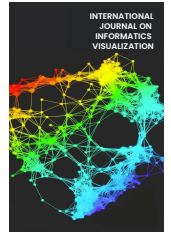




INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv



Avoiding Overfitting dan Overlapping in Handling Class Imbalanced Using Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection

Hartono ^{a,*}, Erianto Ongko ^b

^a Department of Computer Science, Universitas Potensi Utama, Medan, 20241, Indonesia

^b Department of Informatics, Akademi Teknologi Industri Immanuel, 20114, Medan, Indonesia

Corresponding author: *hartonoibbi@gmail.com

Abstract— The dataset tends to have the possibility to experience imbalance as indicated by the presence of a class with a much larger number (majority) compared to other classes(minority). This condition results in the possibility of failing to obtain a minority class even though the accuracy obtained is high. In handling class imbalance, the problems of diversity and classifier performance must be considered. Hence, the Hybrid Approach method that combines the sampling method and classifier ensembles presents satisfactory results. The Hybrid Approach generally uses the oversampling method, which is prone to overfitting problems. The overfitting condition is indicated by high accuracy in the training data, but the testing data can show differences in accuracy. Therefore, in this study, Smoothed Bootstrap Resampling is the oversampling method used in the Hybrid Approach, which can prevent overfitting. However, it is not only the class imbalance that contributes to the decline in classifier performance. There are also overlapping issues that need to be considered. The approach that can be used to overcome overlapping is Feature Selection. Feature selection can reduce overlap by minimizing the overlap degree. This research combined the application of Feature Selection with Hybrid Approach Redefinition, which modifies the use of Smoothed Bootstrap Resampling in handling class imbalance in medical datasets. The preprocessing stage in the proposed method was carried out using Smoothed Bootstrap Resampling and Feature Selection. The Feature Selection method used is Feature Assessment by Sliding Thresholds (FAST). While the processing is done using Random Under Sampling and SMOTE. The overlapping measurement parameters use Augmented R-Value, and Classifier Performance uses the Balanced Error Rate, Precision, Recall, and F-Value parameters. The Balanced Error Rate states the combined error of the majority and minority classes in the 10-Fold Validation test, allowing each subset to become training data. The results showed that the proposed method provides better performance when compared to the comparison method.

Keywords— Class imbalance; overfitting; hybrid approach redefinition; overlapping; feature selection.

Manuscript received 18 Jan. 2022; revised 19 Mar. 2022; accepted 15 Apr. 2022. Date of publication 30 Jun. 2022.
International Journal on Informatics Visualization is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.



I. INTRODUCTION

The problem of dataset imbalance is often experienced in classification algorithms caused by the fact that datasets in the real world are rarely perfectly balanced [1]. The classification algorithm provides optimum results in a situation where the sample distribution is balanced in each class and requires special handling of the sample imbalance problem to achieve optimum performance [2]. Classes with fewer instances (minority class) are often ignored in the classification algorithm, or there is misclassification of the minority class into another class even though the minority class is a class with a high value because it is the center of observation [3]. Class imbalance is unavoidable; for example, medical

datasets are obtained from patient medical data, where the number of patients suffering from the disease is much less than the number of patients without the disease[4].

There are 2 (two) algorithms for dealing with class imbalance problems: data-level techniques and algorithm-level methods[5]. Data-level techniques are used in the form of sampling to reduce imbalance by increasing the number of samples in the minority class (oversampling) or reducing the number of samples in the majority class (undersampling)[6]. Criticism of Data-Level is especially related to overfitting problems in the application of oversampling or omitting important data from a class in undersampling [7]. The Algorithm-level works by generating many classifiers through a modification process to the classification algorithm.

Algorithm-level accuracy tends to decrease in high-dimensional datasets [8]. Many researchers have proposed a Hybrid Approach that combines the advantages of data-level and algorithm-level in handling class imbalance [9], [10]. The Hybrid Approach has the advantage of overcoming a weakness at both the data-level and algorithm level to complement each other to provide better performance[11]. Akbani et al. [12] shows that combining data-level and algorithm-level with SVM and SMOTE gives better results than using only data levels such as RUS and SMOTE or only using algorithm levels such as SVM.

