Indonesian Journal of Electrical Engineering and Computer Science
Vol. 35, No. 1, July 2024, pp. 485~493
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v35.i1.pp485-493

Performance evaluation of multiclass classification models for ToN-IoT network device datasets

Soni[1,2], Muhammad Akmal Remli[2,3], Kauthar Mohd Daud[4], Januar Al Amien[1,2]
[1]Department of Informatics Engineering, Faculty of Computer Science, Universitas Muhammadiyah Riau, Riau, Indonesia
[2]Faculty of Data Science and Computing, Universiti Malaysia Kelantan, Kelantan, Malaysia
[3]Institute for Artificial Intelligence and Big Data, Universiti Malaysia Kelantan, Kelantan, Malaysia
[4]Institute Center for Artificial Intelligence Technology, Faculty of Information Science and Technology,
Universiti Kebangsaan Malaysia, Bangi Selangor, Malaysia

Article Info

Article history:

Received Jan 26, 2024
Revised Mar 3, 2024
Accepted Mar 22, 2024

Keywords:

Classification
Intrusion detection
LightGBM
Multiclass
ToN-IoT
XGBoost

ABSTRACT

Internet of things (IoT) technology has empowered tangible objects to establish internet connections, facilitating data exchange with computational capabilities. With significant potential across sectors like healthcare, environmental monitoring, and industrial control, IoT represents a promising technological advancement. This study explores datasets from ToN-IoT's IoT devices, focusing on multi-class classification, including normal and attack classes, with an additional aim of identifying potential attack sub-classes. Datasets comprise various IoT devices, such as refrigerators, garage doors, global positioning systems (GPS) sensors, motion lights, modbus devices, thermostats, and weather sensors. Comparative analysis is conducted between two prominent multiclass classification models, extreme gradient boosting (XGBoost) and light gradient boosting machine (LightGBM), utilizing accuracy and computational time metrics as evaluation criteria. Research findings highlight that the LightGBM model achieves superior accuracy at 78%, surpassing XGBoost 74.31%. However, XGBoost demonstrates an advantage with a shorter computational time of 1.23 seconds, compared to LightGBM 6.79 seconds. This study not only provides insights into multiclass classification model selection but also underscores the crucial consideration of the trade-off between accuracy and computational efficiency in decision-making. Research contributes to advancing our understanding of IoT security through effective classification methodologies. The findings offer valuable information for researchers and practitioners, emphasizing the nuanced decisions needed when selecting models based on specific priorities like accuracy and computational efficiency.

Corresponding Author:

Muhammad Akmal Remli
Faculty of Data Science and Computing, Universiti Malaysia Kelantan
Kelantan, Malaysia
Email: akmal@umk.edu.my

1. INTRODUCTION

The internet of things (IoT) is an emerging technology that enables the connection of numerous devices through computerized systems, facilitating data sharing among them. IoT integrates physical and digital objects to enhance various aspects of our daily lives, including applications like industrial IoT (IIoT), intelligent residences and urban areas, advanced power distribution networks, and cyber-physical systems (CPS) [1]–[3]. The escalating apprehensions regarding potential threats to IoT applications and the urgent necessity to mitigate associated risks have recently garnered substantial attention in the cybersecurity domain [4]. Cybercrime activities on the internet primarily involve communication-based attacks. Detecting malicious network traffic with minimal cost is increasingly challenging for information security researchers [5]. Although there has been considerable interest in understanding the cyber threat landscape of IoT networks and devising security methodologies incorporating artificial intelligence (AI), a gap exists in distributed architecture. This gap has led to the generation of heterogeneous datasets that encompass the authentic behaviors of real-world IoT networks and intricate cyber threat scenarios, intended for evaluating the robustness of novel systems [6], [7]. In the domain of intrusion detection (ID), data mining, machine learning, and deep learning have emerged as pivotal approaches for processing and forecasting data [8], [9]. Notably, machine learning stands out as the most effective method due to its superior true positive rate (TPR) compared to alternative techniques. To enhance network profiling and monitoring in IoT, [10] proposed a dynamic anomaly-based intrusion detection system (IDS). The study emphasized the importance of adaptive security measures in investigating suspicious transactions and potential attacks within IoT networks.

