International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 49 No. 1 February 2025, pp. 158-169

Model Development and Optimization of Data Security Attack Detection Using Neural Network Technology

Devi Tiana Octaviani Supriyadi[1], Bisyron Wahyudi[2], Danang Rimbawa[3], Rayasa Puringgar Prasadha Putra[4]

[1234]Cyber Defense Engineering, Faculty of Defense Science and Technology,
Universitas Pertahanan, Bogor, Indonesia
[1] devi.supriyadi@tp.idu.ac.id
[2] bisyron.wahyudi@idu.ac.id
[3] hadr71@idu.ac.id
[4] rayasa.putra@tp.idu.ac.id

Abstract—This research examines the development and optimization of a data security attack detection model utilizing an artificial neural network (ANN) approach. The primary objective is to enhance the robustness and effectiveness of data security systems in addressing the growing complexity of cyber threats. By leveraging state-of-the-art ANN technologies, this research seeks to improve the efficiency and accuracy of detecting various types of security attacks, including Malware, DDoS, and Intrusion, in dynamic network environments. The methodology involves an in-depth analysis of comprehensive security attack datasets, the application of advanced optimization techniques, and the implementation of cutting-edge ANN models. Additionally, this study integrates machine learning methods such as Convolutional Neural Networks (CNN), Random Forest (RF), and Support Vector Machines (SVM) to evaluate and compare their performance in threat detection. Through a rigorous analysis, the strengths and limitations of each model are assessed to identify the most effective approach for classifying and mitigating security threats. The findings underscore the potential of ANN technologies in ensuring data integrity and resilience against increasingly sophisticated cyberattacks. However, challenges such as dataset imbalances and model biases necessitate further refinement. This research contributes to the advancement of cybersecurity by providing critical insights into the application of ANN technologies and fostering innovative strategies for developing adaptive and reliable security solutions to meet the demands of the evolving digital landscape.

Keywords—Data Security Attack Detection, Artificial Neural Network, Data Security Optimization, Cyber Threats.

I. INTRODUCTION

Data security is a major concern in today's connected digital age. With the increasing complexity of cyberattacks, sophisticated and innovative approaches are needed to detect and mitigate these threats. One promising approach is the use of artificial neural network technology, which is capable of understanding complex patterns in data and providing adaptive solutions. This research aims to develop a data security attack detection model using artificial neural networks which focuses on data security optimization. By integrating the latest technologies, this research is geared towards improving the efficiency and accuracy of security attack detection, thereby strengthening defenses against evolving cyber threats. Through in-depth analysis of relevant security attack datasets and implementation of innovative optimization strategies, it is expected that the results of this research can make significant contributions to the development of adaptive and effective data security solutions. With a deeper understanding of the

role of neural network technology, it is expected to open up new avenues in protecting data integrity in the dynamic and challenging digital era.

From previous research, there are many uses of artificial neural networks to secure data or analyze a cyber attack that can harm users or victims. So this research will discuss Model Development and Optimization of Data Security Attack Detection Using Neural Network Technology which will take a dataset about cyber attacks that occur and will analyze the attack using Convolutional Neural Network (CNN), Random Forest and Support Vector Machine (SVM). In general, this research will conduct experiments:

1. Main Program (Neural Network), Data Load and Preprocessing, Data is loaded from CSV files containing information about security attacks that will be analyzed first and split Classification report includes precision, recall, and f1-score for each category.
2. The Random Forest model is used to compare the results with the neural network model and involves creating a RandomForestClassifier instance to train it with the training data.
3. The SVM model is also used to compare the results and will also analyze with previous results.
4. CNNs are evaluated using validation data, The results of the CNN model evaluation are printed and the accuracy is calculated.
5. The final results of the CNN model evaluation are printed and include classification and accuracy reports, CNN is then compared with Neural Network, Random Forest, and SVM models in a visualization using pie charts.

The structure of the paper, as shown below. Section II reviews related work. Section III introduces the datasets for experiments, data preprocessing methods, CNN, Random Forest and SVM, the models proposed in this paper. Section IV makes a comprehensive analysis of the experimental results. Finally, the paper is summarized and projected.