The Hybrid Approach tends to use oversampling compared to undersampling because, based on research from many researchers, it is found that oversampling gives better results than undersampling on severely imbalanced datasets, although the differences are not significant [13], [14]. However, overfitting problems in oversampling should be emphasized because overfitting can cause good accuracy in training data, but this is not the case with testing data [15]. Therefore, a number of oversampling methods have been proposed that offer the ability to handle overfitting, and one of them is Smoothed Bootstrap Resampling. The Smoothed Bootstrap Resampling method has shown good performance in terms of performance on training data and testing data [16].

It is not only the class imbalance that needs attention to obtain good classification results. The problem of overlapping often goes unnoticed, even though this overlap can also affect the prediction results [17]. One of the efforts to handle overlapping is to minimize overlapping degrees by using Feature Selection [18]. One method that combines feature selection with oversampling is Wrapper Approach-SMOTE [19]. The use of Feature Selection and Oversampling, in addition to being effective in dealing with overlapping is also proven to provide accurate results and also fast detection of class imbalance problems [20]. The advantage of feature selection with the wrapper approach is that it can find the appropriate region classifier for the sampling process so that it could be more effective [21]. Research conducted by Ghazikhani et al. [22] shows that the Wrapper Approach is the most suitable feature selection method to be combined with SMOTE in dealing with overlapping and class imbalance.

Based on the consideration of the importance of efforts to deal with overfitting and overlapping in handling class imbalance, this research combined the application of Feature Selection with Hybrid Approach Redefinition, which modifies the use of Smoothed Bootstrap Resampling in handling class imbalance. The results of this study were compared with the Wrapper Approach-SMOTE.

II. MATERIALS AND METHOD

A. Hybrid Approach

The pseudocode of the Hybrid Approach is as follows[23].

```

Input:  $D_T = \{x_1, x_2, \dots, x_n\}$  // Training Dataset
 $N$  = Number of Classifier
Output: Classification Prediction  $P$ 
Method:
Step 1 Preprocessing using Preprocessing Method
Step 2 For  $i = 1$  to  $N$  do
    i. Apply Machine Learning Classification Algorithm
    on The Attributes of  $D_T$ 
    ii. Obtain Classification Prediction  $P_i$  from machine

```

```

learning classification algorithm
End For
Step 3 For  $i = 1$  to  $n$ 
    Apply processing using bagging, boosting or sampling
End For

```

Based on the pseudocode above, it can be seen that in the Hybrid Approach, data-level and algorithm-level are used, which are applied to the preprocessing and processing stages. The preprocessing stage is carried out to ensure that the dataset or samples are ready to undergo the processing stage.

B. Smoothed Bootstrap Resampling (SBR)

The pseudocode of the SBR is as follows[16].

```

Input: Data ( $N$ )
Output: The Synthetic Minority Class (Samples)
 $X$  = Transform  $N$  in sequence vertically
 $Y$  = Transform  $N$  in sequence Horizontally
 $X_{min}$  = add  $X$  and  $Y$ 
 $CalcStd$  = ComputeStdDev( $X_{min}$ )
 $Value_{data}$  = ReturnValue( $N$ )
 $ValueX_{min}$  = ReturnValue( $X_{min}$ )
 $H_{Matrix}$  = ComputeMatrix( $CalcStd, Value_{data}, ValueX_{min}$  //
Using Equation 1

```

$$h_q^{(j)} = \left(\frac{4}{(d+2n)} \right)^{\frac{1}{d+4}} \hat{\sigma}_q^{(j)} \quad (q = 1, \dots, d = 1, 2) \quad (1)$$

```

Samples = {}
For each item in  $N$ 
    Rand = Random Number
     $ValueX_{min}$  = ReturnValue( $X_{min}$ )
     $H_{Index}$  = ComputeRandomIndex(Rand,  $ValueX_{min}$ )
     $Value_{Gauss}$  = ComputeGaussianDistrib( $X_{min} | H_{Index}, H_{Matrix}$ ) //
Using Equation 2

```

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2)$$

```

Add  $Value_{Gauss}$  to Samples
End For
Return Samples

```

Based on the pseudocode above, several parameters need to be considered, namely: $\hat{\sigma}_q^{(j)}$ is a sample estimate of the standard deviation of the q -th dimension belong to the class y_j . $h_q^{(j)}$ is matrix smoothing, μ is the mean, and σ is the value of the standard deviation, and σ^2 is the variance.

