ISSN 2278-3091
Volume 9 No.2, March - April 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse146922020.pdf
https://doi.org/10.30534/ijatcse/2020/146922020

Artificial Neural Network Architecture Optimization for Heart Disease Classification using Genetic Algorithm

Dicky Liegar[1], Steven Ivander Junaidi[1], Sani M. Isa[1]
[1]Computer Science Department, BINUS Graduate Program – Master of Computer Science,
Bina Nusantara University
Jakarta, Indonesia 11480
{dicky.liegar; steven.junaidi; sani.m.isa}@binus.ac.id

ABSTRACT

Heart disease is a dangerous disease to be underestimated. The lack of cardiologists makes it difficult for the community to get medical care. The development of artificial intelligence made is possible to assist the community in detecting whether the patient's heart is healthy or not. In this work, we implemented genetic algorithm to optimize the architecture of artificial neural network from previous work and used same dataset from Cleveland Heart Disease Data. We have succeeded in getting an accuracy, precision, and recall of 93.4%, 89.7%, and 97.2% respectively, which is relatively higher compared to previous papers.
Key words : Genetic Algorithm, Heart Disease, Neural Network, Optimizer

1. INTRODUCTION

Human blood carries nutrients that the human body needs to function properly. Left Ventricular Hypertrophy is one example of what may disrupt the cardiac cycle process. Left ventricular hypertrophy may cause an enlargement and thickening of the walls of the left ventricle, this would mean that the heart would need to do an extra work to pump the blood from the left atrium to the left ventricle which would eventually cause the heart failing to pump.

Heart disease, often used interchangeably with the term cardiovascular disease or angina, is a serious matter that should not be underestimated. The American Heart Association has stated that Cardiovascular disease and stroke causes immense health and economic burden globally [1]. One in every four deaths is caused by heart disease. A lot of things can be considered as features when we are dealing with heart disease. Smoking, physical inactivity, overweight, genetics, and high blood cholesterol are some major examples of what can increase the risk of stroke. Our heart pumps blood containing oxygen and nutrition to the whole body [2]. Failing to do so should be accounted as heart disease. Anything that may damage or interfere with the cardiac cycle could be identified as the cause of heart disease.

Lack of cardiologist has always been a world problem [3]. An accurate and efficient method is a must when we are dealing with medical issues. The rise of artificial intelligence and deep learning could be used to aid physiologist or cardiologist on giving an exact result on whether the patient's heart is healthy or not. Machine learning has been avoided in the work of medical services because of how it tends to have low performance. Given the rise of deep learning and similar methods, researchers have started to move to use machine learning to do medical diagnosis [4]. However [5] states an AI model that does decision making is no easy task. There are challenges such as time limit, consumption cost or uninformed information. Which complements the argument made by [6] that machine learning has two problems. First, they do not integrate real-time progress, and they fall under the principle of "one model for all" [6]. Both of these problems causes the model's performance to gradually decreases.

Knowing so, there are several previous works [7][8][9] that focus on heart disease classification. These works used open source dataset from Cleveland Heart Disease Data [10]. This dataset contains 303 records of patient with normal condition and heart disease. Each record has 13 independent variables and 1 dependent variable. Independent variables consist of both continuous variables and categorical variables.

Maheswari have successfully implemented modern model such as deep learning and traditional model such as Neural Network (NN) with 84% accuracy [7]. What could be the downfall of this research is that the author did not consider the variables that are categorical. A neural network will perform less given categorical variables. Hence, we applied dummification or one-hot-encoding to the dataset. A researcher implemented support vector machine (SVM) as a prediction model which performs well and achieved 89%

accuracy [8]. SVM would perform better on a linearly separable dataset, which the dataset is not. The Cleveland dataset had a combination of linearly separable and non-linearly separable variables. Thus, a neural network would obviously perform better. Chitra Jegan [9] leveraged Particle Swarm Optimization (PSO) with Neural Network reached an accuracy of 90.8%. PSO is compatible for continuous variables while Genetic Algorithm (GA) works well on discrete values [11]. Hypothetically, genetic algorithm and neural network would perform better given the dataset is combination of categorical variable and numerical, rather than PSO on a continuous variable. Therefore, we decided to select GA and NN as our predicting model. Given these literature reviews, it is observable that GA has never been tried to be used as an optimizer for this specific topic.

