Indonesian Journal of Electrical Engineering and Computer Science
Vol. 38, No. 1, April 2025, pp. 569~577
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v38.i1.pp569-577

Tree-based models and hyperparameter optimization for assessing employee performance

Rendra Gustriansyah[1], Shinta Puspasari[1], Ahmad Sanmorino[2], Nazori Suhandi[1], Dewi Sartika[3]
[1]Department of Informatics Engineering, Faculty of Computer and Natural Science, Universitas Indo Global Mandiri, Palembang, Indonesia
[2]Department of Information System, Faculty of Computer and Natural Science, Universitas Indo Global Mandiri, Palembang, Indonesia
[3]Department of Informatics Engineering, Faculty of Computer Science, Universitas Sriwijaya, Palembang, Indonesia

Article Info

Article history:

Received May 8, 2024
Revised Oct 14, 2024
Accepted Oct 30, 2024

Keywords:

Decision tree
Employee performance
Extreme gradient boosting
Hyperparameter optimization
Machine learning
Random forest

ABSTRACT

The Palembang city fire and rescue service (FRS) is encountering challenges in adhering to national standards for fire response time. Hence, the Palembang city FRS is committed to enhancing employee performance through quarterly performance assessments based on various criteria such as attendance, work targets, behavior, education, and performance reports. This study proposes tree-based models in machine learning (ML) and hyperparameter optimization to assess the performance of Palembang city FRS employees. Tree-based models encompass decision trees (DT), random forests (RF), and extreme gradient boosting (XGB). The predictive performance of each model was evaluated using the confusion matrix (CM), the area under the receiver operating characteristic (AUROC), and the kappa coefficient (KC). The results indicate that RF performs better than DT and XGB in the sensitivity, AUROC, and KC metrics by 1.0000, 0.9874, and 0.8584, respectively.

Corresponding Author:

Rendra Gustriansyah
Department of Informatics Engineering, Faculty of Computer and Natural Science
Universitas Indo Global Mandiri
Street of Jenderal Sudirman No. 629, Palembang, 30129, Indonesia
Email: rendra@uigm.ac.id

1. INTRODUCTION

According to Permendagri 86 of 2017, the national standard for the fire and rescue service (FRS) performance is that the FRS must be able to handle a fire management area with a maximum response time of 15 minutes. However, the Palembang city FRS is not meeting this target. It is shown from the performance indicator that fire response time in 2022 will only reach 80 percent [1]. Therefore, the Palembang city FRS is committed to improving its performance in the next three years.

One way is to enhance employee performance. The Palembang city FRS has initiated employee performance assessments every quarter. To ensure a fair and objective assessment, FRS utilizes multiple criteria, including attendance, work targets, behavior, education, and performance reports. The results of the performance assessment are the basis for determining allowances. Therefore, an objective appraisal is needed to ensure the assessment team's findings align with each employee's accomplishments.

Machine learning (ML) methods can be applied as the right solution to reduce bias in performance assessment. ML can build models to identify specific patterns in a dataset, and models can be utilized to evaluate/predict new data. However, there are only a few implementations of ML to assess employee performance. Some studies use methods such as the Naïve Bayes (NB) [2]-[4], decision tree (DT) [2], [5], and K-nearest neighbor (KNN) [3]. Galih and Eriyadi [6] and Sartika and Gustriansyah [5] show that the

performance of the DT method outperforms the NB. Meanwhile, other studies generally use multi-criteria decision-making methods, including the analytical hierarchy process [7], [8], simple additive weighting [9], [10], technique for order preference by similarity to ideal solution [11], [12], profile matching [13], [14], weighted product [15], and others that must involve experts to assess employee performance. Besides that, there are several similar studies but for the case of student/academic performance assessment with numerical [16]-[18] and categorical data [19]-[21].

Therefore, this study proposes tree-based models in ML to assess the Palembang city FRS employees' performance. Tree-based models in ML include DT, random forest (RF), and extreme gradient boosting (XGB). DT in ML is like a flowchart that makes decisions based on data attributes and leads to a decision [5]. It is beneficial in situations that require straightforward and logical judgments. Meanwhile, pruning techniques are essential to ensure the DT model's accuracy. Therefore, they help decrease model complexity and prevent overfitting to achieve optimal performance.

