TARGET: Navigating Class Imbalance with a Target Reduction Clustering Approach

Md Rakibul Hasan 1, a), Hui-Ngo Goh 1, 2, b), Choo-Yee Ting 1, 3, c), Albert Quek 1, 4, d) and Radiah Haque 1, e)

1Faculty of Computing & Informatics, Multimedia University Persiaran Multimedia, 63100 Cyberjaya, Selangor, Malaysia

2Centre for Natural Language Processing, CoE for Artificial Intelligence, Multimedia University, 63100 Cyberjaya, Selangor, Malaysia

3Centre for Big Data and Blockchain Technologies, CoE for Advanced Cloud, Multimedia University, 63100 Cyberjaya, Selangor, Malaysia

4Centre for Image and Vision Computing, CoE for Artificial Intelligence, Multimedia University, 63100 Cyberjaya, Selangor, Malaysia

b) Corresponding author: hngoh@mmu.edu.my

a) rakibulh170@gmail.com

c) cyting@mmu.edu.my

d) quek.albert@mmu.edu.my

e) radumoni2@gmail.com

**Abstract**. Multi-class classification over small, imbalanced datasets is a very challenging problem, particularly if minority classes are poorly represented. It is a relevant problem in the domain of student employment prediction, where accurate classification into various categories of jobs is highly important for optimal career guidance. Traditional machine learning classifiers tend to be biased towards majority classes, resulting in weak generalization and poor performance with regards to minority classes. This article presents TARGET, a target reduction method called target-mnm, that clusters class labels into two classes of "Major" and "Minor," according to a specified frequency threshold. The key contribution is to enhance classification efficacy by reducing class imbalance through label grouping. Five classifiers, namely Logistic Regression (LR), Support Vector Machine (SVM), Extreme Gradient Boosting (XGB), Categorical Boosting (CB), and Random Forest (RF), are utilized for the classification of a real-world student employment dataset characterized by its extreme imbalance and small size. The suggested target-mnm approach outperforms the baseline result with a considerable improvement in accuracy (0.879). The experiments demonstrate the efficacy of targeted class clustering for managing imbalanced multi-class issues, providing a feasible solution in the context of student employment prediction.

# Introduction

Classification of instances in small and imbalanced datasets is particularly challenging, especially in multi-class scenarios with skewed class distributions [1]. In such scenarios, machine learning algorithms generally develop a bias towards majority classes, resulting in poor performance with respect to minority classes [2]. This issue is common in domains such as healthcare, fraud detection, and image classification, where class imbalance can be extreme, such as a 1000:1 ratio in credit card fraud detection [3]. Moreover, metrics like accuracy can be misleading, as models may achieve high accuracy while failing to predict minority class instances effectively [4]. This study addresses the problem of class imbalance in student employment prediction. The dataset contains 11 job sectors with highly unequal representation. Accurate prediction in this context is essential for generating reliable career counseling insights and for matching students to appropriate job sectors.

However, class imbalance impairs model fairness and accuracy, limiting its practical value [5]. Initially, we applied conventional data balancing techniques such as oversampling [6], but these approaches did not improve classification performance significantly. To overcome this limitation, we propose a target reduction strategy named target-mnm, which groups classes into two categories of "Major" and "Minor," based on their frequencies. This targeted clustering approach simplifies the classification task and balances class representation inside the groups, thereby enhancing model performance. The main objective of this research is to evaluate the effectiveness of the proposed target-mnm method in improving classification metrics on an imbalanced, multi-class dataset. The method is tested using five well-established machine learning algorithms on a real-world student employment dataset. The findings reveal notable improvements in accuracy, precision, recall, and F1-score compared to baseline models without target reduction. The key contributions of this study are as follows:

* Identification of the limitations of traditional balancing techniques for small, imbalanced, multi-class datasets.
* Proposal of a target reduction method (target-mnm) that consolidates infrequent classes to address imbalance
* Empirical validation of the proposed method using multiple classifiers on a real-world student employment dataset.