II. RELATED WORK

Shi Dong, el.at in his research on Network Abnormal Traffic Detection Model Based on Semi-Supervised Deep Reinforcement Learning said This paper proposes a semi-supervised Double Deep Q-Network (SSDDQN) based optimization method for network abnormal traffic detection, mainly based on Double Deep Q-Network (DDQN), a representative of Deep Reinforcement Learning algorithm. In SSDDQN, the current network first adopts an autoencoder to reconstruct traffic features and then uses a deep neural network as a classifier. The target network first uses unsupervised learning K-Means clustering algorithm and then uses deep neural prediction network. This experiment uses NSL-KDD and AWID for training and testing and conducts a comprehensive comparison with existing machine learning models. Experimental The experimental results show that SSDDQN has certain advantages in time complexity and achieves good results in various evaluations metrik.[1]

In addition, there is also research by Yan Sin et.al on Machine Learning and Deep Learning Methods for Cybersecurity where the results of his research reveal that describes a survey of the main literature on machine learning (ML) and deep learning (DL) methods for intrusion detection network analysis and provides a brief tutorial brief description of each ML / DL method. Papers representing each method are indexed, read, and summarized based on their temporal or thermal correlation. Since data is critical in ML/DL methods, we describe some common network datasets used in ML/DL, discuss the challenges of using ML/DL for cybersecurity and provide suggestions for research directions.[2]

In addition, there is also research from Arif, Imam and Sunardi on DDoS Attack Detection Using Neural Network with Fixed Moving Average Window Function where the results of research on testing the neural network method with the fixed moving average window (FMAW) function produce an average percentage of recognition of three network conditions (normal, slow DDoS, and DDoS) of 90.52%. The existence of a new approach in detecting DDoS attacks is expected to be a complement to the IDS system in predicting the occurrence of DDoS attacks.[3].

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 49 No. 1 February 2025, pp. 158-169

III. WORK DESCRIPTION

This section presents our work in the following aspects: (1) In Section III-B, the related theories of neural network technology, RF, SVM, and CNN are briefly explained. (3) In Section III-C, the dataset preprocessing methods of neural network technology, RF, SVM, and CNN are introduced. (4) The technology models of neural network, RF, SVM, and CNN are described in detail in Section III-D.

A. Datasets

This work considers datasets that fulfill the following aspects: (1) Public and well-known datasets with many researchers using such datasets for experiments. (2) Datasets that are highly imbalanced datasets with varying degrees of imbalance closer to the actual network environment. (3) Datasets that are divided into training and testing sets. (4) The dataset contains enough samples to facilitate training and testing. Therefore, this paper selects the cyberattack dataset as the main dataset for experiments. The experimental results are biased, with higher accuracy generally biased towards more frequent records.

The cyberattacks dataset contains 25 varied metrics and 40,000 records, which contain 3 types of attacks namely malware, DDoS and Intrusion. This will be the study of Model Development and Optimisation of Attack Detection with Neural Network Technology.

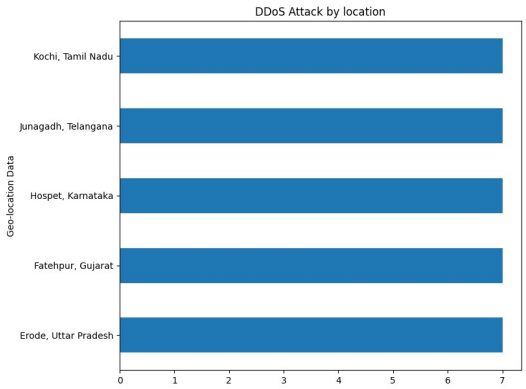

Fig. 1. DDoS Attack by location

DoS: Attackers occupy multiple locations used to conduct DDoS attacks, destroying other users' regular access to resources such as networks and servers so that normal users cannot get service.

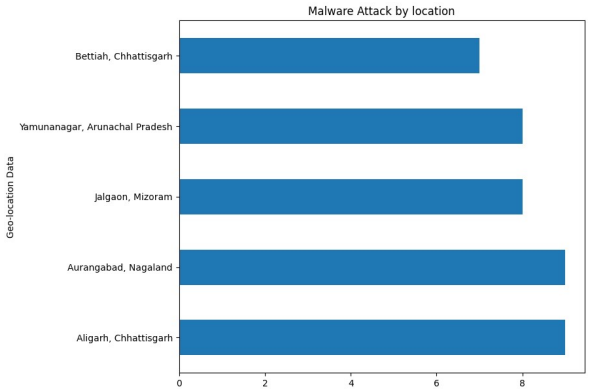

Fig. 2. Malware Attack by location

In the malware image there are also attacks that occurred in several locations.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 49 No. 1 February 2025, pp. 158-169