RF is an ensemble learning method that combines several DTs to produce more accurate predictions [22]. In handling complex data sets, this method generally outperforms DT [23]. XGB, on the other hand, is another ensemble learning method with a different approach than RF [24]. RF builds trees independently, while XGB builds trees sequentially. Each new tree helps correct the mistakes made by the previous tree. It's as if each tree in the sequence learns from the previous tree's mistakes, resulting in a more accurate model. Additionally, hyperparameters will be fine-tuned to be optimal [25], and each model's predictive performance is evaluated using metrics such as accuracy, sensitivity, precision, specificity, area under receiver operating characteristic, and kappa coefficient (KC).

2. METHOD

The study's approach is based on the modified cross-industry standard process for data mining [26]. This model comprises five sequential processes, as depicted in Figure 1. The initial phase of this model is Business Understanding. This phase comprehends the business from various perspectives, such as application field, project goals, requirements, and management regulations.

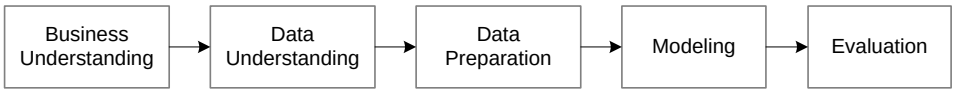


Figure 1. Research phases

2.1. Data understanding

The second phase involves collecting, describing, analyzing, and manipulating data using various techniques to familiarize users with the data. The Palembang city FRS dataset from the first quarter of 2023 (January to March) will be utilized for this study. The dataset has 218 employee records, each containing six attributes with values as presented in Table 1. Furthermore, the correlation between variables is verified using the rank correlation coefficient.

Table 1. Attributes description

Attributes	Values	Type
Attendance	Discipline/undisciplined	Categorical
Performance_report	Satisfied/dissatisfied	Categorical
Work_target	Needs improvement/good/very good	Categorical
Work_behavior	Needs improvement/good/very good	Categorical
Education	SMP/SMA (SMK)/D3/S1/S2	Categorical
Result	Worthy/unworthy	Categorical

2.2. Data preparation

All activities required to construct the final data set to be used in the modeling phase are covered in this phase. Data preparation is a crucial phase in enhancing model performance. During this process, invalid data, including empty, incomplete, or null data is removed from the dataset [27]. Additionally, all variables are converted to numeric values to facilitate calculations and then encoded as categorical. After this, the

dataset is randomly partitioned into two parts, with 80% of the data allocated for training and the remaining 20% for testing. The data balance across each class in the target variable (result) is carefully considered for the training and test data.

2.3. Modeling

2.3.1. Decision tree

DT is a method used for classification that represents data in a tree structure. Each attribute (*p*) is a node in the tree, represented by branches for its value and leaves for the class [5]. DT works by dividing data into smaller groups based on attribute values. Entropy is used to measure the uncertainty of data (*j*) at each branch division (*s*). This process is repeated recursively by dividing the groups into smaller groups. The process stops when all groups have high homogeneity. When there are no more attributes that can be used to divide the groups. In addition to entropy (*E*), information gain (*IG*) is also calculated. *IG* measures the difference in the entropy of data before and after dividing it based on a feature or attribute. A higher *IG* indicates better and more informative separation.

$$E(s) = \sum_{j=1}^{n} p_j \log_2 p_j \quad (1)$$

$$IG(p) = E(s) - \sum_{i=1}^{n} \frac{|s_i|}{|s|} E(s_i) \quad (2)$$

2.3.2. Random forest

RF is a supervised ML method that uses DT techniques to categorize data [22]. This method consists of a collection of DTs that are trained randomly and assign labels to each leaf node. Each DT in the set is built using different data from the original dataset. RF uses a majority vote to determine the final label. The two most crucial parameters in RF are ntree and mtry. "ntree" defines the number of trees in the model, and "mtry" defines how many labels are required in each step. These two parameters produce the explained variance and percentage error rate. This method has several advantages, including high accuracy, resistance to overfitting, and effortless interpretation.

The process of how RF works can be explained in the following steps:

Step 1: randomly select *j* samples from the training dataset using bootstrap resampling with replacement.

Step 2: randomly select *m* attributes from *p* attributes (where $m \leq p$).