The following study is organized as follows. Chapter 2 provides information about the flowchart, dataset, data preprocessing, and methods used in this study such as artificial neural network and genetic algorithm. Chapter 3 will explain about the results including the analysis of the results. Finally, Chapter 4 concludes result of our study.

2. MATERIALS AND METHODS

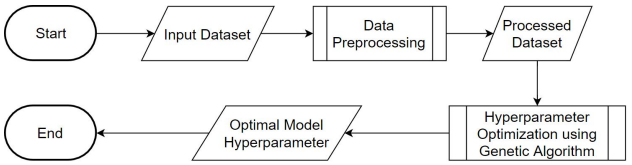

Figure 1: Flowchart of Experiment

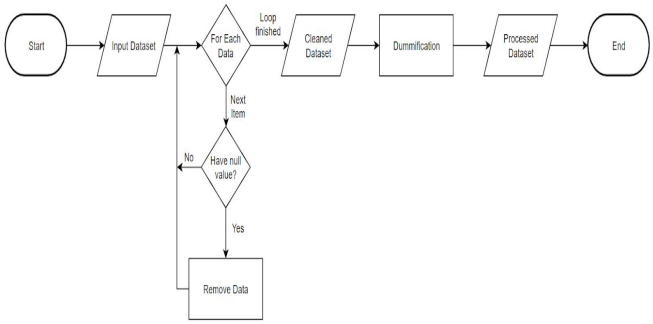

Figure 2: Flowchart of Data Preprocessing

2.1 Experiment

As shown in Figure 1, we could see the steps of what we did in this research. The first thing that we did was Data Preprocessing, which is needed to handle the raw dataset, since we need to make sure that the dataset can be used to train the neural network model and there will be no noise that can affect the proposed model. After that, we split the dataset randomly into training and testing dataset. Finally, we feed those datasets to search optimal model using Genetic Algorithm.

2.2 Dataset

We experimented with the dataset which originated from a machine learning repository with 76 attributes [10]. The heart disease datasets have exactly 303 rows with 6 rows having missing values. Most papers that uses this dataset has stated that 14 of these attributes are the most crucial for prediction. As stated on the introduction, these attributes have been converted to either numerical or categorical variables shown in Table 1.

Table 1: Feature Description

No	Variable	Type	Description
1	age	Numerical	The person's age in years (29 - 77 years old)
2	sex	Categorical	The person's sex 0: Female 1: Male
3	cp	Categorical	The chest pain experienced 1: Typical Angina 2: Atypical Angina 3: Non Anginal Pain 4: Asymptomatic
4	trestbps	Numerical	The person's resting blood pressure in mmHg (94 - 200 mmHg)
5	chol	Numerical	The person's cholesterol measurement in mg/dl (126 - 564 mg/dL)
6	fbs	Categorical	The person's fasting blood sugar 0: Less than or equal to 120 mg/dL 1: More than 120 mg/dL
7	restecg	Categorical	Resting Electrocardiographic measurement 0: Normal 1: Having ST-T wave abnormality 2: Showing probable or definite left ventricular hypertrophy
8	thalach	Numerical	The person's maximum heart rate achieved (71 - 202)
9	exang	Categorical	Exercise Induced Angina 0: No 1: Yes
10	oldpeak	Numerical	ST depression induced by exercise relative to rest (0 - 6.2)
11	slope	Categorical	The slope of the peak exercise ST segment 1: Upsloping 2: Flat

			3: Downsloping
12	ca	Numerical	The number of major vessels (0 - 4)
13	thal	Categorical	A blood disorder called thalassemia 3: Normal 6: Fixed defect
			7: Reversible defect
14	target	Categorical	Having heart disease 0: No 1: Yes