# Related Works

Current research has tackled imbalanced dataset classification, particularly small sets. Patel and Bhavsar [7] proposed the Kernel Trick for Imbalanced Data (KTI), improving Support Vector Machines (SVM) for binary datasets and realizing substantial performance improvements. For multiclass imbalanced classification, Pant and Das [8] proposed the MCMRC\_IB algorithm, an extension of the MCMRC method, which refined micro-clustering processes to improve predictive accuracy, F-measure, and G-mean, although they did not explicitly discuss its limitations. Wang and Awang [9] developed MKC-SMOTE, a synthetic oversampling technique combining over- and under-sampling strategies to generate high-quality minority samples, resulting in significant classification gains on real-world datasets. Zhou et al. [10] suggested the Split Difference Decision Tree (SDDT) and Weighted Split Difference Classification and Regression Tree (WSD-CART) with a new split index on the Class Key Decision (CKD) factor for better node splitting feature representation, but without taking computational complexity into account. Manjula and Layaq [11] suggested IDROSUS, a high-performance hybrid imbalanced data reduction technique for software defect prediction without testing on various classifiers or multiclass problems. Acharya et al. [12] utilized Bagging and AdaBoost for network intrusion detection with effectiveness on binary and multiclass problems but referring to the high computational cost of deep learning models. Jiang et al. [13] utilized XGB for multiclass classification (MCC) and recognized the need for manual parameter tuning. Ramirez et al. [14] applied SVM to MCC and stated that PRC and F1 measures can be negatively impacted by low true positive rates in imbalanced data. They suggested class-wise testing with binary confusion matrices and underlined the importance of root causes of imbalance identification. The summarized results of these studies are provided in Table 1.

These existing approaches either focus solely on binary classifications or fail to adequately address the unique challenges posed by multiclass imbalances, resulting in insufficient generalization to real-world applications. Additionally, issues such as computational complexity, the necessity for manual parameter tuning, and the inadequacy of performance metrics in reflecting true model efficacy further complicate the landscape of imbalanced dataset classification. By recognizing these shortcomings, our research aims to propose a method that addresses these critical gaps, ultimately enhancing classification performance for imbalanced datasets.

# Methodology

The primary goal of this study is to enhance MCC's performance by implementing a target reduction method that addresses the challenges associated with imbalanced datasets. Figure 1 illustrates the streamlined methodology employed in this study, providing a clear visual representation of the processes involved.

## Dataset

This study expands our previous study [15]that utilized the same datasets, demographic (Ddmg) and academic (Dacd) with attributes, including Permanent Address State, Nationality, Race, Gender, Disability Status, Marital Status, Campus, Study Program, Credit Transfer, Trimester GPAs (T1–T11), Faculty Domain, Sponsor Category, Entry Eligibility, MUET Score, and grades in core subjects like Sejarah, Malay Language, English, Mathematics, Additional Mathematics, Physics, Biology, Moral Studies, Accounting, Science, Chemistry, and Chinese Language (Bahasa Ci).

The Employability Dataset (Demp) keeps track of alumni job titles and employers, while the Job-Street Job Requirements Dataset (Djjr), which was derived from job advertisements crawled in September 2023, has job titles, sectors, sub-sectors, and job descriptions. D*emp* job titles were labeled based on Djjr's 11 job sectors, namely Accounting/Finance, Admin/Human Resources, Arts/Media/Communications, Engineering, Computer/IT, Education/Training, Manufacturing, Hotel/Restaurant, Sales/Marketing, Services, and Others.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TABLE 1**. Analyzed results and findings from previous studies | | | | |
| **Ref** | **Method** | **Findings** | **Future work** | **Limitation** |
| [7] | KTI | Handling binary class  imbalanced dataset | Expanding the study  from binary to MCC tasks | - |
| [8] | MCMRC\_IB | MCC on an imbalanced dataset | - | - |
| [9] | SVM | Multi-class imbalanced data classification | Investigating the use of different distance metrics. | Addressing the limitations of handling various types of data attributes |
| [10] | SDDT | Imbalanced classification | Investigating split difference applicability in diverse ML models | The computational complexity of the algorithms is noticed |
| [11] | IDROSUS | Imbalanced data reduction | Evaluating performance with other classifiers. | Addressing the limitation of two-class applicability. |
| [12] | Bagging and AdaBoost | Binary and Multi-Class Network Intrusion Detection | Exploring more efficient methods for handling large NIDS datasets. | Requiring high computing power for deep learning methods limits practicality. |
| [13] | XGB | MCC | Developing Auto ML for automatic parameter tuning | Requiring manual selection |
| [14] | SVM | MCC | Proposing performance evaluation by class using binary confusion matrices. | Analyzing class imbalance causes metrics like PRC and F1 to penalize low TP proportions |