Step 3: construct a DT and then split the best node(s) from the DT using the Gini index:

$$Gini(T) = 1 - \sum_{j=1}^{m} p_j^2 \quad (3)$$

Step 4: repeat steps 1-3 until a forest of n trees (ntree) is formed and the optimal mtry is obtained.

Step 5: use a majority vote to determine the label of a leaf node (class).

2.3.3. Extreme gradient boosting

XGB is an ensemble ML method used for classification and regression tasks. It utilizes the outputs of multiple DT to create a more precise and robust model [24]. In the XGB method, each DT is trained to reduce the prediction error of the previous model. The newly added DT learns from the prediction errors incurred by the previous model and tries to minimize those errors. XGB has several advantages, including high efficiency, accuracy, and stability.

These are the steps for building an XGB model:

Step 1: set the initial prediction probability (Pr_i) as the base output, with $i = 1, 2, \ldots, n$.

Step 2: compute the residual error (*RE*) based on the actual output (Y_i).

$$RE_i^t = Y_i - Pr_i^t \quad (4)$$

Step 3: construct a binary DT.

Step 4: evaluate the similarity score (*SS*).

$$SS_{node} = \frac{\left(\sum_{i=1}^{n} RE_i^t\right)^2}{\sum_{i=1}^{n} Pr_i^t\left(1-Pr_i^t\right) + \lambda} \quad (5)$$

Step 5: determine the *Gain* for each attribute. Select the attribute with the highest *Gain* to build the DT.

$$Gain(p) = SS_{left} + SS_{right} - SS_{root} \quad (6)$$

Step 6: expand the DT and determine the leaf output (*Lo*) by applying optimization parameters (λ).

$$Lo_i = \frac{\sum_{i=1}^{n} RE_i^t}{\sum_{i=1}^{n} Pr_i^t\left(1-Pr_i^t\right)+\lambda} \quad (7)$$

Step 7: calculate the log odds value of the base model.

$$\log odds_i^t = \log\left(\frac{Pr_i^t}{1-Pr_i^t}\right) \quad (8)$$

Step 8: compute new probabilities for each record and normalize them using the learning rate (η).

$$Pr_i^{t+1} = \log odds_i^t + \eta \,.\, Lo_i \quad (9)$$

Step 9: standardize the probability value using the binary sigmoid function (*Sig*).

$$Sig(Pr_i^{t+1}) = \frac{e^{Pr_i^{t+1}}}{1+e^{Pr_i^{t+1}}} \quad (10)$$

Step 10: recalculate the RE using the new Pr_i^{t+1} and repeat the process from step 2 to step 9 to create a new DT.

2.3.4. Hyperparameter

The hyperparameter fine-tuning phase is crucial in determining the optimal model parameters before the training process begins [25]. Each model involves unique sets of hyperparameters [28]. The 'mlr' library in R was used to optimize these hyperparameters.

2.4. Evaluation

2.4.1. Confusion matrix

Confusion matrix (CM) consists of four categories: true positive (TP), false positive (FP), true negative (TN), and false negative (FN) as presented in Table 2 [22]. CM is the basis for assessing several validation metrics as presented in Table 3, such as accuracy (1), precision (2), sensitivity (3), and specificity (4), that are used to evaluate models [29]. Precision is a metric to measure the ratio of correct positive predictions. Meanwhile, sensitivity measures the model's ability to identify all positive cases. Specificity, on the other hand, identifies true negative results. Accuracy measures the ratio of correct predictions, both positive and negative, out of all the data. It is important to select appropriate metrics based on specific scenarios to evaluate the model's performance accurately.

Table 2. The confusion matrix

	Actual "Worthy"	Actual "Unworthy"
Predicted "Worthy"	TP	FP
Predicted "Unworthy"	FN	TN

Table 3. The validation metrics

Metrics	Equation	
Accuracy	$TP+TN/(TP+TN+FP+FN)$	(1)
Precision	$TP/(TP+FP)$	(2)
Sensitivity	$TP/(TP+FN)$	(3)
Specificity	$TN/(TN+FP)$	(4)

2.4.2. Area under the receiver operating characteristics curve

area under the receiver operating characteristic (AUROC) is a metric used to evaluate the overall performance of a classification system. It determines a threshold value and measures the model's ability to differentiate between two classes or groups accurately. In a multi-class study, a "positive" class is designated, and the remaining classes are defined as "negative." The AUROC value of a model increases as its accuracy in predicting the class improves, with a value closer to 1 indicating better accuracy [30].