Previous research has extensively investigated the classification of IoT devices employing various machine learning techniques for multiclass classification. Logistic regression (LR) achieved an accuracy of 0.61%, linear discriminant analysis (LDA) demonstrated 0.62% accuracy, k-nearest neighbour (KNN) resulted in 0.72% accuracy, random forest (RF) showed 0.71% accuracy, classification and regression trees (CART) exhibited 0.77% accuracy, Naïve Bayes (NB) yielded 0.54% accuracy, support vector machine (SVM) attained 0.60% accuracy, and long short-term memory (LSTM) presented 0.68% accuracy in prior studies on IoT device classification. These findings offer insights into the comparative performance of diverse machine learning algorithms within this domain [2]. Khan *et al.* [11] has explored a similar investigation into multiclass classification in IoT. The conducted research reported accuracies of 76% for the combination of decision trees (DT), RF, KNN, and NB, 75% for the combination of DT, RF, and NB, and 76% for the combination of DT, RF, and KNN.

While previous studies have delved into multiclass classification using machine learning, the obtained results have been less satisfactory. Additionally, there is a gap in the literature as none have explored the utilization of ensemble learning techniques, such as XGBoost and LightGBM, to address the challenges associated with existing multiclass scenarios. Notably, references suggest that ensemble learning exhibits promising performance in effectively handling multiclass classification. Hence, there is a compelling need for further investigation to overcome multiclass challenges by leveraging ensemble learning. This exploration aims to contribute novel insights and methodologies to the existing body of knowledge.

In the pursuit of addressing the outlined challenges comprehensively, Alsaedi *et al.* [2] conducted a study focusing on the comparison of ensemble learning techniques. Their work aimed to provide insights into the efficacy of ensemble learning for solving both binary and multiclassification problems. Ensemble learning, which involves combining multiple models to improve overall performance, has shown promise in mitigating multiclass classification issues and enhancing the predictive capabilities of machine learning models. Alsaedi *et al.* [2] delved into the evaluation of various ensemble learning algorithms, including RF, AdaBoost, and gradient boosting, to discern the most effective approach. Their findings contributed valuable knowledge to the ongoing discourse on handling complex datasets, especially those with multiclass classification, categorical features, and missing values. The existing body of related work underscores the importance of addressing multiclass classification, categorical features, and missing values in datasets, particularly in the context of the ToN-IoT dataset. The study conducted by [2] lays the foundation for the present research, which seeks to build upon their findings and identify the optimal ensemble learning algorithm for addressing the multifaceted challenges of the ToN-IoT dataset. The ToN-IoT dataset poses significant challenges, notably in dealing with multiclass classification, categorical features, and missing values [12]. In addressing the multiclass classification issue, various methodologies have been proposed, with oversampling, undersampling, and hybrid approaches being prominent solutions. Oversampling involves replicating instances of the minority class, aiming to balance the class distribution within the dataset.

Furthermore, the integration of AI in IoT security approaches holds immense promise. However, a critical deficiency persists in the absence of a distributed architecture, resulting in the creation of datasets that lack authenticity in representing real-world IoT network behaviors and complex cyber threat scenarios. To address this gap, future research endeavors should focus on developing distributed architectures that facilitate the creation of datasets reflective of the diverse and dynamic nature of IoT networks. The primary contributions of this study are:

- Comprehensive comparative evaluation analysis of XGBoost and LightGBM in IoT intrusion detection, providing insights into their respective strengths and weaknesses.
- The optimization of intrusion detection patterns through multiclass classification, with a focus on enhancing speed and detection performance using ensemble learning techniques.

These contributions provide insightful perspectives and methodologies for addressing the intricate challenges posed by cyber threats in the dynamic and evolving landscape of IoT cybersecurity.

2. LITERATURE REVIEW

2.1. Multi-class classification problem

Binary classification is the process of categorizing output into two distinct groups. In our scenario, our binary classifiers should possess the capability to discern whether a given record constitutes an intrusion or not. In order to accomplish this, we categorize the labels into two classes: normal and attack. Additionally, to address issues arising from multiclass classification, we implemented a random sampling methodology [13]. Multiclass classification involves the categorization of output into three or more classes. Owing to the challenge of multiclass classification, we have organized attacks into three specific categories: normal, denial of service (DoS), and all other instances falling within the R2L category [13].

2.2. LightGBM

The light gradient boosting machine (LightGBM) is a unified algorithm designed for constructing gradient boosting decision trees (GBDT). It is distinguished by its accelerated training pace, reduced memory requirements, enhanced accuracy, and the ability to facilitate parallel processing of extensive datasets [14]–[16]. The proposition of LightGBM addresses the challenges confronted by GBDT in handling extensive datasets, thereby enabling more efficient and rapid application of GBDT in practical scenarios. Diverging from conventional algorithms employed in generating GBDTs, LightGBM offers distinct advantages, such as XGBoost [17], scikit-learn [18], and PGBRT [19].