C. Feature Selection

The Feature Selection method used in this study is Feature Assessment by Sliding Thresholds (FAST)[24]. The pseudocode of FAST is as follows.

```

 $K$ : number of bins
 $N$ : number of samples in dataset
 $M$ : number of Features in dataset
Split = 0 to  $N$  with a step size  $N/K$ 
For  $i = 1$  to  $M$ 
     $X$  is a vector of samples values for feature  $i$ 
    Sort  $X$ 
    For  $j = 1$  to  $K$ 
        Bottom = round(Split( $j$ )) + 1
        Top = round(Split( $j+1$ ))
        MU = mean( $X$ (bottom to top))
        Classify  $X$  using MU as threshold
        tpr( $i, j$ ) = tp/#positive
        fpr( $i, j$ ) = fp/#negative
    Calculate Area Under ROC by tpr, fpr

```

In the pseudocode above, it can be seen that Feature Selection with FAST starts with determining the number of attributes or features from the dataset. The loop was executed based on the number of existing features. Each stage used each feature to determine the value of tpr, fpr, and Area Under ROC.

D. Augmented R-Value

Augmented R-Value states how much overlapping occurs. The greater the Augmented R-Value, the greater the overlapping[25].

$$R_{aug}(D[V]) = \frac{\sum_{i=0}^{k-1} |C_{k-1-i}| R(C_i)}{\sum_{i=0}^{k-1} |C_i|} \quad (3)$$

Where C_0, C_1, \dots, C_{k-1} are k class labels with $|C_0| \geq |C_1| \geq \dots \geq |C_{k-1}|$ and $D[V]$: Dataset D containing predictors in set V . Larger R_{Aug} is higher overlap degree of a dataset.

E. Classifier Performance

Classifier Performance was measured using Accuracy, Precision, Recall, MicroF1, and MacroF1. This classifier performance measurement is carried out based on the confusion matrix, which can be seen in Table 1[26][27][5].

TABLE I
CONFUSION MATRIX

	Predictive Positive Class	Predictive Negative Class
Actual Positive Class	True Positive (TP)	False Negative (FN)
Actual Negative Class	False Positive (FP)	True Negative (TN)

The Balanced Error Rate, Precision, Recall, MicroF1, and MacroF1 calculations can be seen in the following equation[27][5].

$$Balanced\ Error\ Rate = \frac{1}{2} \left(\frac{FP}{FP+TP} + \frac{FN}{FN+TN} \right) \quad (4)$$

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

$$F-Value = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (7)$$

In Equation 4, it can be seen that the balanced Error Rate states the average error that occurs in both the minority class and majority class, which becomes more accurate if it is used to calculate the accuracy of the imbalanced dataset. Equation 5 states that precision is the number of minority classes (positive samples) that are correctly classified from the overall classification results, which declare an instance as a minority class. Meanwhile, Equation 6 states that recall is the number of minority classes (positive samples) that are correctly classified from the entire minority class, including those incorrectly classified as majority class. Equation 7 F-Value states the accuracy associated with the balance of precision and recall.

F. Proposed Method / Algorithm

The research stages can be seen in Figure 1. Figure 1 shows the stages of research that passed in this research. The research process can be briefly described as consisting of 2 (two) major stages: preprocessing and processing. The preprocessing stage begins with the resampling process using Smoothed Bootstrap Resampling. The Smoothed Bootstrap Resampling process is a resampling process that calculates the Gaussian Distribution value of each sample. This process is important to prevent overfitting in the oversampling process.