2.4.3. Kappa coefficient

KC is used to quantify inter-rater reliability or agreement between annotators when performing a classification task. KC can be utilized for classification tasks involving several classes or imbalanced class distributions. The coefficient values are 0 for low agreement and 1 for perfect agreement [31]. It is important to note that the KC should only be used to measure outcomes that involve qualitative (categorical) data.

3. RESULTS AND DISCUSSION

3.1. Data understanding

In this phase, the class composition of the target variable (result) is presented for the 218 employees in the dataset in Table 4. The results indicate that the class composition is relatively balanced among the different classes. Besides, the relationships between variables are shown in Figure 2. The results demonstrate that the education variable most dominantly influences the result variable. It can assist in determining the root variable for the DT and RF methods.

Table 4. The class composition of the result variable

	Worthy	Unworthy
No of employee	128	90

Figure 2. The relationships between variables

3.2. Data preparation

In this phase, verification is performed for empty and null data. There is no data reduction in this phase. Furthermore, the dataset is randomly partitioned into training and testing data in a ratio of 80:20. The class balance for each dataset is verified to minimize any bias. Figure 3 illustrates the class composition of the result variable for both the training and testing data. Figure 3(a) depicts a class composition of 59:41 for the training data, while Figure 3(b) shows a slightly improved class composition for the testing data. However, both compositions indicate that the class composition is relatively balanced.

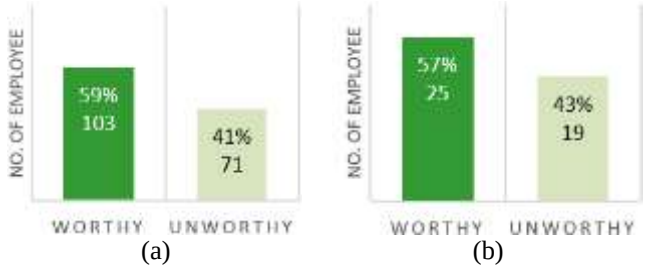


Figure 3. The class composition of the result variable for (a) training data and (b) test data

3.3. Hyperparameter optimization

The optimal hyperparameter configurations obtained using the 'mlr' library for each model are listed in Table 5. The hyperparameters fine-tuning for the DT and RF methods is straightforward. In contrast, the XGB method's hyperparameter fine-tuning is more comprehensive.

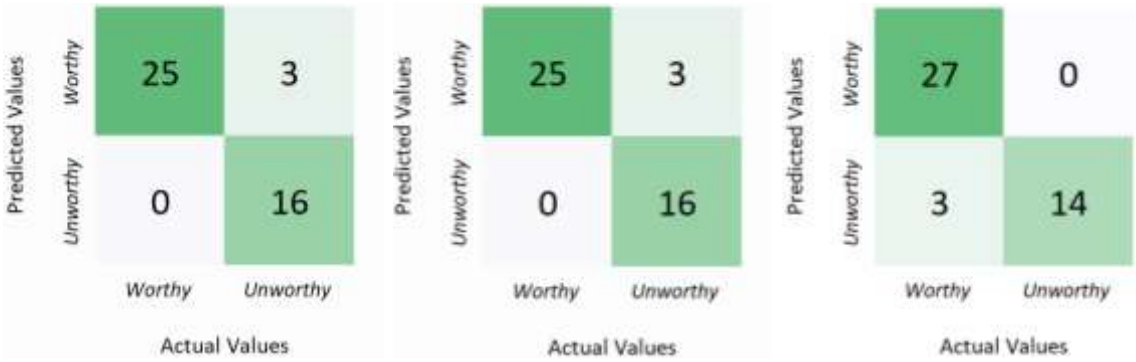
Table 5. The hyperparameters fine-tuning in ML methods

Method	Hyperparameters
DT	Maxdepth=23, cp=0.009, minsplit=8
RF	Ntree=10, mtry=2
XGB	Eta=0.379, max_depth=3, min_child_weight=2.59, colsample_bytree=0.969, colsample_bylevel=0.863, subsample=0.966, nrounds=139, booster=gbtree