2.3. XGBoost

Extreme gradient boosting (XGBoost) is grounded in the principles of GDBT, renowned for its remarkable speed and superior performance in comparison to alternative machine learning techniques. It functions as a methodology to augment the capabilities of machine learning models, particularly emphasizing tree boosting methods [9], [20], [21]. XGBoost, an abbreviation for XGBoost, demonstrates proficiency in efficient utilization of memory and hardware resources, thereby enhancing algorithmic efficiency and model refinement. By employing Taylor expansion on the cost function, incorporating considerations of the second derivative, XGBoost ensures heightened accuracy in result outcomes. This method optimizes the objective function through an iterative training process, where each subsequent phase of optimization relies on the outcomes derived from the preceding stage [12], [22], [23].

3. RESEARCH METHOD

The research method outlines the research framework in this study. The framework presents step by step procedures to be carried out at each stage of the research. Additionally, the required datasets for testing and evaluating the method are introduced. The utilized research method will be explained in Figure 1.

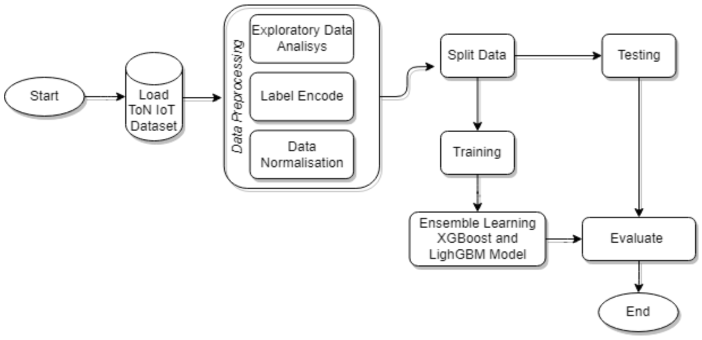


Figure 1. Research method

3.1. Dataset

The ToN-IoT datasets represent an innovative collection within the IoT/IIoT domain, network traffic, and operating systems. They are specifically crafted to evaluate the precision and efficacy of a variety of AI-driven cybersecurity applications. These datasets amalgamate diverse data origins, encompassing telemetry data from IoT/IIoT services, logs from operating systems, and network traffic sourced from an authentic depiction of a medium-scale network established at the cyber range and IoT labs at UNSW Canberra. The focal point is on the proposed dataset containing telemetry data from IoT/IIoT services and its inherent characteristics. Access to the ToN-IoT datasets is available through the ToN-IoT repository [24], [25].

The dataset employed in this study is the ToN-IoT dataset, presenting challenges in both binary and multiclass classification scenarios. It amalgamates seven distinct IoT training datasets, encompassing categories such as fridge, garage-door, GPS-tracker, modbus, motion-light, thermostat, and weather. These individual datasets are readily available at the ToN-IoT repository, providing a valuable resource for scholarly exploration and advancements in the field of IoT cybersecurity. The ToN-IoT datasets play a crucial role as a significant repository for research and development endeavors in this domain. For the purpose of this study, the seven datasets are combinated into a singular dataset, and preprocessing steps are undertaken to enhance its usability and facilitate comprehensive analysis.

3.2. ToN-IoT dataset preprocessing

Data preprocessing is a crucial phase undertaken to categorize the information within this dataset effectively. The dataset’s lack of well-defined structure is attributed to the presence of non-numeric data. To render the data suitable for analytical purposes, it must undergo a series of preprocessing stages. Furthermore, additional measures, such as addressing features with categorical elements (both primary and subcategories) via label encoder, can be implemented.

3.2.1. Exploratory data analysis

Exploratory data analysis (EDA) on the IoT dataset involves a systematic step-by-step process to gain insights into its structure and characteristics. Initially, an in-depth understanding of the dataset is established, identifying key attributes and features. The next step is data cleaning, where missing values are addressed, and outliers are examined. Descriptive statistics are computed to summarize the central tendencies and spread of the data. Data visualization techniques, such as histograms, scatter plots, and box plots, are employed to reveal patterns, relationships, and potential anomalies within the dataset. Feature engineering may be performed to create new variables that enhance the analysis. Correlation analysis assesses relationships between variables. Furthermore, statistical tests and machine learning models are applied for in-depth exploration. The EDA process concludes with a comprehensive interpretation of findings, guiding further analyses or model development based on a thorough understanding of the IoT dataset.