A screenshot of a computer

Description automatically generated

**Figure 1.** Overview of the target reduction strategy for MCC

## Preprocessing and Data Transformation

Preprocessing ensures the dataset quality by removing unwanted symbols, untidy spacing, and irrelevant punctuation to make the data more readable. Plurals were made standard, and the job positions labels were normalized with the dataset of Djjr for consistency and structural readability in the dataset. As part of this process, data transformation is the process of converting unprocessed or semi-structured data into a more analysis-ready form [16]. It entails the categorization of raw job titles and their assignment to broad categories using the Djjr taxonomy, 11 sectors and 27 sub-sectors. Job titles were cleaned and mapped against entries of the Djjr using the Levenshtein Distance algorithm [17], marking the titles with similarity of 80% and above. Titles that did not match were categorized systematically based on context information extracted from the Dacd and Demp datasets. The overall approach allowed for accurate and comprehensive categorization.

## Target Reduction Method

The experimental dataset has 3177 records across 11 job sectors and is imbalanced. Computer/IT is the largest sector (761 records), while Hotel/Restaurant is the smallest (40 records). To reduce complexity, this study aims at classifying 11 job sectors. A target-by-reduction strategy simplifies MCC by grouping sectors into “Major” and “Minor” classes based on a 10% representation threshold identified through exploratory data analysis (EDA) [18]. Sectors with ≥10% presence are labeled as “Major” and others are grouped into “Minor” improving classifier performance and computational efficiency in imbalanced data.

Before finalizing this binary grouping, an ablation process tested several preliminary groupings:

* **Target-4:** Job sectors were initially divided into four groups to achieve an approximately equal distribution of records across each group. This broad categorization tested model performance and ensured balanced sector representation.
* Group-1 included Computer/Information Technology (761 records) and Manufacturing (48 records), totaling 809 records.
* Group-2 consisted of Services (515), Arts/Media/Communications (206), and Others (58), totaling 779 records.
* Group-3 comprised Accounting/Finance (512), Engineering (208), and Education/Training (73), with a total of 793 records.
* Group-4 combined Admin/Human Resources (395), Sales/Marketing (361), and Hotel/Restaurant (40), totaling 796 records.
* **Target-3:** Based on insights from Target-4, job sectors were further consolidated into three groups.
* Group-1 contained Computer/Information Technology (761), Manufacturing (48), Hotel/Restaurant (40), Education/Training (73), and Others (58), amounting to 980 records.
* Group-2 included Services (515) and Accounting/Finance (512), totaling 1,027 records.
* Group-3 grouped Admin/Human Resources (395), Arts/Media/Communications (206), Sales/Marketing (361), and Engineering (208), totaling 1,170 records.
* **Target-2:** Sectors with overlapping features and record counts were combined into two groups.
* Group-1 consisted of Computer/Information Technology (761), Manufacturing (48), Services (515), Arts/Media/Communications (206), and Others (58), with 1,588 records.
* Group-2 included Accounting/Finance (512), Engineering (208), Education/Training (73), Admin/Human Resources (395), Sales/Marketing (361), and Hotel/Restaurant (40), totaling 1,589 records.

The initial groupings also demonstrated the effect of class reduction on classification measures and model stability. From EDA insights, the Target-mnm simplifies the classification task by combining job sectors that not only are conceptually close but also have relatively similar sizes. Table 2 indicates job sector distribution under target-mnm.

This study employs a two-stage ML classification system to effectively categorize job sectors. Stage-1, the model classifies job sectors into “Major” and “Minor”. Stage-2, separate models are trained for each category to identify specific job sectors. This targeted training approach enhances prediction accuracy and ensures optimal handling of the class imbalance. The pseudocode from algorithm 1 outlines predicting job sectors based on the two-stage classification system. For handling classification issues in small, imbalanced datasets, we employed several ML models: LR, SVM, XGB, CB, and RF. LR was employed as a simple, interpretable baseline with one-vs-rest (OvR) for multi-class classification [18]. SVM, thanks to its robustness in high-dimensional space, was employed with the RBF kernel and class-weight balancing [19]. XGB exploited weighted loss functions and boosted trees for optimal minority class performance [20], while CB, through its native support of categorical features and class imbalance, effectively captured feature interactions [21]. RF, through its ensemble and bootstrap sampling properties, displayed strong performance on small datasets with no bias or overfitting [22].