3.4. Evaluation

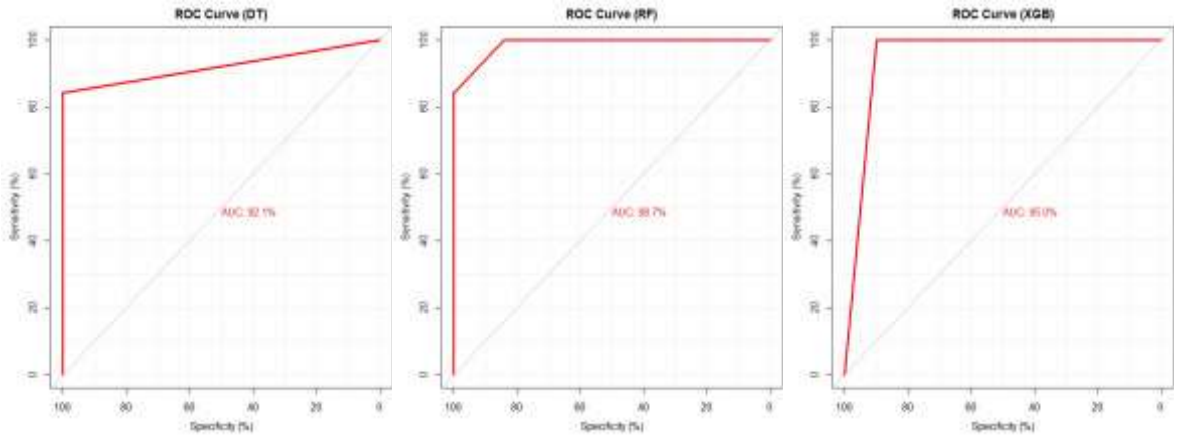
We employed CM, AUCROC, and KC to assess the performance of the classification models. The CM delineates the classification performance of each model using the test data, as depicted in Figure 4. Figures 4(a) and 4(b) exemplify the CM results for the DT and RF models; conversely, Figure 4(c) illustrates the marginally different CM yielded by the XGB model.

In addition, the AUROC results for each model are depicted in Figure 5. Figures 5(a)-5(c) present the AUROC values for the DT, RF, and XGB models, all of which are notably high (above 90%). However, the AUROC achieved by the RF model outperforms the other two models.



(a) (b) (c)

Figure 4. The CM for: (a) DT, (b) RF, and (b) XGB models



(a) (b) (c)

Figure 5. The AUROC for: (a) DT, (b) RF, and (b) XGB models

Based on CM, accuracy, precision, sensitivity, and specificity metrics can be determined for each model. Furthermore, the evaluation results based on metrics for each model are shown in Table 6. These results prove that RF has better performance than DT and XGB. Because it got the best scores on three metrics: sensitivity, AUROC, and KC. Meanwhile, DT and XGB are only superior in two metrics.

Table 6. The evaluation results

Metrics	DT	RF	XGB
Accuracy	0.9318	0.9318	0.9318
Sensitivity	**1.0000**	**1.0000**	0.9000
Precision	0.8929	0.8929	**1.0000**
Specificity	0.8421	0.8421	**1.0000**
AUROC	0.9211	**0.9874**	0.9500
KC	**0.8584**	**0.8584**	0.8514

If observed, the accuracy level for all three tree-based models is the same, indicating that the accuracy metric cannot be utilized to measure model performance. In addition, the CM approach used is not appropriate if we compare the DT and RF as they both produce the same performance. However, additional studies with various datasets are needed to support this conclusion. In this case, the AUROC metric is the most appropriate for measuring model performance as it shows the results' differentiation. The proposed method in this study shows superior performance (accuracy) compared to previous studies [5], [6].

This study also found that RF as a tree-based model in ML has worked better to solve employee performance assessment issues. The RF model uses a combination of many trees to produce more accurate and robust results compared to the DT method. Meanwhile, the XGB method builds trees one by one and uses each new tree to correct mistakes made by the previous tree, resulting in highly accurate models. However, XGB may require more sensitivity and adjustment compared to RF.

Tree-based models in ML with hyperparameter optimization have produced excellent performance in assessing the performance of DKP employees in Palembang city. These findings could be a foundational step for future research. However, this study has limitations as it is a specific case study. Therefore, the results must be generalized by implementing them in various other cases. In addition, variable selection methods can also be tried to enhance the model's results and gain a better perspective on crucial variables.

