Predictive Maintenance of Heavy Equipment Machines using Neural Network Based on Operational Data

Ahya Radiatul Kamila*[1], Gerry Hudera Derhass [2], Johanes Fernandes Andry[3], Francka Sakti Lee[4], Very Budiyanto[5], Velly Anatasia[6]

[1,3,4,5,6]Universitas Bunda Mulia, Jl. Lodan Raya No. 2 Ancol, Jakarta Utara 14430
[1]Program Studi Sains Data, Universitas Bunda Mulia, Jakarta
[2]Kurnia Prima Nastari Corp
[3,4]Program Studi Sistem Informasi, Universitas Bunda Mulia, Jakarta
e-mail: *[1]**ahyaradiatul@gmail.com**, [2]gerryhudera@yahoo.com, [3]janry@bundamulia.ac.id, [4]flee@bundamulia.ac.id, [5]vbudiyanto@bundamulia.ac.id, [6]vanatasia@bundamulia.ac.id

Abstract

Preventive maintenance is a routine maintenance strategy that aims to maximize equipment life cycle and prevent unplanned downtime which causes increased repair costs. When carrying out this maintenance, error in selecting machines need to be anticipated to avoid company losses. This research aims to reduce human error in machine selection for preventive maintenance using deep learning. The dataset used in this research is operational data of heavy equipment machine dataset from one of the palm oil companies in Indonesia with 9 independent features and 1 dependent feature. Dependent feature is a target feature contain two target classes representing effective and ineffective machines. The dataset in this study contains outlier, feature scales that are very different, and imbalanced data class. To handle outlier and standardise data scale, the Z-score method is used. Meanwhile, the over sampling method is used to handle imbalanced data classes. To obtain the best model performance, the number of epochs and two types of optimizers (adam&adamax) of neural network are selected. In selecting the number of epochs, experiments were carried out using 100 epochs. This research obtained the linearity relationship between the number of epochs and accuracy with the accuracy values using Adam and Adamax optimizers were 94.82% and 93.11% at the 100th epoch.

Keywords— *Artificial neural network, Over sampling method, Preventive maintenance, Z-score*

1. INTRODUCTION

Plantation industry is an industry that engages in planting specific crops in growing media, such as soil in a suitable ecosystem. In addition, the plantation sector industry also carries out downstream activities, including processing raw agricultural products into finished or semi-finished goods and marketing them to both domestic and international markets. In doing so, the industry uses science, technology, capital, and management to reduce costs and increase company revenue [1]. Plantation industry, especially palm oil, is one of the industry that contribute to Indonesia's economic growth. This is reflected in the palm oil production capacity, which increased by more than 300% from 1990 to 2017, where Indonesia has been the world's largest palm oil producer since 2008 [2]. This sector is one of the source of economy and welfare for population because the palm oil industry is a labor-intensive industry with large-scale production. This creates need for heavy machinery to ensure that operational can be done effectively and efficiently. High productivity and human safety factors are also foundations for the use of heavy machinery in palm oil industry [3]. Therefore, the performance of heavy equipment plays a crucial role in palm oil industry [4].

One way to maintain the performance of heavy equipment is to implement predictive maintenance strategies. Predictive maintenance involves utilizing historical and real-time data to predict potential failures and identify machines that require maintenance before damage or failure occurs [5]. Unlike traditional preventive maintenance, which is performed on machines that are still in good condition, predictive maintenance focuses on accurately forecasting machine conditions to minimize unplanned downtime and reduce maintenance costs. The goal of predictive maintenance is to ensure maintenance activities are targeted and efficient, preventing unexpected failures while optimizing resources [6]. This approach leverages data-driven models and advanced technologies, such as machine learning and deep learning, to analyze patterns, detect anomalies, and predict failures. By distinguishing machines that are likely to fail from those that are functioning normally, predictive maintenance ensures that interventions are carried out only when necessary, improving overall maintenance effectiveness. However, implementing predictive maintenance addresses several challenges commonly faced in traditional maintenance methods, such as time constraints, limited human resources, inadequate testing equipment, and insufficient maintenance records. Relying on human inspections alone often leads to errors due to superficial assessments and undetectable issues without specialized tools. Predictive maintenance overcomes these limitations by providing an automated and accurate system for classifying machine conditions, enabling timely decision-making and reducing operational inefficiencies.

Various studies have been conducted to design systems for effective machine failure detection. For instance, predictive maintenance using the Internet of Things (IoT) and machine learning has been explored [7]. Machine learning algorithms, such as K-Nearest Neighbor and Support Vector Machine, have been compared for predictive maintenance classification. Other research has utilized methods autoencoder, which the encoder and the decoder parts consist of two LSTM layers [8]. Machine learning and deep learning algorithms are also used to predict the losses in predictive maintenance with data that obtained from the gearbox fault diagnosis simulator and machinery fault simulator [9]. Research conducted by [10] utilized the unsupervised learning method with a deep learning algorithm to identify faults accurately. Apart from that, a hybrid deep learning approach was conducted by [11] that combines the convolutional neural network, long-short-term memory, and attention technique.