3.2.2. Label encoder

The label encoding process applied to the IoT dataset entails a series of systematic steps. Initially, categorical columns containing data slated for transformation into numerical representations are identified. Subsequently, essential libraries, such as scikit-learns label encoder, are incorporated. An instance of the label encoder is instantiated, and the fit transform method is executed on the specified columns necessitating label encoding. The data frame undergoes an update with the transformed values. In the scenario of novel data, it is imperative to employ the identical label encoder that underwent fitting during the training data phase. It is noteworthy that label encoding introduces a potential ordinal interpretation to the model in relation to the encoded values. Consequently, deliberation on alternative encoding methodologies, such as one-hot encoding, is recommended particularly in cases where no inherent ordinal relationship exists. Label encoding is employed to convert categorical labels into numerical values, assigning the numerical value of 1 to high-ranking categories and 0 to low-ranking ones.

3.2.3. Data normalization

Normalization of data in an IoT dataset involves a systematic series of steps to bring the values within the dataset into a consistent or standardized range. Initially, a profound understanding of the characteristics and distribution of the IoT dataset is conducted, including the identification of attributes or features requiring normalization. Features deemed relevant for normalization are then extracted. The subsequent step involves the selection of an appropriate normalization method, such as min-max scaling, z-score normalization, or robust normalization, contingent upon the data characteristics. For instance, min-max scaling transforms the values of each feature into the range of 0 to 1. Z-score normalization standardizes values to have a mean of zero and a standard deviation of one. Meanwhile, robust normalization is designed to handle outliers. After selecting the method, the normalization step is implemented on the IoT dataset. Evaluation of outcomes is conducted by examining the distribution and range of values post-normalization, ensuring that this process does not discard pertinent information. If necessary, iterations and adjustments are made, and the results are meticulously documented for future reference. This process guarantees that each feature in the dataset exerts a balanced impact on the model or analysis to be conducted, maintaining data integrity and quality.

3.3. Dataset splitting

The process of data splitting in machine learning involves partitioning the dataset into different subsets for specific purposes. The division of the dataset is generally conducted to enable an objective evaluation of the model's performance on data that was not utilized during the training process. The stages of data splitting are carried out to assess the performance of the algorithmically utilized model, divided into two main components: the training data and the testing data. The data splitting in this context follows an 80:20 ratio. The training data is utilized by the machine learning algorithm to adjust the parameters or weights of the model, enabling it to learn patterns or relationships within the data.

3.4. Model ensemble learning XGBoost and LightGBM

In the process of applying ensemble learning to the IoT dataset, particularly using the XGBoost model, a systematic series of steps was followed. First, the XGBoost model was selected for its efficiency in handling complex classification problems. The dataset was then divided into training and testing sets, and features were carefully chosen to ensure their significant impact on the desired outcomes. Following this, label encoding was applied to categorical features, and numerical features underwent normalization for consistent scaling. The XGBoost model was initialized with specific parameters such as learning rate, tree depth, and the number of estimators. The model was trained using the training data, and its performance was evaluated using the testing data, employing metrics like accuracy, precision, recall, and F1-score. If necessary, the model parameters were adjusted for better performance. The final step involved using the trained XGBoost model to make predictions on new or unseen data, providing valuable insights into the dataset's characteristics and facilitating informed decision-making in the context of the IoT dataset.

The application of ensemble learning to the IoT dataset involved the utilization of the LightGBM model, a powerful implementation of gradient boosting. The process followed a systematic series of steps to enhance the overall predictive performance. Initially, the LightGBM model was chosen due to its efficiency and scalability, making it particularly suitable for handling large datasets like those encountered in the IoT context. The dataset was partitioned into training and testing subsets, and features were judiciously selected for their relevance to the desired outcomes. Further preprocessing steps included label encoding for categorical features and normalization for numerical features, ensuring consistent scaling. After these preparations, the LightGBM model was initialized with specific parameters, such as the learning rate, tree depth, and the number of leaves. The model was trained using the training dataset, leveraging its capabilities in processing data through histogram-based methods and supporting direct handling of categorical variables. Evaluation of the model's performance was conducted on the testing dataset, employing metrics like accuracy, precision, RC, and F1-score. If necessary, the model's parameters were fine-tuned to optimize its predictive capabilities. The trained LightGBM model was then used to make predictions on new or unseen data, contributing valuable insights into the intricate patterns and relationships within the IoT dataset. Overall, the application of ensemble learning with the LightGBM model facilitated the creation of a robust and efficient predictive model for classification or prediction tasks within the realm of IoT.