4. CONCLUSION

This study highlights the significance of using tree-based models in ML techniques such as DT, RF, and XGB in evaluating employee performance for the Palembang city FRS. The research findings indicate that tree-based models in ML exhibit high performance in resolving employee performance assessment issues. The graduate variable is the most significant factor influencing the result variable. Of the three tree-based models, the RF model performs better than DT and XGB in addressing employee performance evaluation issues. RF's superiority is demonstrated in the sensitivity, AUROC, and KC values by 1.0000, 0.9874, and 0.8584, respectively. These findings could serve as a crucial step for inspiring future research. However, to achieve generalizability, it is necessary to apply these models to various other cases. Finally, new ensemble and hybrid models can be introduced to achieve higher performance.

ACKNOWLEDGEMENTS

The authors would like to extend my heartfelt appreciation to the head of the Palembang city FRS for his valuable cooperation, especially in providing the dataset for this study. The authors also extend their appreciation for the invaluable support received from Universitas Indo Global Mandiri.

REFERENCES

[1] Pemerintah-Kota-Palembang, "DPK strategic plan and saving the city of Palembang for 2024-2026 (In Indonesian: renstra DPK dan penyelamatan Kota Palembang tahun 2024-2026)," 2023. [Online]. Available: http://esakip.palembang.go.id/1823/dokumen/116/2023/f117cb7d5b411d63656a5c465ef01047.pdf

[2] A. S. De Oliveira Goes and R. C. L. De Oliveira, "A Process for human resource performance evaluation using computational intelligence: an approach using a combination of rule-based classifiers and supervised learning algorithms," *IEEE Access*, vol. 8, pp. 39403–39419, 2020, doi: 10.1109/ACCESS.2020.2975485.

[3] K. G. Ayu *et al.*, "Classification of employee competency assessment using Naïve Bayes and K-nearest neighbor (KNN) algorithms," *Journal of Advances in Information Technology*, vol. 15, no. 7, pp. 879–885, 2024, doi: 10.12720/jait.15.7.879-885.

