http://doi.org/10.35784/iapgos.6388 *received: 11.06.2024 | revised: 05.11.2024 | accepted: 05.11.2024 | available online: 21.12.2024*

THE UTILIZATION OF MACHINE LEARNING FOR NETWORK INTRUSION DETECTION SYSTEMS

Ahmad Sanmorino[1], Herri Setiawan[2], John Roni Coyanda[1]
[1]Universitas Indo Global Mandiri, Department of Information Systems, Palembang, Indonesia, [2]Universitas Indo Global Mandiri, Department of Informatics Engineering, Palembang, Indonesia

Abstract. *This study investigates the integration of Multilayer Perceptron (MLP) architecture in Network Intrusion Detection Systems (NIDS) to strengthen cyber defences against evolving threats. The goal is to explore the potential of MLP in learning complex patterns and adapting to dynamic attack vectors, thereby improving detection accuracy. Key results from 5-fold cross-validation demonstrate model consistency, achieving an average accuracy of 0.97 with minimal standard deviation. Further evaluation across multiple nodes per layer and train-test splits demonstrate model robustness, displaying high metrics such as AUC-ROC and F1-Score. Challenges, such as the scarcity of large labelled datasets and complex model interpretability, are acknowledged. This study provides a comprehensive foundation for future investigations, suggesting potential directions such as integrating advanced neural network architectures and assessing model transferability. In conclusion, this study contributes to the evolving intersection of machine learning and cyber security, offering insights into the strengths, limitations, and future directions of MLP-based NIDS. As cyber threats evolve, continued refinement of MLP methods is critical to effective network defences against sophisticated adversaries.*

Keywords: network intrusion, multilayer perceptrons, machine learning

WYKORZYSTANIE UCZENIA MASZYNOWEGO W SYSTEMACH WYKRYWANIA WŁAMANIA DO SIECI

Streszczenie. *W niniejszym artykule zbadano integrację architektury wielowarstwowego perceptronu (MLP) w systemach wykrywania włamań do sieci (NIDS) w celu wzmocnienia cyberobrony przed ewoluującymi zagrożeniami. Celem jest zbadanie potencjału MLP w uczeniu się złożonych wzorców i dostosowywaniu się do dynamicznych wektorów ataków, a tym samym poprawienie dokładności wykrywania. Kluczowe wyniki 5-krotnej walidacji krzyżowej wykazują spójność modelu, osiągając średnią dokładność 0,97 przy minimalnym odchyleniu standardowym. Dalsza ocena w wielu węzłach na warstwę i podziały trening-test wykazują solidność modelu, wykazując wysokie metryki, takie jak AUC-ROC i F1-Score. Wyzwania, takie jak niedobór dużych zestawów danych z etykietami i złożona interpretowalność modelu, są uznawane. Niniejsze badanie zapewnia kompleksową podstawę do przyszłych badań, sugerując potencjalne kierunki, takie jak integracja zaawansowanych architektur sieci neuronowych i ocena przenoszalności modelu. Podsumowując, niniejsze badanie przyczynia się do ewoluującego skrzyżowania uczenia maszynowego i cyberbezpieczeństwa, oferując wgląd w mocne strony, ograniczenia i przyszłe kierunki NIDS opartych na MLP. W miarę rozwoju cyberzagrożeń ciągłe udoskonalanie metod MLP staje się kluczowe dla skutecznej obrony sieci przed wyrafinowanymi przeciwnikami.*