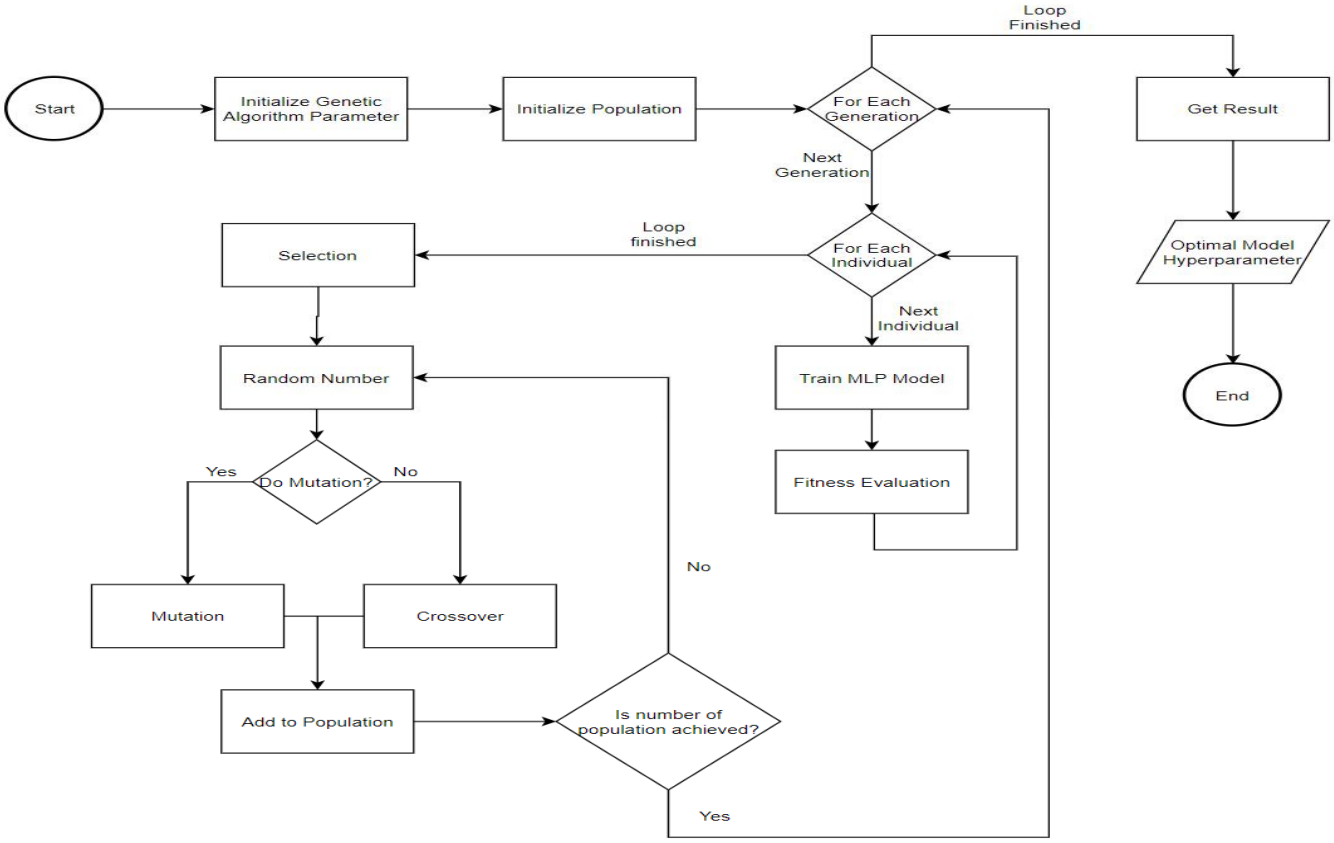

Figure 3: Flowchart of Genetic Algorithm

2.3 Data Preprocessing

As shown in Figure 2, we used 2 methods to preprocess the dataset which are data cleaning and dummification. In 303 rows of data, we found out there were 6 rows of missing values, which are in 2 variables, which are ca and thal. Therefore, we will only use 297 rows of data from the dataset. Neural Network is not optimized to take categorical data as input. Thus, we applied one-hot encoding, which will convert value for each categorical variable with more than 2 values into separate columns. We applied this method to all categorical variables. However, there is a chance that this may create a highly correlated relationship between the new variables, known as dummy variable trap. We removed one column for each generated categorical data to handle the problem. Therefore, our neural network will take input from 18 independent variables.

2.4 Artificial Neural Network

Neural network is a model that has been used everywhere recently. Neural networks can adapt to changing input, so the network generates the best possible result without needing to redesign the output criteria. Neural network has been widely used across fields in the industry, such as: voice recognition, and image recognition. Neural network imitates on how the human brain thinks and make decision. A neuron in a neural network is a mathematical function that collects and classifies information according to a specific architecture. For this study, we decided to use scikit-learn library and Multi Layer Perceptron (MLP) classifier for our neural network framework. Scikit-learn gives various machine learning models to be used at ease, and all that we needed to do is to tune the parameters.