The main objective of this research is to classify machines into two groups: effective machines and ineffective machines. This research contributes to the optimization of predictive maintenance methods. It employs an approach that maps data into distinct classes (classification) [12] utilizing a technique that enables devices to simulate human-like thinking through artificial intelligence [13]. The dataset used in this research was obtained from oil palm plantation companies and consists of both independent and dependent variables based on operationl data. The dependent variable represents the condition of heavy equipment at a given time, categorized into two classes: damaged machines and non-damaged (efficient) machines. However, the dataset is imbalanced, with fewer instances in the minority class. This imbalance can lead to detection errors, as the machine learning model receives less training on the minority class. To address this issue, the Synthetic Minority Oversampling Technique (SMOTE) was applied to balance the data classes. For data processing, the Artificial Neural Network (ANN) algorithm was utilized with hyperparameter tuning. To determine optimal hyperparameter values, this research compared the performance of the model across different numbers of epochs and two optimizers, Adam and Adamax.

2. RESEARCH METHODS

In this research, the research method is divided into two main parts. The first part discusses the stages in research design. This section explains the method of formulating important points in research. Meanwhile, the second part discusses the stages in processing data to produce good model performance.

.

2.1. Research Design

The sources used in this stage come from experts who work in the oil palm plantation sector and related literature. The stages in the research design process include several key steps. First, the interview stage involves obtaining information by collecting data related to the research problem [14]. In addition to this, an interview with an employee working in the plantation industry, specifically in the area of mechanization cost control and financial planning, was conducted to help the researcher gain a deeper understanding of the research objectives, targets, and contributions. The second stage is the study of literature, which involves reading, citing, and analyzing relevant theories and concepts that support the research field. Finally, the research design stage focuses on selecting the appropriate research methods, data collection techniques, data analysis approaches, and the tools that will be utilized throughout the research process.

2.2. Data Processing

Next section of research methodology discusses the stages in processing data in Figure 1. In processing the data, the author used several stages to produce good model performance. This is done by selecting data processing stages that are relevant to the dataset used.
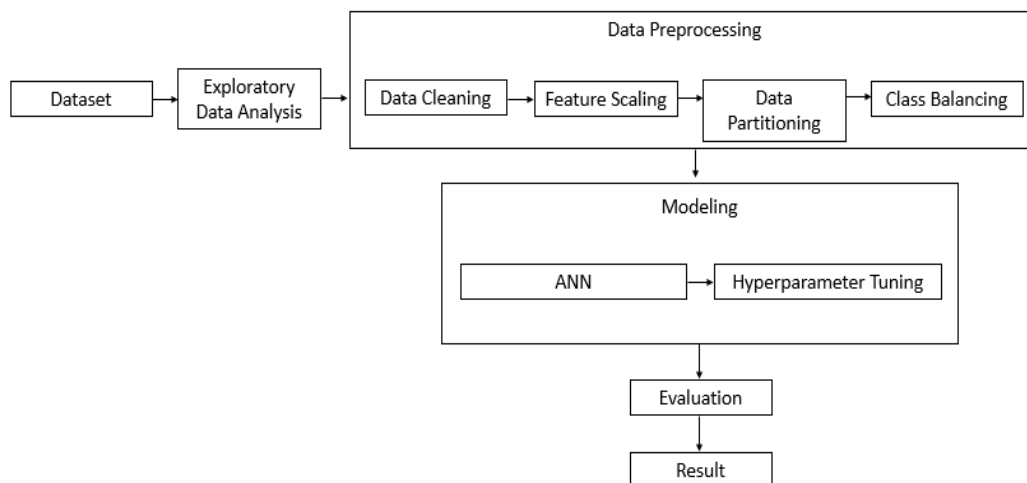

Figure 1. Block diagram of data preprocessing and modeling stages

The block diagram represents a typical machine learning workflow. It begins with the dataset input, where data is collected and loaded for analysis. Next, exploratory data analysis (EDA) is conducted to understand the characteristics of the data, identify patterns, and detect anomalies. Following EDA, data preprocessing steps are performed to prepare the data for modeling. This includes data cleaning to handle missing or erroneous values, feature scaling to standardize the range of numeric features, data partitioning to split the dataset into training and testing sets, and class balancing to address any class imbalances in classification problems. After preprocessing, the modeling stage begins, where an artificial neural network (ANN) is chosen as the model. During this phase, hyperparameter tuning is performed to optimize the performance of the neural network by adjusting parameters such as learning rate, number of layers, and nodes. Once the model is trained, it is evaluated using appropriate evaluation metrics, the result is presented, summarizing the model's performance and its ability to make accurate predictions.

2.2.1. Dataset

The dataset used by the writers was provided by one of Indonesia's palm oil corporations, consisting of 124,493 rows, each representing a heavy machinery unit with 9 independent features and 1 dependent feature. The data used in this study was obtained from operational activity unit recorded in the workshop of a palm oil company in Indonesia, which has plantations in Kalimantan. These operational data encompass various aspects of machinery performance and

usage in the field. Every activity involving the machinery—whether it pertains to maintenance schedules, repair actions, or daily operational logs—is manually recorded by technicians or equipment operators in the workshop. This workshop is fully responsible for ensuring the functionality, repair, and preventive maintenance of all heavy equipment used in plantation operations. As a result, the dataset reflects real-world operational conditions, offering valuable insights into the machinery's health and effectiveness over time.