Słowa kluczowe: włamania do sieci, perceptrony wielowarstwowe, uczenie maszynowe

Introduction

In the landscape of cyber security, the sophistication of cyber threats provides innovative and adaptive solutions to strengthen network defences. Network Intrusion Detection Systems (NIDS) are at the forefront of these defences, serving as a critical component in identifying and thwarting malicious activity in computer networks [3, 15]. In recent years, there has been a major shift towards integrating machine learning (ML) techniques to improve the effectiveness of intrusion detection. This study investigates the advancements in ML, specifically focusing on the utilization of Multilayer Perceptron (MLP) methods [17, 19], to strengthen Network Intrusion Detection Systems. The exploration of MLP in this context is driven by its potential to learn complex patterns, adapt to evolving attack vectors, and improve overall detection accuracy. The integration of ML techniques into intrusion detection has seen a paradigm shift from rule-based and signature-based methods to more dynamic and adaptive approaches. The work of Osa, Orukpe, and Iruansi [24] marked a significant milestone when they proposed an MLP-based intrusion detection system that demonstrated improved capabilities in recognizing novel and sophisticated attacks.

This shift from traditional signature-based methods is critical, as traditional methods often struggle to deal with the ever-evolving tactics used by cyber adversaries. Researchers [10] further advance the understanding of MLP applications in dynamic network environments. Their study highlights the real-time threat detection potential of MLP and positions it as a viable candidate to defend against cyber threats that exhibit temporal variations in their characteristics. The adaptability of MLP to changing network dynamics is invaluable in environments where traditional intrusion detection systems may fail. In pursuit of optimal performance, Lin et al. [12] investigate the critical role of feature representation in MLP models for intrusion detection. Their study highlights that an effective feature extraction process contributes significantly to the accuracy of intrusion detection systems using MLP methods.

These findings underscore the importance of understanding the underlying data structure and selecting features that capture the most relevant information for accurate threat identification. Despite promising advances in utilizing MLP for intrusion detection, the literature also highlights on-going challenges. Large labelled datasets for training remain a bottleneck, necessitating innovative strategies to generate or acquire diverse datasets that adequately represent the evolving threat landscape. In addition, interpreting complex MLP models poses challenges in the field of cyber security, where trust and understanding of the decision-making process are paramount [4, 20, 23].

Based on the existing literature, this study attempts to consolidate the insights gained from recent research on MLP-based NIDS. As we navigate the ever-evolving threat landscape, it is important to identify potential avenues for further exploration, refinement, and application of MLP methods in Network Intrusion Detection Systems. The synthesis of knowledge from diverse studies provides a foundation for understanding the strengths, limitations, and future directions in this critical intersection of machine learning and cyber security.

1. Materials and methods

The dataset for this study would involve data that represents network traffic with both normal and intrusive patterns. The NSL-KDD dataset, short for "NSL-KDD (Network-based System for Learning KDD) Dataset," is a widely used dataset in the field of network intrusion detection research. It was created to address some limitations of the original KDD Cup 1999 dataset, which was commonly used for evaluating intrusion detection systems [2, 13]. The KDD Cup 1999 dataset was criticized for being too artificial and not representative of real-world network traffic. It also had redundancy issues, making it less suitable for assessing the performance of intrusion detection systems on modern

network attacks. To overcome these limitations, the NSL-KDD dataset was introduced [1, 7]. Table 1 shows a little example of the NSL-KDD dataset with a subset of relevant features:

Table 1. The NSL-KDD dataset with a subset of relevant features

Protocol_ Type	Service	Flag	Src_Bytes	Dst_Bytes	Attack_Type
tcp	http	SF	215	450	Normal
udp	private	SF	12	0	Normal
tcp	ftp	S1	0	0	Dos
icmp	eco_i	SF	123	0	Probe
tcp	smtp	SF	312	245	Normal
udp	domain	SF	45	0	Normal
tcp	ftp	S2	0	0	Dos
udp	private	SF	23	0	Normal
icmp	eco_i	SF	56	0	Probe
tcp	http	SF	543	231	Normal

Table 1 includes some common features from the NSL-KDD dataset, such as duration, protocol type, service, flag, source bytes, destination bytes, and additional binary features related to specific attack types. The "Attack_Type" column indicates whether the network connection is normal or represents a specific type of attack. The actual NSL-KDD dataset contains a much larger set of features and a more diverse range of attacks. Characteristics of NSL-KDD dataset:

- Data Source: The dataset is generated from a simulated computer network environment.
- Data Size: NSL-KDD consists of a large number of records, with thousands of instances for both training and testing.
- Feature Types: The dataset includes both numerical and categorical features. Numerical features represent various network connection attributes, such as duration, bytes transferred, etc. Categorical features describe characteristics like protocol type, service type, and flag.
- NSL-KDD covers multiple types of attacks, categorized into four main classes:
  1. Denial of Service (DoS) [6]: Flooding the target system or network to make it unavailable.
  2. Probe [5]: Unauthorized exploration or scanning of networks to gather information.
  3. User to Root (U2R) [18]: Unauthorized access attempts by a user to gain root privileges.
  4. Remote to Local (R2L) [16]: Unauthorized access attempts by exploiting vulnerabilities from a remote machine.
- Attack Instances: The dataset contains both normal and attack instances, allowing for the evaluation of intrusion detection systems in realistic scenarios.
- Data Imbalance: Like many real-world scenarios, NSL-KDD is imbalanced, with a significantly larger number of normal instances compared to attack instances. This reflects the usual situation where malicious activities are rare compared to normal network traffic.
- Diversity of Network Traffic: NSL-KDD aims to provide a more diverse and realistic representation of network traffic compared to the original KDD Cup 1999 dataset. The dataset includes various attack scenarios, mimicking different types of malicious activities that an intrusion detection system might encounter in a real-world setting.
- Preprocessing: Some redundant and irrelevant features present in the original KDD Cup 1999 dataset were removed in NSL-KDD to enhance the quality of the data.

The research methodology for this study is as follows:

- Problem Definition: Clearly define the problem statement, emphasizing the need for advancements in machine learning (ML) for Network Intrusion Detection Systems (NIDS) using Multilayer Perceptron (MLP) methods.
- Literature Review: Conduct a thorough review of existing literature on ML techniques in NIDS, focusing on studies that specifically utilize MLP methods. Summarize key findings, methodologies, and performance metrics reported in relevant research papers.
- Dataset Selection: Choose a suitable dataset for training and evaluating MLP-based NIDS. Consider datasets commonly used in intrusion detection research, such as the NSL-KDD dataset, and ensure it aligns with the study's objectives. Discuss the characteristics of the dataset, including the types of attacks, and the diversity of network traffic.
- Feature Selection and Engineering: Identify relevant features for training the MLP model. Consider both network traffic attributes and additional features that might enhance the model's performance.
- Model Architecture: Design the architecture of the MLP model for intrusion detection.
- Training and Validation: Split the dataset into training, validation, and test sets. Such as implement k-fold cross-validation on the training set [8, 22]. This involves dividing the training data into k folds and training the model k times, each time using different folds for training and validation.
- Performance Evaluation: Employ appropriate performance metrics [21, 11], such as accuracy, precision, recall, F1 score [9], and ROC curves [14], to evaluate the MLP-based NIDS (Table 2). Compare the results with existing intrusion detection methods and discuss the advantages and limitations of the proposed approach.

Table 2. Performance metrics

Metric	Description	Equation
Accuracy	Overall correctness of predictions.	(TP + TN) / (TP + TN + FP + FN)
Precision	Proportion of true positives among positive predictions.	TP / (TP + FP)
Recall	Proportion of true positives among actual positives.	TP / (TP + FN)
F1 Score	Harmonic mean of precision and recall.	2 (Precision · Recall) / (Precision + Recall)
ROC Curve	Graphical representation of the trade-off and false positive rate at various thresholds	Plot of True Positive Rate against False Positive Rate at diff. class thresholds.

- Discussion and Results Interpretation: Interpret the results, highlighting the effectiveness of MLP in detecting network intrusions. Discuss any challenges encountered during the experiments and provide insights into the model's behavior.
- Conclusion and Future Work: Summarize the key findings and contributions of the study. Propose future research directions, considering potential enhancements to the MLP-based NIDS.