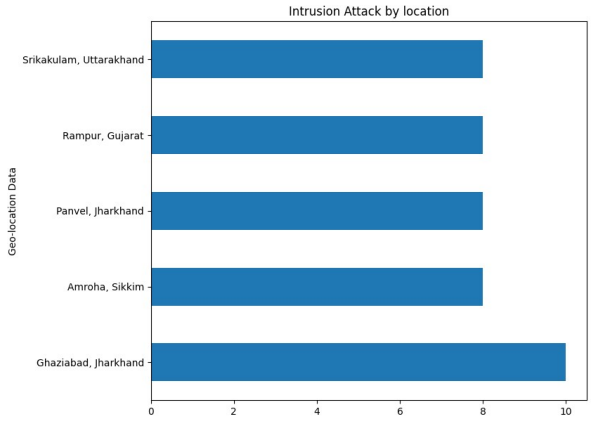

Fig. 3. Intrusion Attack by location

In the intrusion image, it can also be seen that the attack location is not only in one place but many locations that carry out the attack.

B. Model Description

The Neural Network (NN) model is a simple artificial neural network consisting of three layers. The input layer has neurons for the number of features after preprocessing. The two hidden layers use ReLU activation with 64 and 32 neurons respectively. The output layer uses a softmax function with the number of neurons corresponding to the number of categories (Attack Type). In addition, there is Training and Evaluation where the NN Model is trained using training data with validation techniques using validation data. The loss function used is *'sparse_categorical_crossentropy'*, and the optimizer used is Adam. The performance of the model was evaluated using test data by printing classification and accuracy reports. where there is the following equation;

Input Layer: This layer receives input in the form of vectors $x^{(0)}$ which contains the element $x_1, x_2, \ldots, x_{n_{input}}$

Hidden Layer 1: In this layer $Z^{(1)} = W^{(1)}.X^{(0)} + b^{(1)}$ Here, $W^{(1)}$ is the weight matrix for the first layer, $x^{(0)}$ is the input vector, and $b^{(1)}$ is the bias for the first layer. This operation produces a vector $Z^{(1)}$.

Hidden Layer 2: The same process is repeated on this layer. $A^{(1)}$. Multiplied by the weight matrix $W^{(2)}$ and added with bias $b^{(2)}$ to produce $Z^{(2)}$. ReLU activation function applied to $Z^{(2)}$ to produce $A^{(2)}$.

Output Layer: In this layer, $A^{(2)}$. Multiplied by the weight matrix $W^{(3)}$. and added with bias $b^{(3)}$. to produce $Z^{(3)}$. Softmax activation function applied to $Z^{(3)}$. to produce $A^{(3)}$ which is the output of the model.

The Convolutional Neural Network (CNN) model is an artificial neural network specifically designed to handle spatial or grid data such as images. This model starts with the Reshape layer to adjust the input dimensions. This is followed by the Conv1D layer with 64 filters and ReLU activation. Then there is a Flatten layer to convert the data into vectors. Finally, the output layer with softmax function. Training and Evaluation: CNN models are trained using training data with validation techniques using validation data. Model performance is evaluated using test data by printing classification and accuracy reports. The convolution operation in CNN can be represented as follows:

$$FM[i]_{j,k} = \left(\Sigma_m \Sigma_n N_{[j-m,k-n]} F_{[m,n]}\right) + bF \quad (1)$$

Where:

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 49 No. 1 February 2025, pp. 158-169

FM[i] : The i-th Feature Map Matrix

N : Input image matrix

F : Convolution filter matrix

bF : The bias value of the filter

j,k : Pixel position in the input image matrix

m,n : Pixel position in the convolution filter matrix

Where in this programming can be made with the following equation:

Input Layer: The input vector $x^{(0)}$ is defined as $x^{(0)} = [x_1, x_2, ..., x_{n\ input}]$

Reshape Layer: The input vector $x^{(0)}$ is reshaped into a matrix $x^{(1)}$ with dimensions $(n_{features}, 1)$. This is represented as $x^{(1)}$=Reshape($x^{(0)}$,($n_{features}$, 1)).