Table 1. Dataset

Features	Explanation of The Features
Breakdown day	The day of unit breaks down
Standby day	The day of standby (not breaks down)
Cost maintenance	Unit repair costs
Kilometer/days	Units's kilometer/day
Times service	Number of units routine service
Weekly check	Weekly checks of unit condition
Daily check	Daily checks of unit condition
Oil leak	Total oil leak
Machine Efficiency	Target (dependent feature)

An overview of the key features recorded and used in this study is presented in Table 1. In the workshop, each time a machine is used or encounters an issue, the operators and technicians make detailed records of the machine's condition, including Breakdown day (the number of days the machine experiences failure or downtime), Standby day (the number of days the machine is ready but not in operation), and Cost maintenance (the maintenance cost incurred). Additionally, data regarding Kilometer/days (distance or operational duration of the machine), Times service (frequency of machine servicing), Weekly check and Daily check (routine weekly and daily inspections), as well as Oil leak (detected oil leaks) are also recorded every time maintenance is performed. The data collected in the workshop is then entered into the company's management system or internal database, which can be accessed by the analytics team. This manual recording process in the workshop ensures that the information gathered is based on direct observation of the machine's condition, providing an accurate picture of how operational and maintenance conditions can affect its efficiency. The data obtained is used for predictive analysis, helping the company plan maintenance and improve operational efficiency in the plantation.

2.2.2. Exploratory Data Analysis

In this stage, the data is analyzed to obtain insights with a deeper understanding of the data using visualization. Apart from that, this stage is also carried out to identify defects in the data. This is done by looking at the relevance, distribution, outliers and data anomalies. The use of data visualization in the exploratory data analysis stage also aims to help recognize data patterns to make analysis easier [15]. The analysis that carried out at this stage used as a reference in carrying out the next stage, data preprocessing stage. The data preprocessing stage has a big role in determining the performance of the final model. This is the main reason to carried out data analysis stage.

The initial step in exploratory data analysis is analyzing the completeness of data. In this research, no features were found with NaN values, in other words, all features had complete data. After analyzing the completeness of data, we carried out an outlier data analysis, which is an analysis to find out whether there are features that contain values which deviate greatly from other values. This is done to prevent a deterioration in model performance caused by outlier data. In this research, outlier data analysis was carried out using visualization with a boxplot as in Figure 2.
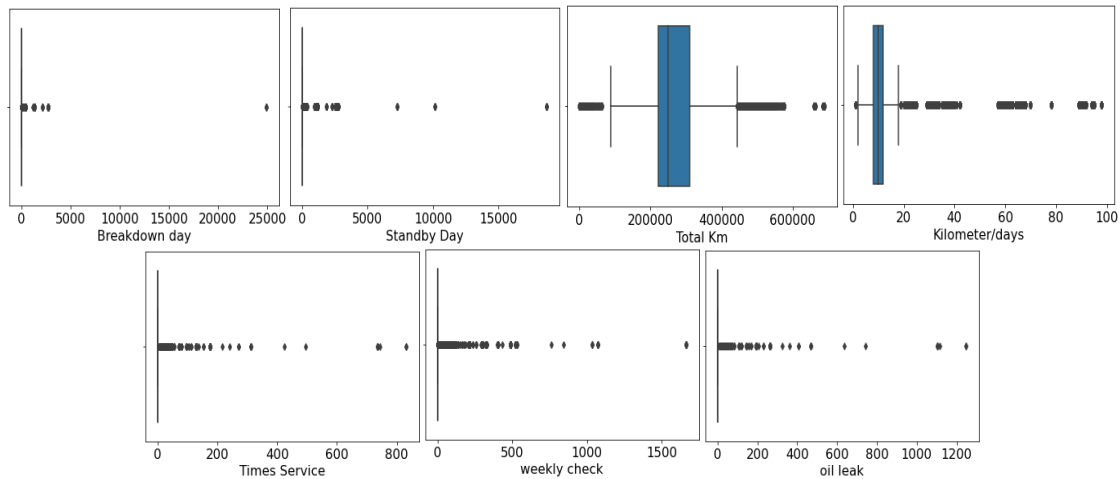
Figure 2. Boxplot

Figure 2 presents boxplots of the features, focusing on identifying outliers in the dataset. Outliers are typically managed in the following way after identification. First, outliers are detected using boxplots, where any data point lying outside of the "whiskers" (usually 1.5 times the interquartile range above the third quartile or below the first quartile) is considered an outlier. Once identified, it is crucial to examine whether these outliers are legitimate data points or the result of data entry errors. For instance, if Breakdown day shows an outlier of 50 days, it could either be a legitimate long-term breakdown event or a data recording mistake. Understanding the context behind the data is essential before deciding how to proceed. If an outlier is found to be an error or irrelevant, it may be removed entirely from the dataset. Alternatively, outliers can be capped to a maximum or minimum threshold based on domain knowledge. For example, if Kilometer/days shows unusually high values due to operational constraints, it could be capped at a more reasonable value. In some cases, transformations like logarithmic adjustments are applied to reduce the impact of outliers and normalize the data, particularly for features like Total km or Times service, where values can vary significantly. By managing outliers in a thoughtful and data-driven manner, the analysis remains robust, ensuring that predictions and insights derived from the data are more accurate and reliable.