|  |  |  |  |
| --- | --- | --- | --- |
| **TABLE 2**. Target-mnm group assigning | | | |
| **Group** | **Job Sector** | **Total Records** | |
| Major | Computer/Information Technology (761), Accounting/Finance (512), Admin/Human Resources (395), Service (515), Sales/Marketing (361) | | 2544 |
| Minor | Engineering (208), Arts/Media/Communication (206), Education/Training (73), Others (58), Manufacturing (48), Hotel/Restaurant (40) | | 633 |

|  |  |
| --- | --- |
| **ALGORITHM 1.** Assign job sector and sub-sector to technician job position | |
| 1: | **Load** dataset |
| 2: | **Define** features as [ddmg, dacd] |
| 3: | **Define** target as [target-mnm] |
| 4: | **Define** test\_features as [ddmg, dacd] |
| 5: | **Train** train\_target\_mnm |
| 6: | **Separate** dataset **into**: |
| 7: | major\_group **where** target-mnm is ‘Major’ |
| 8: | minor\_group **where** target-mnm is ‘Minor’ |
| 9: | **Set** target to [Job Sectors] |
| 10: | **Train** major group |
| 11: | **Train** minor group |
| 12: | Predict target mnm ← models prediction(test\_features) |
| 13: | **if** predict\_target\_mnm == ‘Major’ **then** |
| 14: | predict major job sectors ← models prediction(test\_features), evaluation(major) |
| 15: | **else** |
| 16: | predict minor job sectors ← models prediction(test\_features), evaluation (minor) |
| 17: | **end if** |
| 18: | result← average(evaluation(target-mnm), evaluation(major) **or** evaluation (minor)) |

Each model's performance was assessed by confusion matrix metrics (TP, TN, FP, FN) to estimate accuracy, precision, recall, and F1-score [23]. Accuracy measured overall correctness, precision specified accuracy of positive predictions, recall assessed sensitivity to positives, and the F1-score balanced precision and recall. Class-separation ability was also assessed with ROC curves and AUC values [24]. Among all models, CB consistently outperformed others across all metrics, with a special advantage in handling minority classes due to its categorical feature processing, gradient boosting, and class balancing [25]. As shown in Tables 4 and 5, CB performed the most stably and consistently and thus was the best model for this classification task because all the features of the dataset are categorical, and CB naturally deals with categorical features and class imbalance and effectively leverages feature interactions [21].

# Results and Discussions

This section reports the findings of the study in three parts: baseline classifier performance, target grouping strategy evaluation, and comparison. As indicated in Table 3, SVM recorded the best baseline accuracy (0.664) and F1 score (0.638), followed closely by LR and XGB. The relatively poor performance indicates the difficulty of dealing with multiple imbalanced classes in a small dataset, highlighting the necessity for better target grouping to improve generalization.

To address class imbalance issues, we experimented with four target grouping methods: target-4, target-3, target-2, and target-mnm. As indicated in Table 4, the target-mnm grouping performed optimally across all classifiers, and CB had the highest accuracy level of 0.879.