Conv1D Layer: A 1D convolution is performed on the reshaped input $x^{(1)}$ using the weight matrix $x^{(2)}$and bias vector $b^{(2)}$ The output of this layer is $Z^{(2)}$= Conv1D($x^{(1)}$, $x^{(2)}$, $b^{(2)}$).

$$z_{in_i} = \sum_{i=1}^{n} X_j * V_{j,1} + V_{0,i} \quad (2)$$

Where:

z_in_i : input for the i-th hidden layer node with the number of nodes n

X_j : jth X node

$V_{j,1}$: weight V fot X_j and node Z_i

V_0 : V bias for this node

ReLU Activation: The ReLU (Rectified Linear Unit) activation function is applied to $Z^{(2)}$ to obtain $A^{(2)}$ReLU($Z^{(2)}$).

Flatten Layer: The output of the ReLU activation $A^{(2)}$ is flattened into a vector $X^{(3)}$=Flatten($A^{(2)}$)).

Output Layer: The output layer computes $Z^{(4)} = W^{(4)} . X^{(3)} + b^{(4)}$.

$$y\_in_i = \sum_{i=1}^{m} Z_j * W_{j,i} + W_{o,i} \quad (3)$$

Where:

y_in_i : Input for the Z-th hidden layer node with the number of nodes m

Z_j : jth Z node

$W_{j,i}$: W weights for Z_j and Y_i

W_0 : W bias for this

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 49 No. 1 February 2025, pp. 158-169

The Softmax activation function is then applied to $Z^{(4)}$ to obtain the final output $A^{(4)}$=Softmax($Z^{(4)}$).

$$Y_i = \frac{e^{y\_in_i}}{\sum_{i=1}^{m} e^{M}} \tag{4}$$

Where:

Y_i : output for the i-th output layer

y_in_i : input for the i-th layer mode

M : all m inputs to the output layer

This sequence of operations represents a simple Convolutional Neural Network (CNN) with a single convolutional layer followed by a fully connected output layer. The Softmax activation function ensures that the output values are probabilities that sum to 1, making them suitable for classification tasks. The ReLU activation function in the convolutional layer introduces non-linearity into the model, allowing it to learn more complex patterns in the input data. The Flatten layer is necessary to reshape the output of the convolutional layer into a format suitable for the fully connected output layer.

In RF, the prediction for a data point is usually calculated by taking the average of the predictions of all the trees in the forest. If we have N trees and pi is the prediction of the i-th tree, then the RF prediction is:

$$p = \frac{1}{N}\sum_{i=1}^{N} p_1 \tag{5}$$

Support Vector Machine (SVM) Model Description The SVM model is a classification model that constructs decision boundariesbased on the support of a vector of training data using a 'linear' kernel to linearly separate the data. Training and Evaluation The SVM model is trained using training data after preprocessing. Model performance was evaluated using test data by scoring accuracy. The equation that can be used is:

$$f(x) = \text{sign}(\mathbf{w} \cdot \mathbf{x} - b) \tag{6}$$

Linear Transformation: The input vector $x^{(0)}$ is transformed by the weight matrix $W^{(2)}$ and bias vector $b^{(2)}$ to produce $Z^{(2)}$. This is represented as $Z^{(2)}$. =$W^{(2)}$. $X^{(0)} + b^{(2)}$. This operation is a linear transformation where $W^{(2)}$. and $b^{(2)}$ are the parameters that the model will learn.

$$w \cdot x_i + b \geq 1 \; if \; y_i = 1 \tag{7}$$

$$w \cdot x_i + b \leq -1 \; if \; y_i = -1 \tag{8}$$

Softmax Activation: The Softmax function is applied to $Z^{(2)}$. To obtain $A^{(2)}$. = Softmax($Z^{(2)}$) The Softmax function transforms its inputs, known as logits or log-odds, into a probability distribution that sums to 1. This makes it suitable for multi-class classification problems, where each class's probability needs to be determined.

$$L_p = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{N} \lambda_i (y_i (w \cdot x_i + b) - 1) \tag{9}$$

These operations represent a fully connected layer (also known as a dense layer) followed by a Softmax activation, which is commonly found in neural networks designed for classification tasks. The parameters $W^{(2)}$. and $b^{(2)}$ are learned during the

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 49 No. 1 February 2025, pp. 158-169

training process to minimize the difference between the model's predictions and the actual labels. The Softmax activation ensures that the output values can be interpreted as probabilities.