The next step in this stage is data scale analysis which carried out to ensure that the data scale for each feature is in the same range of value. This is done because some machine learning algorithms are sensitive of the differences of data scale. If the data features vary greatly in scale, the algorithm will assign higher weights to features with larger values. This will affect the learning process and cause a deterioration in model performance.

The final step in the exploratory data analysis stage is data class imbalanced assessment that greatly influences the performance of machine learning. Model will learn more from classes with a larger amount (majority class) and learn less from classes with fewer numbers (minority class). If this happens, then there is a tendency to ignore the minority class. As a result, machine learning algorithms will tend to choose classes with a larger amount of data and will cause bias [16]. In this research, data class imbalanced assessment was carried out using data visualization. The amount of data for the efficient heavy equipment machine class is much greater than that for the inefficient heavy machine class, with the number of efficient machine classes are 124,388 and the number of data for the inefficient machine class are 106 data.

2.2.3. Preprocessing

The next stage in data processing is data preprocessing, which is the data preparation stage before the data is processed/trained by machine learning algorithms, which involves cleaning of transforming data [17]. In this stage, steps are taken to improve the quality of the raw data. Poor data quality can be caused by several reasons, including incomplete data, irrelevant data, imbalanced data classes, and so on. Data conditions like this will trigger poor model performance.

This is caused by bias, misinterpretation, sensitivity of machine learning algorithms, and so on. Therefore, a data preprocessing stage was carried out using steps to resolve each problem contained in this research.

- Data Cleaning: Inappropriate data can lead to poor model performance. Therefore, in this research data cleaning stage was carried out in order to improve model performance. Data cleaning steps usually include handling empty data, irrelevant data, inappropriate data formats, and outlier data. All of these steps are used according to the problems in the dataset. The dataset used in this research has a complete, and relevant features, but there are data outliers.

- Scaling Feature: The process of standardizing the scale of data for each feature. This is achieved by adjusting the value range for each feature in the dataset. The main purpose of feature scaling is to prevent the dominant influence of features with a larger value range in the data learning process. In standardizing data ranges, several methods can be used, while in this research, the method using z-score is used. This method standardizes the data scale by calculating the difference between the average future values, which is then divided by the standard deviation . It is suitable for use on datasets with outlier data. The equation (1) is used to calculate the z-score, which measures how many standard deviations a particular data point (x) is away from the mean (μ) of the dataset.

$$z = \frac{x - \mu}{\sigma} \qquad (1)$$

Where:
Z = z-score
μ = Average
σ = Std. deviation

- Data Partitioning: The step of separating data that will be used for train and test. This is done so that the results of measuring the performance of machine learning model can represent reliable results, since the data that being tested is the data that has never been seen by machine learning model before. In this research, data was divided into 70% train data and 30% test. This 70-30 split was chosen because it provides a balanced approach that allows for sufficient data to train the model while still leaving a substantial portion for testing its performance. This ratio is often preferred when there is a moderate amount of data available, as it provides a good balance between training accuracy and testing reliability. While common alternatives, such as the 80-20 split, also work well, the 70-30 split is generally considered more conservative and helps avoid overfitting, ensuring that the model has enough data for both training and validation. Furthermore, it helps in ensuring that the model is tested on a diverse set of examples, leading to a more robust evaluation of its generalization ability.

- Data Class Balancing: In the case of classification, imbalanced data classes can cause the machine learning algorithm to learn more from a larger amount of data. This causes errors in the learning process by the machine learning algorithm which causes a degradation in model performance. In this research, data class balancing was used with an over sampling method, SMOTE (Synthetic Minority Over Sampling Technique) algorithm. This method balances data classes by adding data in classes that have a smaller number so that it has a balanced number of data classes with a larger number [18]. In adding data to the minority class, this method uses data grouping based on nearest neighbors which is calculated using eucledean distance [19].

$$x_{syn} = x_i + (x_{knn} - x_i) \; x \; \beta, i = 1,2,....,n \qquad (2)$$

Where :

x_{syn} : Synthesized data

x_i : Data to be synthesized

x_{knn} : Data with the closest distance to the data to be synthesized

β : Random number (between 0-1)

2.2.4. Modeling

In machine learning, modeling refers to the process of building mathematical equation to represent the phenomenon based on data that has been provided using machine learning algorithm. This model is then used to carry out predictions, classification, optimization, etc. The selection of machine learning algorithm in model building really depends on the objectives and dataset used. Choosing the wrong algorithm can lead to poor model performance. In this research, an artificial neural network algorithm was used as an algorithm which is functioned to classify the condition of heavy equipment machines.