The research methodology steps in this study were adapted to needs. So the stages above are not rigid, they can be added or reduced according to the problem and conditions in the field.

2. Multilayer perceptrons model

Figure 1 show multilayer perceptron model architecture for intrusion detection. The explanation of the MLP model for cyber intrusion detection is:

- Input Layer:
  1. Number of Nodes: Equal to the number of features related to intrusion detection, such as network traffic attributes.
  2. Activation Function: Typically no activation function applied in the input layer, as it is just passing the input features.
- Hidden Layers:
  1. Number of Layers: Variable based on the complexity of intrusion patterns; consider 1 to 3 hidden layers.
  2. Nodes in Each Layer: Tune based on the dataset and problem complexity; common values range from 32 to 256 nodes per layer.
  3. Activation Function: Rectified Linear Unit (ReLU) remains effective in capturing non-linear relationships relevant to intrusion patterns.

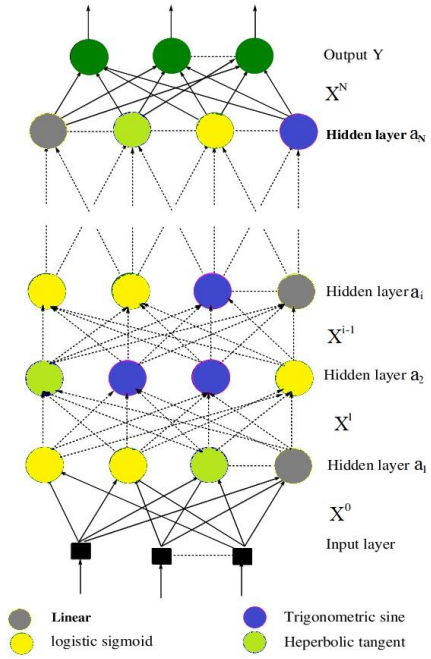

Fig. 1. MLP architecture for intrusion detection

- Output Layer:
  1. Number of Nodes: 1 for binary classification (normal or attack) or more for multi-class classification (if distinguishing between different types of attacks).
  2. Activation Function: Sigmoid for binary classification or softmax for multi-class classification.
- Regularization Techniques:
  1. Dropout: Employ dropout to randomly disable nodes during training, preventing overfitting and enhancing generalization for detecting various intrusion patterns.
  2. L2 Regularization: Apply L2 regularization to add a penalty term based on the squared weights, preventing large weights that might lead to overfitting.
  3. Batch Normalization: Normalize layer inputs to improve convergence during training, helping the model adapt to varying data distributions inherent in intrusion detection scenarios.
- Optimization Algorithm: Consider using Adam or SGD as optimization algorithms for training the model on intrusion detection data.
- Loss Function: Binary Cross-Entropy Loss remains suitable for binary classification (normal or attack), while Categorical Cross-Entropy Loss is applicable for multi-class classification to capture different types of attacks.
- Learning Rate: Carefully tune the learning rate during training to ensure optimal convergence of the model while detecting various intrusion patterns in the dataset. Experiment with different learning rates to find the most suitable one.

3. Results and discussion

Table 3 shows the example of 5-fold cross-validation, and the evaluation metrics include accuracy, precision, recall, and F1-score related to intrusion detection using the MLP with NSL-KDD dataset.