3.5. Evaluation

The confusion matrix serves as a tool for assessing the performance of the generated classification model. Subsequently, these outcomes will be utilized to compute accuracy, precision, RC, and F1-score values [26]. Multiple evaluation metrics hold significance in this context [27], [28]. Recognizing the prevailing agreement that relying solely on accuracy is inadequate for comprehensive performance assessment, we provide values for a majority of these metrics, especially in scenarios where datasets exhibit an abundance of positive examples compared to negative ones. Accuracy, in the binary scenario, denotes the ratio of correct model classifications to the total number of classifications made.

$$Accuracy\ \% = \frac{TP+TN}{TP+FN+FP+TN} \tag{1}$$

Precision: specifies the correlation between correct predictions and the total predictions generated for a specific class. A heightened precision value is correlated with a diminished occurrence of false alarms. In the realm of binary scenarios.

$$Precision\ \% = \frac{TP}{TP+FP} \tag{2}$$

Recall: signifies the connection between correct predictions and the total occurrences within a designated class. An elevated recall value implies that a significant proportion of instances in a class have been accurately recognized. In the context of binary scenarios.

$$Recall\ \% = \frac{TP}{TP+FN} \quad (3)$$

F1-score (F1): metrics such as precision and recall present conflicting requirements, as enhancing one may result in a trade-off with the other. The F1-score is the harmonic mean of these two metrics. In the context of binary scenarios.

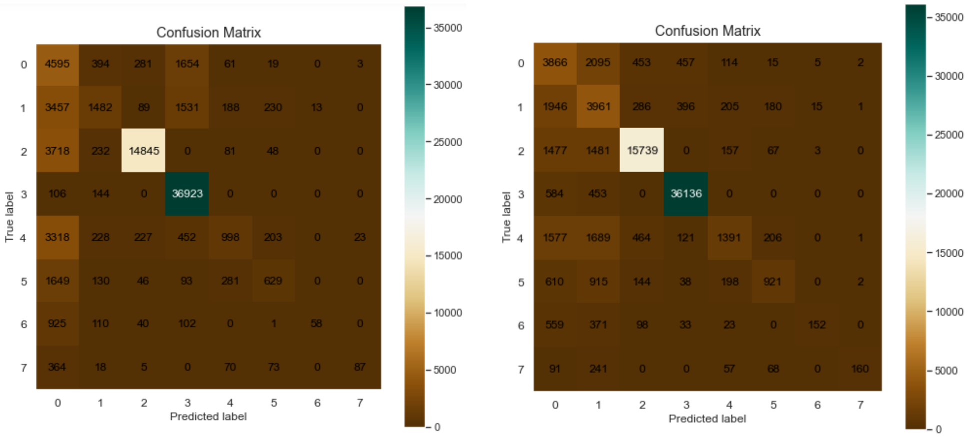
$$F1 = 2 * \frac{Presisi*Recall}{Presisi+Recall} \quad (4)$$

4. RESULTS AND DISCUSSION

The following section delves into the results and discussion stemming from our research conducted on the ToN-IoT dataset, encompassing seven distinct datasets corresponding to various IoT devices: fridge, garage door, GPS tracker, modbus, motion light, thermostat, and weather, with a focus on multi-classification in IoT. Now, let’s explore the outcomes of our study and delve into the implications and insights derived from the classification method developed to discern different types of attacks within the dataset.

Figure 2 represents a confusion matrix in the context of multiclass classification, where Figures 2(a) and 2(b) illustrate the results of an experimental comparison involving XGBoost and LightGBM algorithms within the framework of multiclass classification. The outcomes of this experiment are represented through a multi-class confusion matrix model. Specifically, this study evaluates the performance of both algorithms in classifying multiple classes, and the results are presented in the confusion matrix. Figures 2(a) and 2(b) provide visualizations demonstrating how numbers can be correctly classified diagonally, particularly showcasing the superior performance of the LightGBM model.