The results show that the decrease of target classes significantly improves classification performance by reducing complexity and enhancing generalization. Table 5 presents a comparison between the baseline and the proposed target-mnm partitioning using the two-stage approach outlined in Algorithm 1. All the classifiers perform better with the target-mnm strategy, with LR and CB performing at the best 0.801 accuracy.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TABLE 3**. Baseline performance of classifiers | | | | |
| **Classifier** | **Accuracy** | **Precision** | **Recall** | **F1 Score** |
| LR | 0.657 | 0.618 | 0.657 | 0.633 |
| SVM | 0.664 | 0.624 | 0.664 | 0.638 |
| XGB | **0.659** | 0.623 | 0.659 | 0.632 |
| CB | 0.654 | 0.618 | 0.654 | 0.631 |
| RF | 0.653 | 0.612 | 0.653 | 0.625 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TABLE 4.** Performance metrics for different target groupings | | | | | |
| **Classifier** | **Target Group** | **Accuracy** | **Precision** | **Recall** | **F1 Score** |
| Logistic Regression | Target-4 | 0.722 | 0.727 | 0.722 | 0.722 |
| Target-3 | 0.748 | 0.752 | 0.748 | 0.749 |
| Target-2 | 0.836 | 0.837 | 0.836 | 0.836 |
| Target-mnm | 0.873 | 0.863 | 0.873 | 0.863 |
| SVM | Target-4 | 0.728 | 0.738 | 0.728 | 0.731 |
| Target-3 | 0.750 | 0.764 | 0.750 | 0.752 |
| Target-2 | 0.840 | 0.840 | 0.840 | 0.840 |
| Target-mnm | 0.877 | 0.869 | 0.877 | 0.868 |
| XGB | Target-4 | 0.706 | 0.712 | 0.706 | 0.707 |
| Target-3 | 0.728 | 0.731 | 0.728 | 0.728 |
| Target-2 | 0.811 | 0.811 | 0.811 | 0.811 |
| Target-mnm | 0.849 | 0.835 | 0.849 | 0.837 |
| CB | Target-4 | 0.736 | 0.741 | 0.736 | 0.737 |
| Target-3 | 0.744 | 0.749 | 0.744 | 0.744 |
| Target-2 | 0.844 | 0.844 | 0.844 | 0.844 |
| Target-mnm | **0.879** | 0.871 | 0.879 | 0.869 |
| RF | Target-4 | 0.730 | 0.736 | 0.730 | 0.731 |
| Target-3 | 0.742 | 0.749 | 0.742 | 0.743 |
| Target-2 | 0.827 | 0.827 | 0.827 | 0.827 |
| Target-mnm | 0.873 | 0.864 | 0.873 | 0.860 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TABLE 5.** Comparative Analysis of Baseline and Target-mnm Groupings with the Baseline | | | | | |
| **Classifier** | **Method** | **Accuracy** | **Precision** | **Recall** | **F1 Score** |
| LR | Baseline | 0.657 | 0.618 | 0.657 | 0.633 |
| Target-mnm | 0.801 | 0.793 | 0.801 | 0.793 |
| SVM | Baseline | 0.664 | 0.624 | 0.664 | 0.638 |
| Target-mnm | 0.797 | 0.790 | 0.797 | 0.790 |
| XGB | Baseline | 0.659 | 0.623 | 0.659 | 0.632 |
| Target-mnm | 0.773 | 0.763 | 0.773 | 0.765 |
| CB | Baseline | 0.654 | 0.618 | 0.654 | 0.631 |
| Target-mnm | **0.801** | 0.795 | 0.801 | 0.794 |
| RF | Baseline | 0.653 | 0.612 | 0.653 | 0.625 |
| Target-mnm | 0.794 | 0.786 | 0.794 | 0.784 |

This analysis underscores the effectiveness of the target-mnm grouping strategy in enhancing model performance, particularly for datasets with limited size and numerous target categories. The results validate the proposed method as a robust solution to mitigate class imbalance and improve classification outcomes.

# Conclusion

This study successfully addresses data classification challenges in small, imbalanced datasets by introducing a novel target reduction clustering approach. Our evaluation across a set of classification algorithms, including LR, SVM, XGB, CB, and RF, consistently shows enhanced accuracy, precision, recall, and F1 scores using the suggested target-mnm clustering approach. The findings demonstrate that target reduction greatly alleviates class imbalance, thus offering an effective solution for MCC. This study has applied this process exclusively to a tabular dataset; future research will extend its applicability to resource-constrained environments, providing real-world solutions and actionable suggestions for industries like healthcare, fraud detection, and image classification. The study highlights the importance of novel approaches in classification performance and alludes to the possibility of broader relevance in various fields facing similar data issues.

# ACKNOWLEDGMENTS

This project is funded and supported by TM R&D Fund from Telekom Malaysia, Malaysia.