C. Dataset Preprocessing

Data Pre-processing: In the data pre-processing stage, preliminary steps are taken to ensure the quality and sustainability of the process. The dataset is obtained from its source at "/content/gdrive/MyDrive/AI/MAI2/cybersecurity_attacks.csv", and features and targets are separated for further analysis. The target, which in this case is the 'Attack Type' column, is converted into a numeric format using LabelEncoder. Next, the data was divided into training, validation, and test sets. This division allows for an objective evaluation of the model's performance.

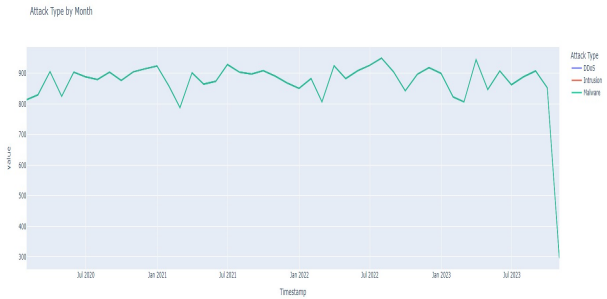

Fig. 4. Attack Type by Month

Numerical and Categorical Pre-processing: Numerical features in the dataset were normalized using StandardScaler, while categorical features were processed with one-hot encoding using OneHotEncoder. Column Transformer was used to combine the results of numerical and categorical pre-processing, creating data that was ready to be trained by the model.

Neural Network (NN) Model: The NN model is implemented with one hidden layer using ReLU activation function and an output layer with softmax activation function. This structure is designed to capture complex patterns in data.

Convolutional Neural Network (CNN) model: CNN has a different architecture by adding a Reshape layer for spatial data preparation. This is followed by a Conv1D layer that uses ReLU activation and is followed by a Flatten layer to transform the data into a format that can be further processed by the output layer.

Random Forest (RF) Model: The RF model is an ensemble model consisting of multiple decision trees. It provides reliability and robustness on complex data with a combination of results from multiple decision models.

Support Vector Machine (SVM) model: SVM utilizes a linear kernel and softmax function in the output layer. This design is suitable for handling classification on data that requires clear boundaries between classes. Result Analysis and Visualization: The performance of each model was evaluated and compared on the test dataset. Graphs and tables are used to provide a clear visual representation of the accuracy and performance of the models.

Model Use Recommendations: Based on the analysis results, recommendations are given for the use of specific models depending on the nature and characteristics of the data. For example, CNN can be more suitable for data with spatial features, while RF or SVM can be more effective on less complex data that requires precision in classification.

IV. EXPERIMENTS AND RESULTS

This section presents the paper's following aspects: (1) Section IV-A introduces our experimental environment. (2) Section IV-B introduces the experimental evaluation metrics. (3) Section IV-C contains the comparison results of CNN, RF and SVM models for data security.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 49 No. 1 February 2025, pp. 158-169

A. Experiment Environment

The experiments were implemented on a PC platform with Windows 11, Intel(R) Core(TM) i5-10210U @ 2.11GHz CPU processor, 8 GB RAM, and all languages using python. In addition, the scikit-learn library was used in the experiments, and the Tensorflow Library and Keras backend were also used.

B. Evaluation Metric

The general method in the program that has been created, model evaluation uses several common classification performance metrics, namely Precision, Recall, and F1 Score. The following is a brief explanation of each metric:

Precision The percentage of correct positive predictions. Interpretation: How well the model identifies true positives compared to its total positive predictions. Formula:

$$\text{Precision} = \frac{True\ Positives + False\ Positives}{True\ Positives} \quad (10)$$

Recall (Sensitivity) The percentage of the total positive classes successfully identified by the model. Interpretation: How well the model can find all instances of the positive class. Formula:

$$\text{Recall} = \frac{True\ Positives + False\ Negatives}{True\ Positives} \quad (11)$$

F1 Score The harmonic mean of Precision and Recall. Interpretation: A balance between Precision and Recall, providing a holistic picture of the model's performance. Formula:

$$\text{F1 Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (12)$$

C. Results

In this research, a model for detecting data security attacks is developed using various methods, including Artificial Neural Network (NN), Convolutional Neural Network (CNN), Random Forest (RF), and Support Vector Machine (SVM). These methods were implemented after careful data pre-processing, including the use of standardization and one-hot encoding techniques on the data. In the model training stage, it is iterated through training data with a certain number of epochs, and then the model is evaluated using test data to measure its performance. Which can be seen in the results in the table below when the epoch is done 10 times and has results:

TABLE I. Classification Performance Evaluation

	Precision	Recall	F1-Score	Support
DDoS	0.32	0.25	0.28	2636
Intrusion	0.34	0.42	0.37	2721
Malware	0.34	0.32	0.33	2643

The performance evaluation of the classification model is based on analyzing the precision, recall, and F1-score metrics for each category. For the DDoS category, the model has a precision of 0.32, indicating the extent to which the positive predictions associated with this category are actually relevant. Recall for DDoS is 0.25, illustrating the model's ability to identify a large number of actual samples from the DDoS category. The F1-score of 0.28 indicates a balance between precision and recall. For the Intrusion category, a precision of 0.34 indicates a positive prediction accuracy for the Intrusion category. The higher recall of 0.42

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 49 No. 1 February 2025, pp. 158-169

illustrates the model's ability to identify most of the actual samples from the Intrusion category. The F1-score of 0.37 reflects the balance between precision and recall. Meanwhile, in the Malware category, the model has a precision and recall of 0.34 and 0.32, respectively. The F1-score of 0.33 indicates a balance between precision and the model's ability to identify actual samples from the Malware category. Through an in-depth understanding of these metrics, steps can be taken to improve and enhance the model to improve classification performance in each category.

There is also a pie chart of the pie chart with the title "Artificial Neural Network Classification Results (Accuracy: 0.34)".

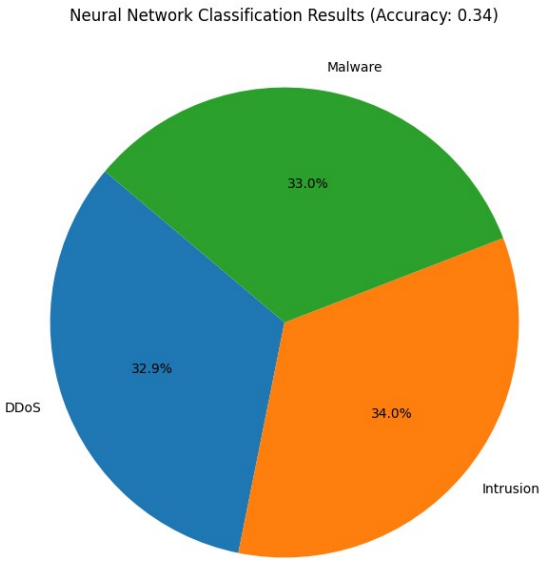

Fig. 5. Chart of Neural Network Classification Results

The chart is divided into three sections representing different types of security threats:

- The green section labeled "Malware" covers 33.0% of the graph.
- The blue section labeled "DDoS" covers 32.9% of the graph.
- The orange section labeled "Intrusion" makes up the largest part with 34.0% of the graph.

This graph shows the classification results of an Artificial Neural Network model against three types of security threats: Malware, DDoS, and Intrusion. The accuracy of this model is 0.34, which indicates that the model is correct in predicting the threat type about 34% of the time.

In addition, there are also accuracy results from CNN

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 49 No. 1 February 2025, pp. 158-169

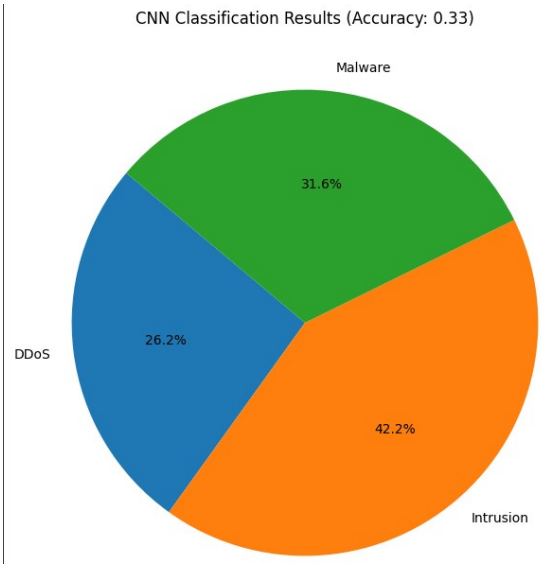

Fig. 6. Chart of CNN Classification Results

The pie chart is the "CNN Classification Result (Accuracy: 0.33)". The graph is divided into three sections representing different types of security threats: The green section labeled "Malware" covers 31.6% of the graph. The blue section labeled "DDoS" covers 26.2% of the graph. The orange section labeled "Intrusion" makes up the largest part with 42.2% of the graph. This graph shows the classification results of a Convolutional Neural Network (CNN) model for three types of security threats: Malware, DDoS, and Intrusion. The accuracy of this model is 0.33, which indicates that the model is correct in predicting the threat type about 33% of the time.