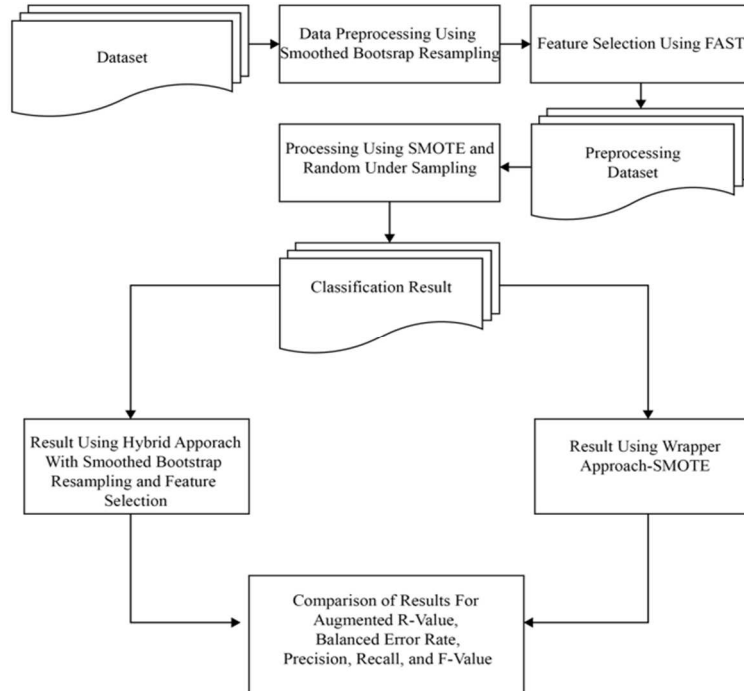


Fig. 1 Research Stage

After that, the stage switches to the Feature Selection process using FAST. The feature selection stage is intended to reduce the degree associated with overlapping. The results of the Smoothed Bootstrap Resampling and FAST processes are preprocessed datasets. The preprocessed dataset then enter the processing stage using Different Contribution Sampling.

1) *Preprocessing Using Smoothed Bootstrap Resampling and FAST*: The pseudocode of the preprocessing stage is as follows.

```

Input: Data
Output: The Synthetic Minority Class (Samples)
1: Compute Matrix Smoothing Using Equation 1
2: For Each Item in Data
3:   Compute GaussianDistribution Using Equation 2
4:   AddValueGauss to Samples
5: End For
6: Return Samples
7: Calculate Number of Samples N
8: Calculate Number of Features M
9: Define Number of Bins K
10: Split = 0 to N with Step N/K
11: For i = 1 to K
12:   Calculate TPR, FPR, and Area Under ROC
13: End For

```

Based on the pseudocode, it can be seen that the very first step is to form a smoothing matrix based on the existing dataset. The smoothing matrix is determined based on the standard deviation value, which played a role in determining the Gaussian distribution value. The purpose of determining the value of the Gaussian distribution is to anticipate the occurrence of overfitting in the oversampling process. Then after that, the process was continued with determining the number of features in the dataset, and an iterative process was carried out as many as the number of features or attributes to determine the TPF, FPR, and Area Under ROC values, which this process is a feature selection process which is the last stage of the preprocessing. This stage gives results in the form of a preprocessed dataset which was continued to the processing stage.

2) *Processing Using SMOTE and RandomUndersampling*: The pseudocode of the processing stage is as follows.

```

Input: Total Size totalSize, Number of Majority SN, Number of Minority SP
1: totalSize ← |S|
2: SN = {(xi, yi) ∈ S | yi = -1}
3: SP = {(xi, yi) ∈ S | yi = +1}
4: majoritySize ← |SN|
5: minoritySize ← |SP|
6: Execute SN to obtain the clusering list named AP
7: Allocate each record of SN to Size (AP)
8: For (i = 1; i ≤ size (AP); i++)
9:   For (j = 1; j ≤ size (AP[i]); j++)
10:    Value = AP[i][j]
11:    S[Value, ncol(S)] = i
12:   End For
13: End For
14: k = Number of Nearest Neighbors
15: numattrs = number of attributes
16: Sample = Minority Class Sample
17: DMajorityReduced = Array of Majority
18: DMinority = Array of Minority
19: For (i = 1; i ≤ majoritySize; i++)
20:   Compute k nearest neighbors
21:   Populate (N, i, nnarray)
22: End For
23: While N ≠ 0 do