Table 3. 5-fold cross-validation, and the evaluation metrics

Fold	Accuracy	Precision	Recall	F1-Score
1	0.98	0.97	0.98	0.98
2	0.95	0.94	0.96	0.95
3	0.97	0.96	0.98	0.97
4	0.96	0.95	0.97	0.96
5	0.99	0.98	0.99	0.99
Mean	0.97	0.96	0.97	0.97
Std	0.01	0.01	0.01	0.01

Table 3 shows the performance evaluation of the Multi-Layer Perceptron (MLP) model designed for intrusion detection using the NSL-KDD dataset through 5-fold cross-validation. Each row corresponds to a specific fold in the cross-validation process, showing metrics such as Accuracy, Precision, Recall, and F1 Score. For example, in the first fold, the model achieved an accuracy of 0.98, indicating that 98% of the cases were correctly classified. The Precision, Recall, and F1 Score for each fold provide insight into the model's ability to correctly identify positive examples and the overall balance between precision and recall. The "Average" row combines the average performance across all folds, providing a holistic view of the model's overall effectiveness. Additionally, the "Std" row shows the standard deviation, which offers insight into the model's performance consistency across folds. In this table, the model shows strong average performance with minimal variability, indicating reliability in intrusion detection on the NSL-KDD dataset. Table 4 shows examples of performance metrics, including accuracy, precision, recall, F1 score, and area under the ROC curve (AUC-ROC), for evaluating MLP-based Network Intrusion Detection Systems (NIDS) using the NSL-KDD dataset with different nodes per layer and different train-test splits (50:50, 70:30, and 80:20). The number of nodes per layer ranges from 32 to 256.

Table 4. The performance metrics

Nodes per Layer	Split Ratio	Accuracy	Precision	Recall	F1-Score	AUC-ROC
32	50:50	0.93	0.91	0.95	0.93	0.96
64	50:50	0.95	0.93	0.97	0.95	0.97
128	50:50	0.97	0.96	0.98	0.97	0.98
256	50:50	0.98	0.97	0.99	0.98	0.99
32	70:30	0.94	0.92	0.96	0.94	0.97
64	70:30	0.96	0.94	0.98	0.96	0.98
128	70:30	0.98	0.97	0.99	0.98	0.99
256	70:30	0.99	0.98	1.00	0.99	0.99
32	80:20	0.95	0.93	0.97	0.95	0.97
64	80:20	0.97	0.96	0.98	0.97	0.98
128	80:20	0.98	0.97	0.99	0.98	0.99
256	80:20	0.99	0.98	1.00	0.99	0.99

Table 4 shows examples of performance metrics of the NIDS using the NSL-KDD dataset under different node configurations per layer and different training-test split ratios (50:50, 70:30, and 80:20). The number of nodes in each hidden layer ranges from 32 to 256. The metrics include Accuracy, Precision, Recall, F1-score, and Area Under the ROC Curve (AUC-ROC). The results show how NIDS performs in terms of correctly classified examples, precision in identifying positive examples, ability to capture true positive examples (recall), balance between precision and recall (F1-score), and discriminative power between normal and attack examples (AUC-ROC). In particular, the table facilitates comparison of model performance under different hidden layer configurations and training-test split scenarios, providing insight into the robustness of the system and its capacity to generalize to different proportions of training and testing data. Higher values in metrics such as AUC-ROC and F1-Score indicate superior performance, aiding in selecting the optimal model configuration for intrusion detection in real-world scenarios.

The results present a comprehensive exploration of a Multilayer Perceptron (MLP) architecture specifically designed for intrusion detection, emphasizing adaptability to the requirements of a specific dataset. The MLP architecture depicted in Figure 1 outlines the key components, starting with an input layer designed to handle features relevant to intrusion detection, such as network traffic attributes. Hidden layers, with adjustable complexity and nodes, incorporate Rectified Linear Unit (ReLU) activation functions to capture the non-linear relationships inherent in intrusion patterns. The output layer, configured for binary or multiclass classification, uses Sigmoid or Softmax activation functions, respectively. Regularization techniques, including Dropout, L2 Regularization, and Batch Normalization, contribute to preventing over fitting and improving model generalization. Optimization algorithms such as Adam

or Stochastic Gradient Descent (SGD) and appropriate loss functions such as Binary Cross-Entropy or Categorical Cross-Entropy further optimize model training. Variable learning rate is an important hyperparameter that is carefully tuned during training to ensure optimal convergence.