# References

1. A. A. Khan, O. Chaudhari, and R. Chandra, “A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation,” *Expert Systems with Applications* **244**, 122778 (2024).
2. M. Altalhan, A. Algarni, and M. Turki-Hadj Alouane, “Imbalanced data problem in machine learning: A review,” *IEEE Access* **13**, 13686–13699 (2025).
3. P. Bhardwaj, M. Gorkhe, R. Kudare, T. M. Joshi, V. M. Keerthi, S. Chavan, et al., “Leveraging machine learning for fraud detection in the financial sector,” in *Utilizing AI and Machine Learning in Financial Analysis* (IGI Global Scientific Publishing, 2025), pp. 21–48.
4. M. A. Lones, “How to avoid machine learning pitfalls: A guide for academic researchers,” *arXiv preprint* arXiv:2108.02497 (2021).
5. L. Dube and T. Verster, “Enhancing classification performance in imbalanced datasets: A comparative analysis of machine learning models,” *Data Science in Finance and Economics* **3**, 354–379 (2023).
6. T.-J. Law, C.-Y. Ting, H. Ng, H.-N. Goh, and A. Quek, “Ensemble-SMOTE: Mitigating class imbalance in graduate on time detection,” *Journal of Information and Web Engineering* **3**(2), 229–250 (2024).
7. V. Patel and H. Bhavsar, “Optimizing kernel transformations to handle binary class imbalanced dataset classification,” *Applied Artificial Intelligence* **38**(1), 2408933 (2024).
8. H. Pant and R. Srivastava, “Microclustering-based multi-class classification on imbalanced multi-relational datasets,” *International Journal of Information Technology and Web Engineering* **17**, 1–13 (2022).
9. J. Wang and N. Awang, “MKC-SMOTE: A novel synthetic oversampling method for multi-class imbalanced data classification,” *IEEE Access* (2024).
10. T. Zhou, X. Gao, X. Sun, and L. Han, “Split difference weighting: An enhanced decision tree approach for imbalanced classification,” *International Journal of Computers Communications and Control* **19** (2024).
11. B. Manjula and S. Layaq, “A hybrid approach for imbalanced data reduction using oversampling and undersampling,” *Journal of Xi’an University of Architecture and Technology*, 1–10 (2024).
12. T. Acharya, I. Khatri, A. Annamalai, and M. F. Chouikha, “Efficacy of machine learning-based classifiers for binary and multi-class network intrusion detection,” in *2021 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)* (IEEE, 2021), pp. 402–407.
13. X. Jiang, J. Wang, Q. Meng, M. Saada, and H. Cai, “An adaptive multi-class imbalanced classification framework based on ensemble methods and deep network,” *Neural Computing and Applications* **35**, 11141–11159 (2023).
14. A. Gonzalez-Ramirez, J. Lopez, D. Torres-Roman, and I. Yañez-Vargas, “Analysis of multi-class classification performance metrics for remote sensing imagery imbalanced datasets,” *Journal of Quantitative Statistical Analysis*, 8–22 (2021).
15. R. Haque, H. N. Goh, C. Y. Ting, A. Quek, and M. R. Hasan, “Imbalanced data classification using oversampling and automatic feature selection methods for undergraduate student career prediction,” in *Proceedings of the 2024 13th International Conference on Educational and Information Technology (ICEIT)*, 247–252 (2024).
16. Z. Jin, M. R. Anderson, M. Cafarella, and H. V. Jagadish, “Foofah: Transforming data by example,” in *Proceedings of the 2017 ACM International Conference on Management of Data* (ACM, 2017), pp. 683–698.
17. L. Yujian and L. Bo, “A normalized Levenshtein distance metric,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **29**, 1091–1095 (2007).
18. M. C. Data, M. Komorowski, D. C. Marshall, J. D. Salciccioli, and Y. Crutain, “Exploratory data analysis,” in *Secondary Analysis of Electronic Health Records* (Springer, 2016), pp. 185–203.
19. H. Wang and D. Hu, “Comparison of SVM and LS-SVM for regression,” in *2005 International Conference on Neural Networks and Brain*, Vol. 1 (IEEE, 2005), pp. 279–283.
20. A. Pathy, S. Meher, and P. Balasubramanian, “Predicting algal biochar yield using eXtreme Gradient Boosting (XGB) algorithm of machine learning methods,” *Algal Research* **50**, 102006 (2020).
21. J. T. Hancock and T. M. Khoshgoftaar, “CatBoost for big data: An interdisciplinary review,” *Journal of Big Data* **7**, 94 (2020).
22. S. J. Rigatti, “Random forest,” *Journal of Insurance Medicine* **47**, 31–39 (2017).
23. J. T. Townsend, “Theoretical analysis of an alphabetic confusion matrix,” *Perception and Psychophysics* **9**, 40–50 (1971).
24. A. P. Bradley, “The use of the area under the ROC curve in the evaluation of machine learning algorithms,” *Pattern Recognition* **30**, 1145–1159 (1997).
25. T.-J. Law, C.-Y. Ting, H. Ng, H.-N. Goh, and A. Quek, “Ensemble-SMOTE: Mitigating Class Imbalance in Graduate on Time Detection,” *Journal of Informatics and Web Engineering* **3**(2), 229–250 (2024).