```

```

24:   for (i = 1; i ≤ numattrs; i++)
25:     dif[i] = sample[nnarray[i][attr] - sample[i][attr]
26:   End For
27: End While
28: For (i = 1; i ≤ majoritySize; i++)
29:   RandomUnderSampling sample[i]
30:   DMajorityReduced[i] = sample[i]
31: End For
32: For (i = 1; i ≤ minoritySize ; i++)
33:   SMOTE sample[i]
34:   DMinority[i] = Dminority + sample[i]
35: End For
36: Combine DMajorityReduced with DMinority become Result

```

In the processing stage, it can be seen that different handling is given to the majority and minority classes. Especially for the majority class, the undersampling process is carried out using Random Under Sampling, while for the minority class, the oversampling process is carried out using SMOTE.

III. RESULTS AND DISCUSSION

A. Dataset Description

KEEL Repository provides access to the dataset used in this study[28]. The dataset used can be seen in Table II.

TABLE II
DATASET DESCRIPTION

Dataset	Number of Examples	Number Of Attributes	Class (%Min;%Maj)	IR
Ecoli1	336	7	22.92;77.08	3.36
Yeast3	1484	8	10.98;89.02	8.11
Page-Blocks	5472	10	10.23;89.77	8.77
Abalone9vs18	731	8	5.65;94.25	16.68
Yeast5	1484	8	2.96;97.04	32.78
Yeast6	1484	8	2.49;97.51	39.15

In Table II, it can be seen that the selected dataset varies in terms of the number of samples, the number of attributes, and the imbalance ratio. It can be said that the results of training and testing using the dataset can accurately describe the results of handling class imbalances.

B. Experimental Setup

Performance testing of the proposed method is carried out on the datasets that have been stated in the previous section. Evaluation is carried out using traditional performance metrics consisting of: Augmented R-Value, Balanced Error Rate, Precision, Recall, and F-Value. The evaluation was carried out using a stratified k-fold (k=10). In the stratified k-fold, it can be said that the training data is divided into 10 subsets of the same size, while still considering the distribution of each class in order to maintain the imbalance ratio. During the testing process, one of the subsets still acts as testing data, and the remaining k-1 subsets act as training data. The process was repeated for k iterations, where each subset of k was used once as testing data. The results obtained are a combination of the results in each iteration.

C. Testing Result

The first test was conducted to obtain Augmented R-Value and Balanced Error Rate (BER). The test results can be seen in Table III.

TABLE III
TESTING FOR AUGMENTED R-VALUE AND BALANCED ERROR RATE

Dataset	Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection		Wrapper Approach-SMOTE	
	Augmented R-Value	BER	Augmented R-Value	BER
Ecoli1	0.291	0.087	0.293	0.091
Yeast3	0.297	0.096	0.301	0.101
Page-Blocks	0.301	0.108	0.321	0.107
Abalone9vs18	0.325	0.118	0.331	0.124
Yeast5	0.337	0.121	0.341	0.127
Yeast6	0.339	0.122	0.344	0.130

Based on the results obtained, it can be seen that both methods show better results at a smaller imbalance ratio. Augmented R-Value and BER values obtained are better at lower imbalance ratios. The results also show that the Augmented R-Value results obtained by the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection are better than the Wrapper Approach-SMOTE. Especially for the BER method of Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection, in addition to the imbalance ratio, the number of instances also has an effect where in a dataset with a not too large number of instances, the results obtained tend to be better. In the Page-Blocks Dataset, where the number of instances is larger, the results obtained by the Wrapper Approach-SMOTE are better than the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection.