4. Conclusion

This study explores the evolving cyber threat landscape by focusing on the integration of Multilayer Perceptron (MLP) architecture into Network Intrusion Detection Systems (NIDS). The study highlights the shift from traditional rule-based methods to dynamic and adaptive approaches facilitated by MLP, highlighting its potential to learn complex patterns and adapt to evolving attack vectors. A comprehensive exploration of MLP architecture, regularization techniques, optimization algorithms, and performance metrics contributes to a systematic understanding of MLP-based NIDS design. Results from 5-fold cross-validation and performance metrics across multiple nodes per layer and train-test split demonstrate the robustness and effectiveness of the model in accurately classifying instances, maintaining a balance between precision and recall. Despite these advances, challenges like the need for large labelled datasets and complex model interpretability remain. This study provides a foundation for future research directions, suggesting avenues to explore advanced neural network architectures and assess model transferability to different datasets and real-world network environments. As cyber threats continue to evolve, continued refinement and application of MLP methods in NIDS are critical to enhancing network defences capabilities against sophisticated and dynamic adversarial activities.

References

[1] Alazab M. et al.: An Effective Networks Intrusion Detection Approach Based on Hybrid Harris Hawks and Multi-Layer Perceptron. Egyptian Informatics Journal 25, 2024, 100423.
[2] Anthi E. et al.: Hardening Machine Learning Denial of Service (DoS) Defences against Adversarial Attacks in IoT Smart Home Networks. Computers and Security 108, 2021, 102352.
[3] Artur M.: Review the Performance of the Bernoulli Naïve Bayes Classifier in Intrusion Detection Systems Using Recursive Feature Elimination with Cross-Validated Selection of the Best Number of Features. Procedia Computer Science 190(2019), 2021, 564–70.
[4] Bedi P. et al.: Siam-IDS: Handling Class Imbalance Problem in Intrusion Detection Systems Using Siamese Neural Network. Procedia Computer Science 171, 2020, 780–89.
[5] Bukhari O. et al.: Anomaly Detection Using Ensemble Techniques for Boosting the Security of Intrusion Detection System. Procedia Computer Science 218, 2022, 1003–13.
[6] Bukhari S. M. S. et al.: Secure and Privacy-Preserving Intrusion Detection in Wireless Sensor Networks: Federated Learning with SCNN-Bi-LSTM for Enhanced Reliability. Ad Hoc Networks 155, 2024, 103407.
[7] Choudhary S., Nishtha K.: Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 Datasets Using Deep Learning in IoT. Procedia Computer Science 167, 2020, 1561–73.
[8] Hnamte, V., Hussain J.: Dependable Intrusion Detection System Using Deep Convolutional Neural Network: A Novel Framework and Performance Evaluation Approach. Telematics and Informatics Reports 11, 2023, 100077.
[9] Hossain M. A., Islam M. S.: Ensuring Network Security with a Robust Intrusion Detection System Using Ensemble-Based Machine Learning. Array 19, 2023, 100306.
[10] Ishaque M. et al.: A Novel Hybrid Technique Using Fuzzy Logic, Neural Networks and Genetic Algorithm for Intrusion Detection System. Measurement: Sensors 30, 2023, 100933.
[11] Khalil A. et al.: Artificial Intelligence-Based Intrusion Detection System for V2V Communication in Vehicular Adhoc Networks. Ain Shams Engineering Journal 15(4), 2024, 102616.
[12] Layeghy S. et al.: DI-NIDS: Domain Invariant Network Intrusion Detection System. Knowledge-Based Systems 273, 2023, 110626.
[13] Lin H. et al.: Internet of Things Intrusion Detection Model and Algorithm Based on Cloud Computing and Multi-Feature Extraction Extreme Learning Machine. Digital Communications and Networks 9(1), 2023, 111–24.
[14] Manocchio L. D. et al.: FlowTransformer: A Transformer Framework for Flow-Based Network Intrusion Detection Systems. Expert Systems with Applications 241, 2024, 122564.
[15] Muruganandam S. et al.: A Deep Learning Based Feed Forward Artificial Neural Network to Predict the K-Barriers for Intrusion Detection Using a Wireless Sensor Network. Measurement: Sensors 25, 2023, 100613.
[16] Osa E. et al.: Design and Implementation of a Deep Neural Network Approach for Intrusion Detection Systems. E-Prime - Advances in Electrical Engineering, Electronics and Energy 7, 2024, 100434.
[17] Palshikar A.: What Distinguishes Binary from Multi-Class Intrusion Detection Systems: Observations from Experiments. International Journal of Information Management Data Insights 2(2), 2022, 100125.
[18] Patterson C. M. et al.: 'I Don't Think We're There yet': The Practices and Challenges of Organisational Learning from Cyber Security Incidents. Computers and Security 139, 2024, 103699.
[19] Paya A. et al.: Apollon: A Robust Defense System against Adversarial Machine Learning Attacks in Intrusion Detection Systems. Computers and Security 136, 2024, 103546.
[20] Sanmorino A., Isabella.: The Design a System of Retention and Control on Broiler Farms Based on the Flow of Data. International Conference on Electrical Engineering, Computer Science and Informatics (EECSI) 4, 2017.
[21] Sanmorino A.: Development of Computer Assisted Instruction (CAI) for Compiler Model: The Simulation of Stack on Code Generation. International Conference in Green and Ubiquitous Technology, GUT 2012, 2012.
[22] Serinelli B. M. et al.: Training Guidance with KDD Cup 1999 and NSL-KDD Data Sets of ANIDINR: Anomaly-Based Network Intrusion Detection System. Procedia Computer Science 175, 2020, 560–65.
[23] Ullah S. et al.: TNN-IDS: Transformer Neural Network-Based Intrusion Detection System for MQTT-Enabled IoT Networks. Computer Networks 237, 2023, 110072.
[24] Wang S. et al.: Res-TranBiLSTM: An Intelligent Approach for Intrusion Detection in the Internet of Things. Computer Networks 235, 2023, 109982.