The experiment presents valuable insights into the multiclass classification capabilities of both algorithms, accentuating LightGBM’s adeptness in accurately categorizing numbers, particularly those aligning with the diagonal elements of the confusion matrix. This nuanced observation implies that LightGBM holds distinct advantages in tailored multiclass classification scenarios. The meticulous comparison enhances our comprehension of the relative strengths and merits of XGBoost and LightGBM within the multiclass classification domain, placing emphasis on the evident superiority demonstrated by LightGBM. These findings contribute significantly to informed decision-making in algorithm selection for multiclass classification, underscoring the nuanced advantages associated with LightGBM’s precise classification performance, and ensuring the authenticity of insights for submission to academic journal



(a) (b)

Figure 2. Multiclass classification (a) XGBoost and (b) LightGBM

Table 1 illustrates the performance evaluation results of two multiclass classification models, namely XGBoost and LightGBM, using common evaluation metrics. The following explanation and conclusions can be drawn from the table:

- Accuracy: indicates how well the model can correctly classify data. LightGBM has an accuracy of 0.78, while XGBoost has an accuracy of 0.74. Therefore, LightGBM achieves a higher accuracy level compared to XGBoost.

- Precision: indicates how well the model correctly identifies positive outcomes. LightGBM has a precision of 0.82, while XGBoost has a precision of 0.74. With a higher precision value, LightGBM is more effective in identifying and maintaining positive outcomes.
- Recall: measures the model's ability to identify all true instances of a class. Both LightGBM and XGBoost have the same recall value, which is 0.78.
- F1-score: represents the harmonic mean of precision and recall. Both models have the same F1-score, which is 0.78.
- Execution time: indicates the time required by each model to complete the classification process. XGBoost takes 1.22 seconds, while LightGBM takes 6.78 seconds. Although LightGBM has a longer execution time, the improved performance in terms of accuracy and precision can be considered an acceptable trade-off.

Based on the evaluation results, the LightGBM model demonstrates better performance in terms of accuracy and precision compared to the XGBoost model. Although LightGBM requires a longer execution time, the enhanced performance can be considered an advantage in the context of this multiclass classification application. The choice of the model depends on the specific project requirements, where aspects of speed and accuracy can be assessed according to the intended use.

Table 1. The performance of multiclass classification result

No	Model	Accuracy	Precision	Recall	F1-score	Time sec
1	XGBoost	0.74	0.74	0.74	0.74	1.22
2	LightGBM	0.78	0.82	0.78	0.78	6.78

5. CONCLUSION

In conclusion, the conducted experiments substantiate the superior performance of the multiclass classification model utilizing LightGBM compared to the XGBoost model. LightGBM achieved commendable metrics, including an accuracy level, precision, recall, and F1-score, all at 0.78. In contrast, the XGBoost model exhibited lower values for these metrics, specifically an accuracy of 0.74, precision of 0.74, recall of 0.74, and an F1-score of 0.74. Despite the higher execution time of the LightGBM model (6.78 seconds) in comparison to XGBoost (1.22 seconds), the observed trade-off between performance and execution time underscores the importance of careful model selection based on practical application requirements. Recent observations further emphasize the significance of employing ensemble learning techniques, specifically LightGBM, in addressing multiclass classification challenges. The achieved accuracy of 0.78% surpasses alternative algorithms. This underscores the potential impact of utilizing LightGBM in real-world applications, where enhanced accuracy is crucial. In addressing the broader context, the importance of these findings lies in advancing the field of multiclass classification within the realm of machine learning. The superiority of LightGBM underscores its potential as a preferred algorithm for such tasks. These results contribute valuable insights for researchers and practitioners seeking effective solutions to multiclass classification challenges. In responding to potential opposing viewpoints, it is essential to acknowledge the higher execution time of LightGBM. However, this drawback should be weighed against the substantially improved performance metrics, emphasizing that the trade-off is justified in scenarios where accuracy and precision are paramount. Readers should support this position to drive advancements in the field and capitalize on the strengths of LightGBM for improved multiclass classification outcomes.

REFERENCES

[1] A. Azmoodeh, A. Dehghantanha, and K. K. R. Choo, "Robust malware detection for internet of (battlefield) things devices using deep eigenspace learning," *IEEE Transactions on Sustainable Computing*, vol. 4, no. 1, pp. 88–95, 2019, doi: 10.1109/TSUSC.2018.2809665.

[2] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. N. Anwar, "ToN-IoT telemetry dataset: a new generation dataset of IoT and IIoT for data-driven intrusion detection systems," *IEEE Access*, vol. 8, pp. 165130–165150, 2020, doi: 10.1109/ACCESS.2020.3022862.

[3] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial internet of things: challenges, opportunities, and directions," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 11, pp. 4724–4734, 2018, doi: 10.1109/TII.2018.2852491.