So it can be said that for overlapping which Augmented R-Value expresses, the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection method is better than the Wrapper Approach-SMOTE. As for overfitting expressed by BER, the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection method is better than the Wrapper Approach-SMOTE method in almost all datasets except Page-Blocks.

The second test was conducted to obtain Precision, Recall, and F-Value. The test results can be seen in Table IV.

TABLE IV
TESTING FOR PRECISION, RECALL, AND F-VALUE

Dataset	Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection			Wrapper Approach-SMOTE		
	Precision	Recall	F-Value	Precision	Recall	F-Value
Ecoli1	0.88	0.92	0.91	0.81	0.86	0.83
Yeast3	0.85	0.89	0.86	0.79	0.88	0.83
Page-Blocks	0.84	0.87	0.85	0.77	0.78	0.79
Abalone9vs18	0.83	0.82	0.84	0.78	0.71	0.72
Yeast5	0.84	0.81	0.81	0.82	0.79	0.71
Yeast6	0.85	0.79	0.78	0.81	0.75	0.71

Based on Table IV, the performance of the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection is generally better than the Wrapper Approach-SMOTE. Like in the previous test, the results obtained are also better at a smaller imbalance ratio.

D. Statistical Tests

The Wilcoxon Signed-Rank Test was conducted to test whether there were significant differences between each

method in each of the measurement parameters that had been carried out[29]. It is said that there is a significant difference if the P-Value < 0.05 . The statistical test results can be seen in Table V.

TABLE V
STATISTICAL TESTS USING WILCOXON SIGNED-RANK TEST

Performance Measurement	P-Value	Significant Difference
Augmented R-Value	0.0355223	There is a significant difference between the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection when compared to the Wrapper Approach-SMOTE
Balanced Error Rate	0.0584753	There is no significant difference between the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection when compared to the Wrapper Approach-SMOTE
Precision	0.0355223	There is a significant difference between the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection when compared to the Wrapper Approach-SMOTE
Recall	0.0312500	There is a significant difference between the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection when compared to the Wrapper Approach-SMOTE
F-Value	0.0312500	There is a significant difference between the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection when compared to the Wrapper Approach-SMOTE

E. Discussion

Based on the experimental results and statistical tests, it can be seen that Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection gives better and more significant results on Augmented R-Value, which indicates that the overlapping treatment results obtained are better than Wrapper Approach-SMOTE. However, this does not mean that the results given Wrapper Approach-SMOTE are not good; both methods provide good overlapping handling results. This is indicated by the two methods providing a very small Augmented R-Value value, meaning that the overlap that occurs is very small. There is a tendency that overlapping problems need more attention in datasets with large imbalance ratios. As for the Balanced Error Rate (BER), which states the error from both the majority and minority classes shows a very low value, with 10-Fold Validation where each subset becomes testing data, the results obtained are good, which shows that the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection and the Wrapper Approach-SMOTE have provided good overfitting results. On BER, there can be no significant difference between the two methods.

On the results of the precision, recall, and F1-Value tests, the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection gives better and more significant results than the Wrapper Approach-SMOTE. Both methods have basically resulted in good handling of class imbalance.

IV. CONCLUSION

Based on the results in Tables III, IV, and V, it is found that the results obtained with the Hybrid Approach with Smoothed

Bootstrap Resampling and Feature Selection in handling overfitting and overlapping on imbalanced datasets are good. The main objective of this study is to treat class imbalance by not forgetting the handling of overfitting and overlapping. For handling class imbalance, the results obtained are good, as indicated by good Precision, Recall, and F-1 Value values. When compared with the Wrapper Approach-SMOTE method as a comparison, there are significant differences.

As for handling Overlapping, the Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection method gives very good and significant results to the Wrapper Approach-SMOTE method. As for BER, the results obtained apart from depending on the imbalance ratio also depend on the number of instances of each dataset.

ACKNOWLEDGMENT

The authors thank the Directorate of Research and Development, under the Ministry of Education, Culture, Research, and Technology, Indonesia, for supporting this research.