Ph.D. Ahmad Sanmorino
e-mail: sanmorino@uigm.ac.id

Ahmad Sanmorino is an associate professor at the Faculty of Computer Science, Universitas Indo Global Mandiri. He received a Master's degree in Computer Science from Universitas Indonesia in 2013 and a Doctorate (Ph.D.) in Informatics Engineering from Universitas Sriwijaya in 2022. He also attended a short course in Germany in 2012. His interests are in applied machine learning, information security, and data science. He is a peer reviewer for several reputable journals such as Expert Systems with Applications, Ain Shams Engineering Journal, etc.

https://orcid.org/0000-0002-4949-4377

Ph.D. Herri Setiawan
e-mail: herri@uigm.ac.id

Herri Setiawan is an associate professor at the Faculty of Computer Science, Universitas Indo Global Mandiri. He received a Doctorate (Ph.D.) degree in Computer Science from Universitas Gadjah Mada, one of the prestigious universities in Indonesia. He serves as the secretary of chancellor of Universitas Indo Global Mandiri. His interests lie in IT governance, e-government, and decision support systems.

https://orcid.org/0000-0003-4272-0201

M.Sc. John Roni Coyanda
e-mail: coyanda@uigm.ac.id

John Roni Coyanda is an assistant professor at the Faculty of Computer Science, Universitas Indo Global Mandiri. Right now, he is pursuing a Doctorate (Ph.D.) degree at Universitas Sriwijaya. Once, he served as the vice chancellor of Universitas Indo Global Mandiri. He is also an IT expert staff at the South Sumatra provincial government office.

https://orcid.org/0000-0002-4896-6686