96.29%, a precision of 98.97%, a recall of 97%. Thus it can be concluded that doing weight optimization using Genetic Algorithms can produce better performance with evidence in the form of an increase in accuracy performance of 12.96%, precision of 2.46%, recall of 12.31%. The suggestion for this research is to add several more varied variables such as larvae-free rate and population density.

REFERENCES

[1] A. Sukohar, "Demam Berdarah Dengue (DBD)," *Medula*, vol. 2, no. 2, 2014.

[2] N. M. Nawi, A. Khan, and M. Z. Rehman, "A New Back-Propagation Neural Network Optimized with Cuckoo Search Algorithm," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, pp. 413–426, 2013, doi: 10.1007/978-3-642-39637-3_33.

[3] C. D. Suhendra and R. Wardoyo, "Penentuan Arsitektur Jaringan Syaraf Tiruan Backpropagation (Bobot Awal dan Bias Awal) Menggunakan Algoritma Genetika," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 9, no. 1, pp. 77–88, 2015, doi: 10.22146/ijccs.6642.

[4] T. Zubaidah, M. Ratodi, and L. Marlinae, "Pemanfaatan Informasi Iklim Sebagai Sinyal Perigatan Dini Kasus DBD Di Banjarbaru," *Vektora*, vol. 8, no. 2, pp. 99–106, 2016.

[5] Kemenkes RI, "Demam Berdarah Dengue," *Buletin Jendela Epidemiologi*, vol. 2, 2010.