2.5 Genetic Algorithm

A better and feasible way to optimize hyperparameter is to use Evolutionary Algorithm [12]. The first generation starts with a population consists of individuals which have different characteristics. The fittest individuals will most likely survive until the next phase of generation, which is known as Selection. Then some of the individuals will do crossover to create new children which resembles some of their characteristics, which is known as Crossover. Some of its children will have different characteristics from their parents, which is known as Mutation.

Figure 3 shows the full flowchart of how we implemented our genetic algorithm to fit the use case upon giving the output of an optimized neural network architecture. We used 1 hidden layer with three neurons which was proposed by [8] as our base architecture or first individual because this model hasn't been implemented with any architecture optimization yet. We will train model with hyperparameters from each individual. Each model will have a cost function to determine how well the model to classify, which is calculated by its error percentage of predicting the dataset. Smaller value of cost function indicates fitter model. Then, Selection process will select some percentages of individuals which will be in the next generation. These individuals will also be the parents during the process of Crossover and Selection to create new children. For each iteration across the new individual's gene, the operator will be selected randomly. This implies that the crossover operator is used for each gene in the chromosome. The probability of getting each parents gene is set equally or 50-50. The mutation operator will generate number randomly in range of maximum number of hidden layer neurons. Since doing mutation may produce zero value, the mutation process will take probability of changing the gene for the new child. If there is no mutation happened, then its gene will be based whether from the first parent or the other one. This process of creating new children will be stopped until the number of maximum individuals achieved, and these will be represented as the next generation. Finally, an optimized architecture would be output of the GA and used for prediction.

2.6 Individual Representation

We think that it is crucial to point out that we proposed that each gene will represent the number of neurons for each hidden layer which would then have the same shape shown in Figure 4. The chromosome size or number of genes for each individual must be same. Therefore, we propose that for each chromosome will have the same number of genes, which is represented as maximum number of hidden layers, and the gene can take input from 0 to a predefined number of hidden neurons.

x_1	x_2	...	x_i

Figure 4: Individual Representation

x: Number of neurons in hidden layer (Gene)
i: Number of maximum hidden layer (Chromosome Size)

In addition, we do have some restrictions for each individual. Every chromosome must have at least one gene with non-zero value and all zero-value genes must be shifted to the right, e.g. (1,0,5,0) shifted to (1,5,0,0). Lastly, no individual should be computed twice, hence we saved the result of an individual for faster computation.

3. RESULTS

3.1 Data Experiment

In this experiment, we tried to classify 297 rows of heart disease dataset. Using the literature reviews as comparison, we divided our model to 75:25 ratio, training and testing dataset respectively. After that, we assigned the hyperparameters for our genetic algorithm, such as: maximum number of generations, hidden layers, and hidden neurons.

3.2 Model Architecture

After over 100 generations, the fittest model of generation which have the best performance remains the same from 26-th generation. This architecture consists of 4 hidden layers with 3, 10, 5, and 8 hidden neurons, respectively. This model achieved better performance than the previous models.

3.3 Metrics

Upon evaluation the performances we decided to use general metrics. Accuracy is a percentage of total items classified correctly. Recall is the number of items correctly identified as positive out of total true positives. Precision is the number of items correctly identified as positive out of total items identified.

Table 2: Metrics Comparison

Model	Year	Metrics		
		Accuracy	Precision	Recall
Neural Network [7]	2017	84%		91.4%
SVM [8]	2018	89%		80.95 %
PSONN [9]	2014	90.8%		90.42 %
Our Model		93.4%	89.7%	97.2%

Through Table 2, we can see that our model has approximately 3.6% improvement of accuracy compared to relevant model. Genetic algorithm plays a huge role upon this result. The evolutionary algorithm helps us to not brute force our way to try every possible combination of layers. This proves that choosing the right hidden layers and neurons in its layers could improve the performance.