REFERENCES

- [1] R. Ahsan, F. Ebrahimi, and M. Ebrahimi, "Classification of imbalanced protein sequences with deep-learning approaches; application on influenza A imbalanced virus classes," *Informatics in Medicine Unlocked*, p. 100860, Jan. 2022, doi: 10.1016/j.imu.2022.100860.
- [2] L. Dou, F. Yang, L. Xu, and Q. Zou, "A comprehensive review of the imbalance classification of protein post-translational modifications," *Briefings in Bioinformatics*, vol. 22, no. 5, p. bbab089, Sep. 2021, doi: 10.1093/bib/bbab089.
- [3] D. I. Tsilimigras *et al.*, "A Machine-Based Approach to Preoperatively Identify Patients with the Most and Least Benefit Associated with Resection for Intrahepatic Cholangiocarcinoma: An International Multi-institutional Analysis of 1146 Patients," *Ann Surg Oncol*, vol. 27, no. 4, pp. 1110–1119, Apr. 2020, doi: 10.1245/s10434-019-08067-3.
- [4] Y.-C. Wang and C.-H. Cheng, "A multiple combined method for rebalancing medical data with class imbalances," *Computers in Biology and Medicine*, vol. 134, p. 104527, Jul. 2021, doi: 10.1016/j.combiomed.2021.104527.
- [5] K. De Angeli *et al.*, "Class imbalance in out-of-distribution datasets: Improving the robustness of the TextCNN for the classification of rare cancer types," *Journal of Biomedical Informatics*, vol. 125, p. 103957, Jan. 2022, doi: 10.1016/j.jbi.2021.103957.
- [6] W.-C. Lin, C.-F. Tsai, Y.-H. Hu, and J.-S. Jhang, "Clustering-based undersampling in class-imbalanced data," *Information Sciences*, vol. 409–410, pp. 17–26, Oct. 2017, doi: 10.1016/j.ins.2017.05.008.
- [7] U. R. Salunkhe and S. N. Mali, "Classifier Ensemble Design for Imbalanced Data Classification: A Hybrid Approach," *Procedia Computer Science*, vol. 85, pp. 725–732, Jan. 2016, doi: 10.1016/j.procs.2016.05.259.
- [8] I. D. Mienye and Y. Sun, "Performance analysis of cost-sensitive learning methods with application to imbalanced medical data," *Informatics in Medicine Unlocked*, vol. 25, p. 100690, Jan. 2021, doi: 10.1016/j.imu.2021.100690.
- [9] N. Liu, X. Li, E. Qi, M. Xu, L. Li, and B. Gao, "A Novel Ensemble Learning Paradigm for Medical Diagnosis With Imbalanced Data," *IEEE Access*, vol. 8, pp. 171263–171280, 2020, doi: 10.1109/ACCESS.2020.3014362.
- [10] S. Balasubramanian, R. Kashyap, S. T. CVN, and M. Anuradha, "Hybrid Prediction Model For Type-2 Diabetes With Class Imbalance," in *2020 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT)*, Dec. 2020, pp. 1–6, doi: 10.1109/ICMLANT50963.2020.9355975.
- [11] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder, and N. Seliya, "A survey on addressing high-class imbalance in big data," *Journal of Big Data*, vol. 5, no. 1, p. 42, Nov. 2018, doi: 10.1186/s40537-018-0151-6.
- [12] R. Akbani, S. Kwek, and N. Japkowicz, "Applying Support Vector Machines to Imbalanced Datasets," in *Machine Learning: ECML 2004*, Berlin, Heidelberg, 2004, pp. 39–50, doi: 10.1007/978-3-540-30115-8_7.
- [13] M. Koziarski, "Radial-Based Undersampling for imbalanced data classification," *Pattern Recognition*, vol. 102, p. 107262, Jun. 2020, doi: 10.1016/j.patcog.2020.107262.
- [14] N. Rodríguez, D. López, A. Fernández, S. García, and F. Herrera, "SOUL: Scala Oversampling and Undersampling Library for imbalance classification," *SoftwareX*, vol. 15, p. 100767, Jul. 2021, doi: 10.1016/j.softx.2021.100767.