Artificial neural network algorithms are part of deep learning whose way of working is inspired by the way of human brain works [20]. Brain is an human's organ consisting of a collection of interconnected nerves that can be used to learn using examples. Likewise artificial neural networks, this algorithm can also learn using examples of data provided and has interconnected 'neurals' which are represented using mathematical calculations.
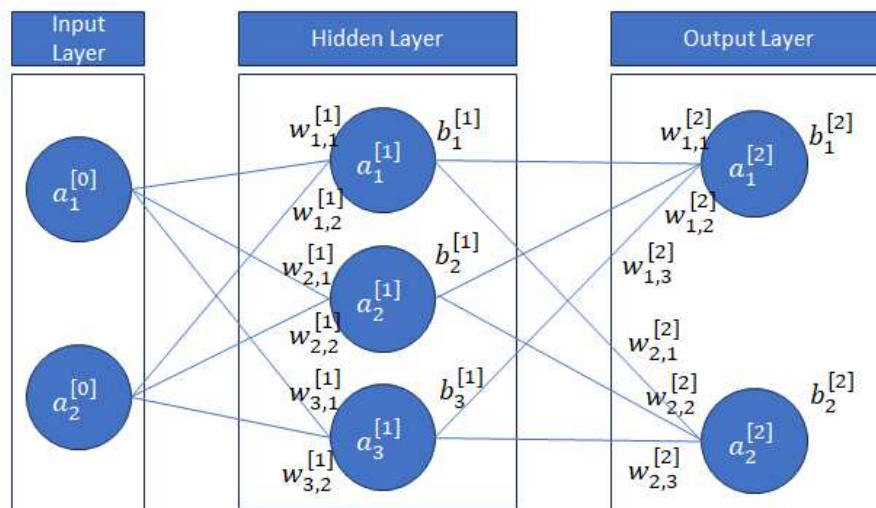

Figure 3. Architecture Neural Network

The human brain nerves that were the inspiration for the creation of this algorithm are interconnected sending elements called 'neurons'. Neurons in artificial neural networks are interconnected elements whose function is to transmit information through synapses (electromagnetic connections) which are then used to solve problems. The connections between neurons then form layers. In each connection, each neuron has its own weight. Weight represents the influence or strength of connections between neurons [21]. In artificial neural network algorithms, there are elements called 'parameters' and 'hyperparameters'. Where parameters are variable configurations used to influence and control the artificial neural networks behavior and performance. Meanwhile, hyperparameters are part of the parameters that are usually determined before the data training process takes place [22]. The learning process in artificial neural networks involves two main stages: feedforward and backpropagation. In the feedforward stage, data flows through the network, starting from the input layer, passing through the hidden layers, and finally reaching the output layer [23]. During this process, each neuron performs mathematical operations by multiplying the inputs with corresponding weights, adding biases, and applying an activation function to generate an output. This forward pass predicts an output based on the current parameter settings. However, the predicted output is rarely perfect in the initial stages. To

address this, the backpropagation process is utilized to minimize errors. Backpropagation adjusts the weights and biases of the network based on the difference between the predicted output and the actual target, as measured by a loss function. This adjustment is achieved by propagating the error backward through the network, layer by layer, using the gradient of the loss function with respect to each weight. Optimizers, are then used to update the parameters to minimize the loss iteratively.

In this research, comparison was made of two hyperparameters in the artificial neural network algorithm, including the number of epochs and the type of optimizer. In deep learning, epoch is a representation of the iterations of the training dataset. In each epoch, the parameters (weights and bias) are updated by algorithm. An optimal number of epochs required to produce good model performance, depending on complexity, dataset size, and the type of task the model is performing. Too few epochs can lead to poor model performance. This happens because the model does not have enough learning iterations to be able to adjust the weight and bias parameters properly. On the other hand, too many epochs can lead the model to adapt 'noise' contained in the training data, which then lead to an over fitting. This comparison is carried out to determine the number of epochs that is most suitable for handling the dataset used. So, no under-fitting or over-fitting model performance is produced. Meanwhile, hyperparameter optimizer in this algorithm is an algorithm used to adjust parameter values (weights and bias). Optimizer helps the model to learn from training data by measuring the weight and bias values with the aim of reducing loss function [24]. The choice of optimizer type is based on research case. This plays an important role in the model performance results. There are several types of optimizers, one of which is adam and adamax. The performance of this second optimizer will be explained in this research.

2.2.3.1. Adaptive Moment Estimation (Adam) Optimizer

Adam optimizer applies adaptive learning rate calculations to each parameter and performs calculations on individual adaptive learning rates on different parameters to estimate the first and second moments of the gradient . This algorithm is a combination of two algorithms, Adaptive Gradient (Adagrad) and Root Mean Square Propagation Algorithm (RMSProp) [25].