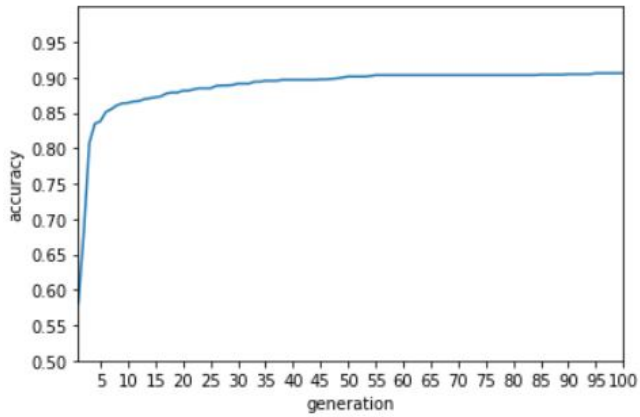

Figure 5: Average Accuracy Overtime

In Figure 5, It is observable that genetic algorithm help outperforms basic neural network. It is noticeable that the performance on the 10-th generation peaked and gradually improves slightly over several generations. After over 100 generations, we have successfully achieved a better model performance on the 25th generation with an accuracy of 93.4%, recall of 97.2%, and precision 89.7%. We noticed that it took an average of 3,77s (± 0,38s) to finish training and 0.001s (± 0,000005s) to finish testing. This proves that this method can be used to determine better architecture for heart disease model in quite short time. Our model's performance has stopped improving after 60 minutes. Thus, we decided to end the process.

4. CONCLUSION

The use of genetic algorithm to get the best architecture of neural network is highlighted in this paper. This research addresses the optimal number of hidden layers and neurons for each layer to achieve a high performance. Our neural network model produced an accuracy of 93,4% with 4 layers consist of 3, 10, 5, and 8 neurons respectively. This model is a better predicting model, considering its precision and recall. Knowing that our scope of this research is fulfilled. We have proven Genetic Algorithm combined with Neural Network can improve the performance on predicting the heart disease UCI dataset. We encourage the next researchers to try to implement another evolutionary algorithm on neural network or other algorithm as seen fit. We also think that it is worth a try to implement ensemble learning as it has a tendency to improve a performance further.

REFERENCES

1. Benjamin EJ, Blaha MJ, Chiuve SE, et al. Heart Disease and Stroke Statistics-2017 Update: A Report From the American Heart Association. Circulation. 2017 Mar;135(10):e146-e603. DOI: 10.1161/CIR.0000000000000485.
2. Jacob, Matthias & Chappell, Daniel & Becker, Bernhard. (2016). Regulation of blood flow and volume exchange across the microcirculation. Critical Care. 20. 10.1186/s13054-016-1485-0.
3. Ross, H., Higginson, L. A., Ferguson, A., O'Neill, B. J., Kells, C. M., Cox, J. L., & Sholdice, M. M. (2006). Too many patients, too few cardiologists to care?. The Canadian journal of cardiology, 22(11), 901–902. https://doi.org/10.1016/s0828-282x(06)70308-7
4. Molia, Hardik. (2019). TCP with Machine Learning - Advances and Opportunities. International Journal of Advanced Trends in Computer Science and Engineering. 8. 3526-3534. 10.30534/ijatcse/2019/132862019.
5. Abdulllah, M. & Alshehri, W. & Alamri, Saleem & Almutairi, Najla. (2019). ADSS: Automated Decision Support Systems. International Journal of Advanced Trends in Computer Science and Engineering. 8. 231-237. 10.30534/ijatcse/2019/4281.12019.
6. Tmimi, Mehdi. (2019). Proposal of a complete model and architecture of an intelligent adaptive hypermedia. International Journal of Advanced Trends in Computer Science and Engineering. 1491-1497. 10.30534/ijatcse/2019/70842019.
7. Maheswari, K., Jasmine, j., 2017. Neural Network based Heart Disease Prediction. INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) RTICCT – 2017 (Volume 5 – Issue 17).
8. Le, H., Tran, T., and Tran, L. 2018. Automatic Heart Disease Prediction Using Feature Selection and Data Mining Technique. Journal of Computer Science and Cybernetics. 34. 33-48. DOI:10.15625/1813-9663/34/1/12665.
9. Jegan, Chitra & seenivasagam.v,. 2014. Risk Prediction of Heart Disease Based on Swarm Optimized Neural Network. 255. 10.1007/978-81-322-1759-6_81.
10. Dua, D. and Graff, C. 2019. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.
11. Kachitvichyanukul, Voratas. (2012). Comparison of three evolutionary algorithms: GA, PSO, and DE. Industrial Engineering and Management Systems. 12. 215-223. 10.7232/iems.2012.11.3.215.
12. Young, Steven & Rose, Derek & Karnowski, Thomas & Lim, Seung-Hwan & Patton, Robert. 2015. Optimizing deep learning hyper-parameters through an evolutionary algorithm. 1-5. DOI:10.1145/2834892.2834896.