[4] H. Hindayanti, W. Septiani, and R. Maulidya, "Employee performance assessment model in survey service companies using analytical hierarchy process and Naive Bayes," *Spektrum Industri*, vol. 22, no. 1, pp. 36–50, May 2024, doi: 10.12928/si.v22i1.150.
[5] D. Sartika and R. Gustriansyah, "Comparison of naive Bayes and decision tree algorithms to assess the performance of Palembang City fire and Disaster management employees," *TEKNOSAINS: Jurnal Sains, Teknologi dan Informatika*, vol. 11, no. 1, pp. 132–138, 2024, doi: 10.37373/tekno.v11i1.843.
[6] G. Galih and M. Eriyadi, "Comparison of the NBC, SVM, and C4.5 models in measuring outstanding employee performance post the Covid-19 pandemic (In Indonesian: *Perbandingan model NBC, SVM, dan C4.5 dalam mengukur kinerja karyawan berprestasi pasca pandemi COVID-19*)," *Jurnal Informatika*, vol. 9, no. 2, pp. 123–130, Oct. 2022, doi: 10.31294/inf.v9i2.13772.
[7] S. G. Fashoto, O. Amaonwu, and A. Afolorunsho, "Development of a decision support system on employee performance appraisal using AHP model," *JOIV : International Journal on Informatics Visualization*, vol. 2, no. 4, pp. 262–267, Aug. 2018, doi: 10.30630/joiv.2.4.160.
[8] B. Laia and B. Sinaga, "Decision support system for employee performance using AHP method (case study: PT. Andhy Putra)," *International Journal of Basic and Applied Science*, vol. 10, no. 3, pp. 107–116, Dec. 2021, doi: 10.35335/ijobas.v10i3.33.
[9] R. Gustriansyah, J. Alie, and N. Suhandi, "Performance evaluation of contract employees using the best-worst and simple additive weighting methods," *JUITA: Jurnal Informatika*, vol. 9, no. 2, pp. 219–227, Nov. 2021, doi: 10.30595/juita.v9i2.11989.
[10] Y. P. Suprapto, H. Haerudin, and A. Danuwidodo, "Decision support system for employee performance assessment using analytical hierarchy process and simple additive weighting methods," *Journal of Information Systems and Informatics*, vol. 6, no. 2, pp. 766–780, Jun. 2024, doi: 10.51519/journalisi.v6i2.721.
[11] T. B. Febrian and A. Simangunsong, "Decision support system employee performance appraisal method using TOPSIS," *Journal Of Computer Networks, Architecture and High Performance Computing*, vol. 2, no. 2, pp. 307–312, Jun. 2020, doi: 10.47709/cnapc.v2i2.412.
[12] M. Lingga, "Evaluating the performance employee using TOPSIS," *IOP Conference Series: Materials Science and Engineering*, vol. 662, no. 6, pp. 1–8, Nov. 2019, doi: 10.1088/1757-899X/662/6/062018.
[13] N. P. Dewi, N. Ramadhani, I. Darmawan, U. Ubaidi, and A. W. Syahroni, "Application of profile matching in determining employee annual bonuses," *Jurnal Informasi dan Teknologi*, vol. 6, no. 2, pp. 133–140, Jun. 2024, doi: 10.60083/jidt.v6i2.542.
[14] Safrizal, L. Tanti, R. Puspasari, and B. Triandi, "Employee performance assessment with profile matching method," in *2018 6th International Conference on Cyber and IT Service Management (CITSM)*, IEEE, Aug. 2018, pp. 1–6. doi: 10.1109/CITSM.2018.8674256.
[15] R. T. A. Agus, M. A. Sembiring, M. F. L. Sibuea, Mardalius, and A. Nata, "Employee performance analysis using analytical hierarchy process and weighted product combinations algorithm," *Journal of Physics: Conference Series*, vol. 1783, no. 1, pp. 1–4, Feb. 2021, doi: 10.1088/1742-6596/1783/1/012010.
[16] B. Sekeroglu, K. Dimililer, and K. Tuncal, "Student performance prediction and classification using machine learning algorithms," in *Proceedings of the 2019 8th International Conference on Educational and Information Technology*, New York, NY, USA: ACM, Mar. 2019, pp. 7–11. doi: 10.1145/3318396.3318419.
[17] A. Qazdar, B. Er-Raha, C. Cherkaoui, and D. Mammass, "A machine learning algorithm framework for predicting students performance: A case study of baccalaureate students in Morocco," *Education and Information Technologies*, vol. 24, no. 6, pp. 3577–3589, Nov. 2019, doi: 10.1007/s10639-019-09946-8.
[18] X. Xu, J. Wang, H. Peng, and R. Wu, "Prediction of academic performance associated with internet usage behaviors using machine learning algorithms," *Computers in Human Behavior*, vol. 98, pp. 166–173, Sep. 2019, doi: 10.1016/j.chb.2019.04.015.
[19] A. S. Hashim, W. A.Awadh, and A. K. Hamoud, "Student performance prediction model based on supervised machine learning algorithms," in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, Nov. 2020, pp. 1–18. doi: 10.1088/1757-899X/928/3/032019.
[20] R. Ghorbani and R. Ghousi, "Comparing different resampling methods in predicting students' performance using machine learning techniques," *IEEE Access*, vol. 8, pp. 67899–67911, 2020, doi: 10.1109/ACCESS.2020.2986809.
[21] B. Albreiki, N. Zaki, and H. Alashwal, "A systematic literature review of student' performance prediction using machine learning techniques," *Education Sciences*, vol. 11, no. 9, pp. 1–27, Sep. 2021, doi: 10.3390/educsci11090552.
[22] R. Gustriansyah, N. Suhandi, S. Puspasari, and A. Sanmorino, "Machine learning method to predict the toddlers' nutritional status," *INFOTEL*, vol. 16, no. 1, pp. 1–6, 2024, doi: 10.20895/infotel.v15i4.988.
[23] V. K. Gupta, A. Gupta, D. Kumar, and A. Sardana, "Prediction of COVID-19 confirmed, death, and cured cases in India using random forest model," *Big Data Mining and Analytics*, vol. 4, no. 2, pp. 116–123, Jun. 2021, doi: 10.26599/BDMA.2020.9020016.
[24] V. Pradeep, R. Khemmar, and F. Jendoubi, "Visual eco-routing (VER): XGBoost based eco-route selection from road scenes and vehicle emissions," *IEEE Access*, vol. 12, pp. 9669–9681, 2024, doi: 10.1109/ACCESS.2024.3353036.
[25] E. Elvin and A. Wibowo, "Forecasting water quality through machine learning and hyperparameter optimization," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 33, no. 1, pp. 496–506, Jan. 2024, doi: 10.11591/ijeecs.v33.i1.pp496-506.
[26] J. Saltz, I. Shamshurin, and K. Crowston, "Comparing data science project management methodologies via a controlled experiment," in *Proceedings of the 50th Hawaii International Conference on System Sciences*, 2017, pp. 1013–1022. doi: 10.24251/HICSS.2017.120.
[27] R. Gustriansyah, E. Ermatita, and D. P. Rini, "An approach for sales forecasting," *Expert Systems with Applications*, vol. 207, pp. 1–8, Nov. 2022, doi: 10.1016/j.eswa.2022.118043.
[28] R. H. Hridoy, A. D. Arni, S. K. Ghosh, N. R. Chakraborty, and I. Mahmud, "Performance enhancement of machine learning algorithm for breast cancer diagnosis using hyperparameter optimization," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 14, no. 2, pp. 2181–2190, Apr. 2024, doi: 10.11591/ijece.v14i2.pp2181-2190.
[29] A. Sanmorino and H. Di Kesuma, "Fine-tuning a pre-trained ResNet50 model to detect distributed denial of service attack," *Bulletin of Electrical Engineering and Informatics*, vol. 13, no. 2, pp. 1362–1370, Apr. 2024, doi: 10.11591/eei.v13i2.7014.
[30] T. Kim and J.-S. Lee, "Maximizing AUC to learn weighted naive Bayes for imbalanced data classification," *Expert Systems with Applications*, vol. 217, pp. 1–17, May 2023, doi: 10.1016/j.eswa.2023.119564.
[31] S. Rafieyan, E. Vasheghani-Farahani, N. Baheiraei, and H. Keshavarz, "MLATE: machine learning for predicting cell behavior on cardiac tissue engineering scaffolds," *Computers in Biology and Medicine*, vol. 158, pp. 1–11, May 2023, doi: 10.1016/j.compbiomed.2023.106804.