- [15] S. Y. Ho, L. Wong, and W. W. B. Goh, "Avoid Oversimplifications in Machine Learning: Going beyond the Class-Prediction Accuracy," *Patterns*, vol. 1, no. 2, p. 100025, May 2020, doi: 10.1016/j.patter.2020.100025.
- [16] P. Wibowo and C. Fatichah, "Pruning-based oversampling technique with smoothed bootstrap resampling for imbalanced clinical dataset of Covid-19," *Journal of King Saud University - Computer and Information Sciences*, Sep. 2021, doi: 10.1016/j.jksuci.2021.09.021.
- [17] P. Vuttipittayamongkol, E. Elyan, and A. Petrovski, "On the class overlap problem in imbalanced data classification," *Knowledge-Based Systems*, vol. 212, p. 106631, Jan. 2021, doi: 10.1016/j.knsys.2020.106631.
- [18] A. Wahid *et al.*, "Feature selection and classification for gene expression data using novel correlation based overlapping score method via Chou's 5-steps rule," *Chemometrics and Intelligent Laboratory Systems*, vol. 199, p. 103958, Apr. 2020, doi: 10.1016/j.chemolab.2020.103958.
- [19] S. Sreejith, H. Khanna Nehemiah, and A. Kannan, "Clinical data classification using an enhanced SMOTE and chaotic evolutionary feature selection," *Computers in Biology and Medicine*, vol. 126, p. 103991, Nov. 2020, doi: 10.1016/j.combiomed.2020.103991.
- [20] S. Huda, J. Yearwood, H. F. Jelinek, M. M. Hassan, G. Fortino, and M. Buckland, "A Hybrid Feature Selection With Ensemble Classification for Imbalanced Healthcare Data: A Case Study for Brain Tumor Diagnosis," *IEEE Access*, vol. 4, pp. 9145–9154, 2016, doi: 10.1109/ACCESS.2016.2647238.
- [21] T. Thaher, M. Mafarja, B. Abdalhaq, and H. Chantar, "Wrapper-based Feature Selection for Imbalanced Data using Binary Queuing Search Algorithm," Oct. 2019, doi: 10.1109/ICTCS.2019.8923039.
- [22] A. Ghazikhani, H. S. Yazdi, and R. Monsefi, "Class imbalance handling using wrapper-based random oversampling," in *20th Iranian Conference on Electrical Engineering (ICEE2012)*, May 2012, pp. 611–616, doi: 10.1109/IranianCEE.2012.6292428.
- [23] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463–484, Jul. 2012, doi: 10.1109/TSMCC.2011.2161285.
- [24] X. Hou, T. Zhang, L. Ji, and Y. Wu, "Combating highly imbalanced steganalysis with small training samples using feature selection," *J. Vis. Commun. Image Represent.*, vol. 49, no. C, pp. 243–256, Nov. 2017, doi: 10.1016/j.jvcir.2017.09.016.
- [25] S. Oh, "A new dataset evaluation method based on category overlap," *Comput. Biol. Med.*, vol. 41, no. 2, pp. 115–122, Feb. 2011, doi: 10.1016/j.combiomed.2010.12.006.
- [26] X. Chen and M. Wasikowski, "FAST: a roc-based feature selection metric for small samples and imbalanced data classification problems," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, Aug. 2008, pp. 124–132, doi: 10.1145/1401890.1401910.
- [27] A. Luque, A. Carrasco, A. Martín, and A. de las Heras, "The impact of class imbalance in classification performance metrics based on the binary confusion matrix," *Pattern Recognition*, vol. 91, pp. 216–231, Jul. 2019, doi: 10.1016/j.patcog.2019.02.023.
- [28] J. Alcalá-Fdez *et al.*, "KEEL: a software tool to assess evolutionary algorithms for data mining problems," *Soft Comput.*, vol. 13, no. 3, pp. 307–318, Feb. 2009, doi: 10.1007/s00500-008-0323-y.
- [29] F. Wilcoxon, "Individual Comparisons by Ranking Methods on JSTOR," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.