In addition, there is also a circle graph comparison of RF

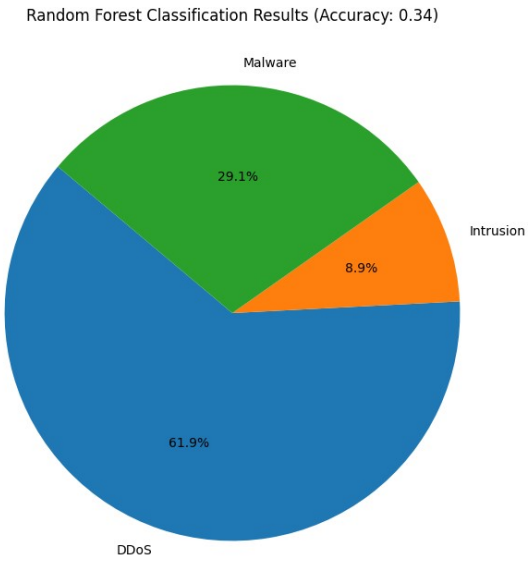

Fig. 7. Chart of Random Forest Classification Results

a pie chart with the title "Random Forest Classification Results (Accuracy: 0.34)". The graph is divided into three sections representing different types of security threats: The green section labeled "Malware" covers 29.1% of the graph. The orange section labeled "Intrusion" covers 8.9% of the graph. The blue section labeled "DDoS" makes up the largest part with 61.9% of

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 49 No. 1 February 2025, pp. 158-169

the graph. This graph shows the classification results of a Random Forest model against three types of security threats: Malware, DDoS, and Intrusion. The accuracy of this model is 0.34, which indicates that the model is correct in predicting the threat type about 34% of the time.

And finally there are measurements from the SVM circle graph

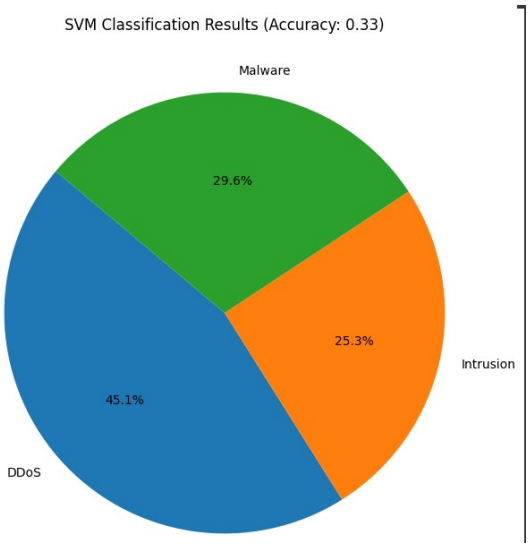

Fig. 8. Chart of SVM Classification Results

a pie chart with the title "SVM Classification Results (Accuracy: 0.33)". The graph is divided into three sections representing different types of security threats: - The blue section labeled "DDoS" covers 45.1% of the graph. This indicates that the SVM model classified about 45.1% of the data as DDoS threats. - The green section labeled "Malware" covers 29.6% of the graph. This indicates that the SVM model classified about 29.6% of the data as Malware threats. - The orange section labeled "Intrusion" covers 25.3% of the graph. This indicates that the SVM model classified about 25.3% of the data as Intrusion threats. This graph shows the classification results of a Support Vector Machine (SVM) model against three types of security threats: Malware, DDoS, and Intrusion. The accuracy of this model is 0.33, which indicates that the model is correct in predicting the threat type about 33% of the time. However, it should be noted that this accuracy may not be sufficient for real-world applications, and the model may need more training or customization. Additionally, the classification distribution shows that the model tends to detect DDoS threats more often than other types of threats. This could be an indication of bias in the training data or in the way the model learns from it. Note that in the context of cybersecurity, it is important to have a model that can detect all types of threats with almost the same accuracy to ensure comprehensive protection.

Model evaluation results are measured using several classification metrics, such as accuracy, precision, recall, and F1 score. These metrics provide a holistic picture of the extent to which the model is able to recognize data security attacks and avoid false predictions. The results are analyzed to understand the strengths and weaknesses of each model. Through this research, a deeper understanding of the effectiveness of these methods in detecting data security attacks is expected. The findings can make important contributions to the development of more efficient and reliable data security solutions.