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{u}_t + \epsilon}} \hat{a}_t \qquad (3)$$

With :

$$\hat{a}_t = \frac{a_t}{1 - \beta_1^t}$$
$$\hat{u}_t = \frac{u_t}{1 - \beta_2^t}$$

Where :

- $\hat{a}_t$: *Exponentially decaying average of previous gradients.*
- $\hat{u}_t$: *Exponentially decaying average from the square of the previous gradient.*

2.2.3.2. Adamax Optimizer

The basic principle of this algorithm adapt the learning rate adaptively during training to improve model convergence. This algorithm has the advantage of dealing with varied gradient problems. Apart from that, the advantage of Adamax also lies in the method of improving the learning rate during training. The difference between the adam and adamax algorithms lies in the use of the infinite norm of the moving average of the squared gradient parameters [26].

$$\theta_{t+1} = \theta_t - \frac{\eta}{v_t} \hat{a}_t \qquad (4)$$

Where :

- v_t : *Infinity norm*

3. RESULTS AND DISCUSSIONS

Final stage of this research is data evaluation stage which aims to test and measure the results of classification carried out by the machine [27]. In evaluating results, this research used an accuracy metric and a loss function. This aims to determine the model's ability to predict data classes correctly. Equation (5) is a formula for calculating accuracy.

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \; x \; 100\% \tag{5}$$

Where :
• TP   : True positive
• TN   : True negative
• FP   : False positive
• FN   : False negative

In artificial neural networks algorithm, the calculation of loss function is a method for evaluating model's ability to predict data correctly. This function calculates the error in each training result. Loss value represents the extent to which the algorithm's prediction results are compared with the actual value. In artificial neural networks, there are various types of loss functions. The selection of this function depends on the research case, such as in the cases of regression and classification, different loss functions are used. In this case, the loss function used was binary cross entropy loss or commonly known as the logarithmic loss function. Since the binary cross entropy is a loss function used in classification cases with two class categories.

3.1 Experimental Results using Adam and Adamax Optimizer with 100 Epochs

Table 2 is the result of loss function calculations using binary cross entropy and accuracy at each multiple of 10 epochs. Figure 3 is a graph that represents the overall results of the loss function calculation and the accuracy value of 100 epochs.

Table 2 Loss and Accuracy Values using the Optimizer at Epoch 1-100

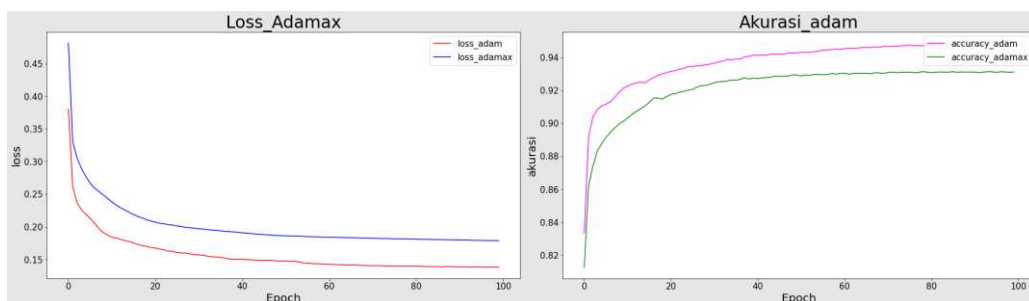| Epoch	Adam Optimizer		Adamax Optimizer	
	Loss	Accuracy	Loss	Accuracy
1	0.3799	0.8333	0.4813	0.8127
10	0.1875	0.9212	0.2435	0.9012
20	0.1682	0.9036	0.2088	0.9162
30	0.1576	0.9357	0.1978	0.9236
40	0.1500	0.9414	0.1915	0.9273
50	0.1475	0.9427	0.1864	0.9292
60	0.1427	0.9448	0.1841	0.9301
70	0.1403	0.9466	0.1827	0.9302
80	0.1399	0.9473	0.1811	0.9309
90	0.1385	0.9480	0.1799	0.9310
100	0.1381	0.9482	0.1787	0.9311

Figure 4. Comparison graph of loss and accuracy using Adam and Adamax Optimizer

Lower loss values indicate better model performance, demonstrating convergence. This is because when calculating the loss function, where this research uses binary cross entropy loss, a

higher loss value indicates that the prediction results produced by the model are farther from the actual value. Equation (6) represents the calculation of binary cross entropy as a loss function.

$$Loss = -\frac{1}{N}\sum_{i=1}^{N}[y.\log(\hat{y}) + (1 - y)\log(1 - \hat{y})] \qquad (6)$$

Where :
- y = Actual value
- $\hat{y}$ = Prediction result value

There are two parts of equation (6). The first part is a calculation that indicates errors in classifying values of positive class. In equation (7), the loss calculation is obtained from multiplying the actual and logarithmic values of the machine's predicted values. In this equation, the smaller loss value is obtained from the prediction value which is closer to the positive class (or in this case closer to the data class with binary 1). On the other hand, if the prediction result is close to 0 (negative class), the resulting loss value will be greater. This indicates the accuracy of predictions in the positive class.