[4] G. Falco, C. Caldera, and H. Shrobe, "IIoT cybersecurity risk modeling for SCADA systems," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4486–4495, 2018, doi: 10.1109/JIOT.2018.2822842.

[5] E. Farzadnia, H. Shirazi, and A. Nowroozi, "A novel sophisticated hybrid method for intrusion detection using the artificial immune system," *Journal of Information Security and Applications*, vol. 58, no. February, p. 102721, 2021, doi: 10.1016/j.jisa.2020.102721.

[6] N. Moustafa, B. Turnbull, and K. K. R. Choo, "Towards automation of vulnerability and exploitation identification in IIoT networks," *Proceedings - 2018 IEEE International Conference on Industrial Internet, ICII 2018*, no. Icii, pp. 139–145, 2018, doi: 10.1109/ICII.2018.00023.

[7] A. Churcher *et al.*, "An experimental analysis of attack classification using machine learning in IoT networks," *Sensors (Switzerland)*, vol. 21, no. 2, pp. 1–32, 2021, doi: 10.3390/s21020446.

[8] C. C. Sun, A. Hahn, and C. C. Liu, "Cyber security of a power grid: state-of-the-art," *International Journal of Electrical Power and Energy Systems*, vol. 99, pp. 45–56, 2018, doi: 10.1016/j.ijepes.2017.12.020.

[9] J. Al Amien, H. A. Ghani, N. I. M. Saleh, E. Ismanto, and R. Gunawan, "Intrusion detection system for imbalance ratio class using weighted XGBoost classifier," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 21, no. 5, pp. 1102–1112, 2023, doi: 10.12928/TELKOMNIKA.v21i5.24735.

[10] J. R. Rose, M. Swann, G. Bendiab, S. Shiaeles, and N. Kolokotronis, "Intrusion detection using network traffic profiling and machine learning for IoT," *Proceedings of the 2021 IEEE Conference on Network Softwarization: Accelerating Network Softwarization in the Cognitive Age, NetSoft 2021*, pp. 409–415, 2021, doi: 10.1109/NetSoft51509.2021.9492685.

[11] M. A. Khan *et al.*, "Voting classifier-based intrusion detection for IoT networks." pp. 313–328, 2022, doi: 10.1007/978-981-16-5559-3_26.

[12] A. R. Gad, A. A. Nashat, and T. M. Barkat, "Iintrusion detection system using machine learning for vehicular ad hoc networks based on ToN-IoT dataset (double)," *IEEE Access*, vol. 9. pp. 142206–142217, 2021, doi: 10.1109/ACCESS.2021.3120626.

[13] S. Nayyar, S. Arora, and M. Singh, "Recurrent neural network based intrusion detection system," in *Proceedings of the 2020 IEEE International Conference on Communication and Signal Processing, ICCSP 2020*, 2020, pp. 136–140, doi: 10.1109/ICCSP48568.2020.9182099.

[14] G. Ke *et al.*, "LightGBM: a highly efficient gradient boosting decision tree," *Advances in Neural Information Processing Systems*, vol. 2017-December, no. Nips, pp. 3147–3155, 2017.

[15] J. Liu, D. Yang, M. Lian, and M. Li, "Research on intrusion detection based on particle swarm optimization in IoT," *IEEE Access*, vol. 9, pp. 38254–38268, 2021, doi: 10.1109/ACCESS.2021.3063671.

[16] M. A. Muslim, Y. Dasril, M. Sam'an, and Y. N. Ifriza, "An improved light gradient boosting machine algorithm based on swarm algorithms for predicting loan default of peer-to-peer lending," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 28, no. 2, pp. 1002–1011, 2022, doi: 10.11591/ijeecs.v28.i2.pp1002-1011.

[17] T. Chen and C. Guestrin, "XGBoost: a scalable tree boosting system," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2016, vol. 13-17-Augu, pp. 785–794, doi: 10.1145/2939672.2939785.

[18] J. V. F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," T*he Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[19] S. Tyree, K. Q. Weinberger, and K. Agrawal, "Parallel boosted regression trees for web search ranking," *Proceedings of the 20th International Conference on World Wide Web, WWW 2011*, pp. 387–396, 2011, doi: 10.1145/1963405.1963461.

[20] N. Memon, S. B. Patel, and D. P. Patel, "Comparative analysis of artificial neural network and XGBoost algorithm for PolSAR image classification," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11941 LNCS, pp. 452–460, 2019, doi: 10.1007/978-3-030-34869-4_49.