In this research, a model for detecting data security attacks using Neural Network technology is developed and optimized. Four main models are used, namely Convolutional Neural Network (CNN), Artificial Neural Network, Random Forest (RF), and Support Vector Machine (SVM). Model testing was conducted on three types of data security attacks, namely Malware, DDoS, and Intrusion. The classification results of the four models showed comparable accuracy levels, ranging from 0.33 to 0.34. Although the models' accuracies were comparable, analysis of the classification distributions highlighted differences in emphasis and detection of specific types of attacks. For example, the SVM model is more likely to detect DDoS attacks, while the Random

Forest model has a greater focus on DDoS and less on Intrusion. These distributions provide valuable insights for further improvements to the models or expansion of the datasets to ensure more accurate classification. In conclusion, the use of Artificial Neural Network technology in data security attack detection shows potential, but further consideration is needed in addressing imbalances and increasing the diversity of attack detection.

V. CONCLUSION

From this research, it can be concluded that the use of Artificial Neural Network technology in data security attack detection shows significant potential. Four main models (CNN, Artificial Neural Network, RF, and SVM) were tested and showed comparable accuracy rates, ranging from 0.33 to 0.34. However, analysis of the classification distribution shows that there are differences in the suppression and detection of certain types of attacks. For example, the SVM model is more likely to detect DDoS attacks, while the RF model focuses more on DDoS and less on Intrusion. This suggests that although the models have comparable accuracy, they may have different tendencies in detecting certain types of attacks. Therefore, for further improvements to the model or expansion of the dataset, it is necessary to consider addressing this imbalance and increasing the diversity of attack detection. In other words, although Artificial Neural Network technology shows potential in data security attack detection, there is still room for further improvement and optimization.

REFERENCES

[1] D. Shi, X. Yuanjun and P. Tao, *Network Abnormal Traffic Detection Model Based on Semi-Supervised Deep Reinforcement Learning*. IEEE Transactions on Network and Service Management, 2021.

[2] X. Yang, el at, *Machine Learning and Deep Learning Methods for Cybersecurity*. IEEE Access, 2018.

[3] M. Arif Wirawan, R. Imam, and Sunardi, *Deteksi Serangan DDoS Menggunakan Neural Network dengan Fungsi Fixed Moving Average Window*. JISKa, 2017.

[4] M. Schofied, G. Alicioglu, R. Binaco, P. Turner, C. Thatcher, A. Lam, and B. Sum, *Convolutional Neural Network for Malware Classification Based on API Call Sequence*. Computer Science & Information Technology (CS & IT), 2021.

[5] Z. Hossain, Md. Mahmudur Rahman Sourov, M. Khan, and P. Rahman, *Network Intrusion Detection using Machine Learning Approaches*. IEEE Xplore, 2021.

[6] A. Anand, S. Rain, D. Anand, H. Moaiteq Aljahdali, and D. Kerr, *An Efficient CNN-Based Deep Learning Model to Detect Malware Attacks (CNN-DMA) in 5G-IoT Healthcare Applications*. Sensors, 2021.

[7] J. Kim, Ji. Kim, H. Kim, M. Shim, and E. Choi, *CNN-Based Network Intrusion Detection against Denial-of-Service Attacks*. Elektronics, 2020.

[8] K. Sagar Sahoo, B. Khrishna Tripathy, K. Naik, S. Ramasubbareddy, and D. Burgos, *An Evolutionary SVM Model for DDoS Attack Detection in Software Defined Networks*. IEEE Access, 2020.

[9] M. Shoaib Akhtar and T. Feng, *Malware Analysis and Detection Using Machine Learning Algorithms*. Symmetry, 2022.

[10] B. Mohammed Khammas, *Ransomware Detection using Random Forest Technique*. ICT Express 6, 2020.

[11] P. Angelo Alves Resende and A. Costa, *A Survey of Random Forest Based Methods for Intrusion Detection Systems*. ACM Computing Surveys, 2018.

[12] N. Ashfaq Najar and N. S Manohar, *DDoS Attack Detection using MLP and Random Forest Algorithms*. Spinger, 2022

[13] Y. Li and S. Abdallah, *A Transfer Learning and Optimized CNN Based Intrusion Detection System for Internet of Vehicles*. IEEE Xplore