$$-(y \times \log(\hat{y})) \qquad (7)$$

In contrast to equation (7), equation (8) is a loss value calculation that focuses on the negative class of the dataset. In equation (8), the smaller loss value is obtained from the machine prediction value which is getting closer to the negative class (or in this case closer to the data class with binary 0). Vice versa, the greater the loss value is obtained from the prediction value which is closer to 1 (positive class). In this way, calculation results are obtained that indicate the model's ability to predict the negative class.

$$-(1 - y) \times (\log(1 - \hat{y})) \qquad (8)$$

Apart from loss calculations, accuracy calculations are also used in model evaluation in this research. This is because accuracy calculations indicate the model's ability to classify data into the correct data class groups. It can be seen in Figure 3 that the accuracy value is still moving up at 100 epochs. This indicates that the machine is still learning well at that epoch. From the results of training data using deep learning algorithm, it is evident that the epoch value is inversely proportional to the loss calculation results. However, this is not the case for accuracy value, which shows a directly proportional relationship to the number fo epochs. Table 2 illustrates that in the initial epoch, the loss value is high and the accuracy calculation value is low. This is because in the first iteration, the model has not yet effectively learned the data, indicating that the number of epoch is insufficient for the model to make accurate predictions.

In figure 3 can also be observed that the performance model with adam and adamax optimizer. The comparative analysis between these optimizers yielded interesting results. In this case, Adam optimizer outperformed adamax with a higher accuracy rate. This results suggest that Adam optimizer is more suitable for the machine efficiency classification problem in the dataset used. The reasons can vary, but one of the main factors may related to Adam's ability to adaptively adjust the learning rate based on the first and second moments of the gradient. Thus, Adam can be more efficient in finding optimal parameters to minimize the loss function.

4. CONCLUSION

An approach using deep learning algorithms aimed at improving model performance by comparing optimizer and epoch hyperparameters is proposed in this study. The evaluation of model performance comparing Adam and Adamax optimizers showed that the model using the

Adam optimizer achieved higher accuracy and lower loss. This suggests that the Adam optimizer performed better for this dataset, which was used for classifying machine efficiency. Additionally, adjusting hyperparameters such as the number of epochs had a significant impact on the final model outcome, as indicated by the increase in accuracy with more epochs. Therefore, selecting the right optimizer and determining the optimal number of epochs are crucial for improving model performance.

For future work, it is recommended to explore additional optimizers, such as RMSprop and Adagrad, to assess their effectiveness in improving model performance. Furthermore, investigating more complex combinations of hyperparameters, including learning rate schedules and batch sizes, could provide further insights into optimizing deep learning models for machine efficiency classification. This research serves as a guide for researchers and practitioners to make more informed decisions in selecting and fine-tuning hyperparameters when developing deep learning models in similar contexts.

REFERENCES

[1]   A. B. Widodo and M. Mahagiyani, "Analisis kebangkrutan dan mitigasi risiko pada perusahaan perkebunan," Jurnal Pengelolaan Perkebunan (JPP), vol. 3, no. 1, Apr. 2022, doi: 10.54387/jpp.v3i1.13.

[2]   L. Belakang, "Ringkasan Kebijakan: Industri Kelapa Sawit, Penanggulangan Kemiskinan dan Ketimpangan 1", 2019.

[3]   Z. Abidin, "Accuppational Safety And Health Management System For Zero Accidents In Palm Oil Plantation Companies In West Borneo", JOSS : Journal of Social Science, 2023, [Online]. Available: https://joss.al-makkipublisher.com/index.php/js.

[4]   M. R. Ahmad et al., "Mechanisation Technology for Small-Scale Oil Palm Plantations," Advances in Agricultural and Food Research Journal, vol. 4, no. 2, Dec. 2023, doi: 10.36877/aafrj.a0000406.

[5]   F. Raza, "AI for Predictive Maintenance in Industrial Systems", Cosmis Bulletin of Business Management, vol.2, no.1, 2023, doi: 10.13140/RG.2.2.27313.35688.

[6]   Jubin Thomas, Piyush Patidar, Kirti Vinod Vedi, and Sandeep Gupta, "An analysis of predictive maintenance strategies in supply chain management," International Journal of Science and Research Archive, vol. 6, no. 1, pp. 308–317, Jun. 2022, doi: 10.30574/ijsra.2022.6.1.0144.

[7]   K. Y. Nazara, "Perancangan Smart Predictive Maintenance untuk Mesin Produksi (Smart Predictive Maintenance Design for Production Machines)", Seminar Nasional Official Statistics, 2022, doi: https://doi.org/10.34123/semnasoffstat.v2022i1.1575

[8]   Y. Bouabdallaoui, Z. Lafhaj, P. Yim, L. Ducoulombier, and B. Bennadji, "Predictive maintenance in building facilities: A machine learning-based approach," Sensors (Switzerland), vol. 21, no. 4, pp. 1–15, Feb. 2021, doi: 10.3390/s21041044.

[9]   J. A. Abbasi, "Predictive Maintenance in Industrial Machinery using Machine Learning", 2021.