BIOGRAPHIES OF AUTHORS

Rendra Gustriansyah is an associate professor at Universitas Indo Global Mandiri, Indonesia. He completed his master's and doctoral degrees with BPPDN Scholarships from the Ministry of Research, Technology, and Higher Education for Master's in computer science from the University of Indonesia and a doctoral in informatics engineering from Sriwijaya University. His research areas are machine learning, data mining, decision-making, and business intelligence. He can be contacted at email: rendra@uigm.ac.id.

Shinta Puspasari received the S.Si. degree in mathematics from the Universitas Sriwijaya, the M.Kom. degree in computing from the Universitas Indonesia, and the doctorate in informatics engineering from the Universitas Sriwijaya, Indonesia. She holds the Head of Institutes for Research and Community Service, Universitas Indo Global Mandiri. She is currently an associate professor with the Department of Informatics Engineering, Universitas Indo Global Mandiri (UIGM). Her research interests include computer vision and intelligent systems. She can be contacted at email: shinta@uigm.ac.id.

Ahmad Sanmorino is an associate professor at the Faculty of Computer and Science, Universitas Indo Global Mandiri, Indonesia. He received a doctorate in informatics engineering from Universitas Sriwijaya. His research interests include artificial intelligence area including applied machine learning, cyber security, and information security. He can be contacted at email: sanmorino@uigm.ac.id.

Nazori Suhandi is a senior lecturer at the Faculty of Computer and Science, Universitas Indo Global Mandiri. He received a master's degree in management from Universitas Sriwijaya. He has research interests in business intelligence and entrepreneurship. He can be contacted at email: nazori@uigm.ac.id.

Dewi Sartika is currently a lecturer at the Faculty of Computer Science, Sriwijaya University, Indonesia. She completed her degree for bachelor of informatics engineering at Sriwijaya University and master of computer science at Universitas Indonesia. She has research interests in software engineering and computational intelligence. She can be contacted at email: dewisartika@unsri.ac.id.