[21] C. Zhang, Y. Zhang, X. Shi, G. Almpanidis, G. Fan, and X. Shen, "On incremental learning for gradient boosting decision trees," *Neural Processing Letters*, vol. 50, no. 1, pp. 957–987, 2019, doi: 10.1007/s11063-019-09999-3.

[22] H. Jiang, Z. He, G. Ye, and H. Zhang, "Network intrusion detection based on PSO-Xgboost model," *IEEE Access*, vol. 8, pp. 58392–58401, 2020, doi: 10.1109/ACCESS.2020.2982418.

[23] X. Ma, J. Sha, D. Wang, Y. Yu, Q. Yang, and X. Niu, "Study on a prediction of P2P network loan default based on the machine learning LightGBM and XGboost algorithms according to different high dimensional data cleaning," *Electronic Commerce Research and Applications*, vol. 31, pp. 24–39, 2018, doi: 10.1016/j.elerap.2018.08.002.

[24] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: network ToN_IoT datasets," *Sustainable Cities and Society*, vol. 72, 2021, doi: 10.1016/j.scs.2021.102994.

[25] T. M. Booij, I. Chiscop, E. Meeuwissen, N. Moustafa, and F. T. H. D. Hartog, "ToN_IoT: the role of heterogeneity and the need for standardization of features and attack types in IoT network intrusion data sets," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 485–496, 2022, doi: 10.1109/JIOT.2021.3085194.

[26] A. Agarwal, P. Sharma, M. Alshehri, A. A. Mohamed, and O. Alfarraj, "Classification model for accuracy and intrusion detection using machine learning approach," *PeerJ Computer Science*, vol. 7, pp. 1–22, 2021, doi: 10.7717/PEERJ-CS.437.

[27] X. Liu *et al.*, "NADS-RA: network anomaly detection scheme based on feature representation and data augmentation," *IEEE Access*, vol. 8, pp. 214781–214800, 2020, doi: 10.1109/ACCESS.2020.3040510.

[28] P. Araujo *et al.*, "Impact of feature selection methods on the classification of DDoS attacks using XGBoost," *Journal of Communication and Information Systems*, vol. 36, no. 1, pp. 200–214, 2021, doi: 10.14209/jcis.2021.22.

BIOGRAPHIES OF AUTHORS

Soni joins Faculty of Computer Science, Universitas Muhammadiyah Riau. He is also a senior lecturer and now he is deputy dean. He received a bachelor degree in Informatics Engineering Department from STMIK AMIK Riau, Indonesia. and a Master degree in Computer Science from Islamic University of Indonesia. His main research interests are data science, artificial intelligence, machine learning, and digital forensic. He can be contacted at email: soni@umri.ac.id.

Muhammad Akmal Remli joins Institute for Artificial Intelligence and Big Data (AIBIG), Universiti Malaysia Kelantan (UMK) as a fellow researcher in early 2020 and now he is AIBIG's director. He is also a senior lecturer at Faculty of Data Science and Computing, U K. He received a Master and a Ph.D. degree in Computer Science from Universiti Teknologi Malaysia in 2014 and 2018 before joining Universiti Malaysia Pahang from 2018 until 2020. In 2016, he worked at The Bioinformatics, Intelligent Systems and Educational Technology (BISITE) Research Group at University of Salamanca, Spain as research attachment and was working in cancer bioinformatics. His main research interests are artificial intelligence, data science, business intelligence and computational systems biology. He has published numerous scientific research papers indexed by Scopus and Clarivate Web of Science including in Expert Systems with Applications (ESWA) and Engineering Applications of Artificial Intelligence (EAAI) journals. He can be contacted at email: akmal@umk.edu.my.

Kauthar Mohd Daud currently serves as a Senior Lecturer in the Center for Artificial Intelligence Technology, Faculty of Information Science and Technology in Universiti Kebangsaan Malaysia. She received her B.Sc. in Bioinformatics and MSc in Bioinformatics from Multimedia University and the University of Malaya. In 2019, she received her Ph.D. in computer science from Universiti Teknologi Malaysia. Her expertise includes optimization, metabolic modeling, artificial intelligence, and machine learning. She can be contacted at email: kauthar.md@ukm.edu.my.

Januar Al Amien completed education bachelor's degree in the Informatics Engineering Department, STMIK-AMIK Riau. And master's degree in Master of Information Technology at Putra Indonesia University Padang. Now working as a lecturer in the Department of Computer Science, University Muhammadiyah of Riau. With research interests in the field of machine learning algorithms and AI. He can be contacted at email: januaralamien@umri.ac.id.