[10]   M. I. Ali, N. S. Lai, and R. Abdulla, "Predictive maintenance of rotational machinery using deep learning," International Journal of Electrical and Computer Engineering, vol. 14, no. 1, pp. 1112–1121, Feb. 2024, doi: 10.11591/ijece.v14i1.pp1112-1121.

[11]   H. Dehghan Shoorkand, M. Nourelfath, and A. Hajji, "A hybrid deep learning approach to integrate predictive maintenance and production planning for multi-state systems," J Manuf Syst, vol. 74, pp. 397–410, Jun. 2024, doi: 10.1016/j.jmsy.2024.04.005.

[12] J. F. Andry, H. Tannady, G. D. Rembulan, and D. F. Dinata, "Analysis of the Omicron virus cases using data mining methods in rapid miner applications," Microbes and Infectious Diseases, vol. 4, no. 2, pp. 323–334, May 2023, doi: 10.21608/MID.2023.194619.1469.

[13] A. Chakir, M. Chergui, and J. F. Andry, "A Smart Updater IT Governance Platform Based on Artificial Intelligence," Advances in Science, Technology and Engineering Systems, vol. 5, no. 5, pp. 47–53, 2020, doi: 10.25046/aj050507.

[14] M. Stefanus, J. Fernandes Andry, F. Teknologi, D. Desain, and U. Bunda Mulia, "Pengembangan Aplikasi E-Learning Berbasis Web Menggunakan Model Waterfall Pada Smk Strada 2 Jakarta", Jurnal Fasilkom, vol.10, no.1, 2020, doi: https://doi.org/10.37859/jf.v10i1.1878

[15] H. P. Josyula, "A Review On Data Visualization For Exploratory Data Analysis", Journal of Data Acquisition and Processing, vol.38, 2023, doi: 10.5281/zenodo.777730.

[16] F. Rodríguez-Torres, J. F. Martínez-Trinidad, and J. A. Carrasco-Ochoa, "An Oversampling Method for Class Imbalance Problems on Large Datasets," Applied Sciences (Switzerland), vol. 12, no. 7, Apr. 2022, doi: 10.3390/app12073424.

[17] E. Wahyu Pujiharto and A. Nasiri, "Analisis Perbandingan Kinerja Algoritma Apriori, FP-Growth dan Eclat dalam menemukan Pola Frekuensi pada Dataset INA-CBG'S Comparative Analysis of The Performance of The Apriori, FP-Growth and Eclat Algorithms In Finding Frequency Patterns In The Ina-Cbg's Dataset," Cogito Smart Journal |, vol. 9, no. 2, 2023.

[18] I. R. Yunita, W. Maulana Baihaqi, A. Shafira, T. Damayanti, and L. Akhaerunnisa, "Analisis Performa Algoritma Klasifikasi pada Sentimen Ulasan Pengguna terhadap Aplikasi Muamalat DIN Analysis of Classification Algorithm Performance on User Review Sentiment of the Muamalat DIN Application," Cogito Smart Journal |, vol. 9, no. 2, 2023.

[19] N. M. Djafar and A. Fauzan, "Implementation of K-Nearest Neighbor using the oversampling technique on mixed data for the classification of household welfare status," Statistics in Transition new series, vol. 25, no. 1, pp. 109–124, Mar. 2024, doi: 10.59170/stattrans-2024-007.

[20] S. Schmidgall, R. Ziaei, J. Achterberg, L. Kirsch, S. P. Hajiseyedrazi, and J. Eshraghian, "Brain-inspired learning in artificial neural networks: A review," APL Machine Learning, vol. 2, no. 2, Jun. 2024, doi: 10.1063/5.0186054.

[21] R. Dastres and M. Soori, "Artificial Neural Network Systems," 2021. [Online]. Available: https://www.researchgate.net/publication/350486076

[22] S. Afaq and S. Rao, "Significance Of Epochs On Training A Neural Network", International Journal of Scientific & Technology Research, vol.9, issue 6, 2020, [Online]. Available: www.ijstr.org

[23] L. Alzubaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," J Big Data, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00444-8.

[24] F. Mehmood, S. Ahmad, and T. K. Whangbo, "An Efficient Optimization Technique for Training Deep Neural Networks," Mathematics, vol. 11, no. 6, Mar. 2023, doi: 10.3390/math11061360.

[25] I. A. Dewi and M. A. N. E. Salawangi, "High performance of optimizers in deep learning for cloth patterns detection," IAES International Journal of Artificial

.

Intelligence, vol. 12, no. 3, pp. 1407–1418, Sep. 2023, doi: 10.11591/ijai.v12.i3.pp1407-1418.

[26]   D. Soydaner, "A Comparison of Optimization Algorithms for Deep Learning," Jul. 2020, doi: 10.1142/S0218001420520138.

[27]   J. M. Waworundeng et al., "Sentiment Analysis of Online Lectures Tweets using Naïve Bayes Classifier Analisis Sentimen Tweet Kuliah Online menggunakan Naïve Bayes Classifier," Cogito Smart Journal |, vol. 8, no. 2, p. 2022.