Published *online* on the journal's web page : http://innovatics.unsil.ac.id

Innovation in Research of Informatics (INNOVATICS)

| ISSN (Online) 2656-8993 |

Air Quality Classification Using Extreme Gradient Boosting (XGBOOST) Algorithm

Albi Mulyadi Sapari [1], Asep Id Hadiana[2], Fajri Rakhmat Umbara[3]

[1,2,3]*Informatics Study Program,Universitas Jendral Achmad Yani, Jl. Terusan Jend. Sudirman, Cibeber, Kota Cimahi, Jawa Barat 40531*
[1]*albi.mulyadi@student.unjani.ac.id,* [2]*asep.hadiana@lecture.unjani.ac.id,* [3]*fajri.rakhmat@lecturer.unjani.ac.id*

ARTICLE INFORMATION

Article History:
Received: August 24, 2023
Last Revision: October 14, 2023
Published *Online*: October 16, 2023

KEYWORDS

Air Pollution,
Air Quality,
Classification,
Imbalance class,
Extreme Gradient Boosting

CORRESPONDENCE

Phone: +6281387354164

E-mail: albi.mulyadi@student.unjani.ac.id

ABSTRACT

Air pollution is a serious issue caused by vehicle exhaust, industrial factories, and piles of garbage. The impact is detrimental to human health and the environment. To quickly and accurately monitor classification, techniques are used. One efficient and accurate classification algorithm is XGBoost, a development of the Gradient Decision Tree (GDBT) with several advantages, such as high scalability and prevention of overfitting. The parameters used in the classification include Particulate Matter 10 (PM_{10}), Particulate Matter 2.5 ($PM_{2.5}$), Sulfur Dioxide (SO_2), Carbon Monoxide (CO), Ozone(O_3) and Nitrogen Dioxide(NO_2). This study aims to classify air quality into three labels or categories: good, moderate, and unhealthy. In the dataset used to experience an imbalance class, to overcome the imbalance class, techniques will be carried out, namely SMOTE, Random UnderSampling, and Random OverSampling, by producing an accuracy of up to 98,61% with the SMOTE technique for class imbalance. Testing the level of accuracy is done by using the Confusion Matrix.

1. INTRODUCTION

Air pollution is a problem that hurts the lives of living things [1]. Contaminated air causes various impacts and diseases. So, that affects humans in their daily activities because humans need a good mood. Air pollution can be caused because human activities combine air with substances, energy, or other components [2]. In addition, air pollution can be caused by motor vehicle smoke pollution and the construction of industrial factories [3]. Contaminated air pollution caused by clean air is compounded with nitrogen dioxide, sulfur dioxide, carbon monoxide, particulate matter, and ozone. When air mixes with the above substances and has high levels, it can cause respiratory problems and even death. The government has handled air pollution regulated by decree no KEP-107/KABAPEDAL/11/1997, which compiled regulations related to guidelines for calculating reporting and providing information on the Air Pollution Standard Index (ISPU)[4][5]. The Air Pollution Standard Index (ISPU) defines five air pollution parameters used for calculation, namely Carbon Monoxide(CO), Sulfur Dioxide (SO_2), Nitrogen Dioxide(NO_2), Ozone (O_3), and Particulate Matter(PM)[6].

To overcome this air pollution and assist the government in making policies to control air pollution, a system is needed to carry out air quality forecasts to monitor air quality [7]. The results of this system have the potential to support government efforts in formulating policies to control air pollution to achieve the desired air quality standards. The government generally provides information regarding air quality classification through data released by the BMKG. However, continuing research in air quality classification is essential to comprehensively understand the process. This step will provide clarity on the classification process, benefiting the public as well as researchers. In addition, the data mining approach in this study can be an essential benchmark for the government in classifying air quality effectively[4].

Research on air quality with various data mining algorithms has been carried out, namely air quality classification with the *K-Nearest Neighbor* (KNN) algorithm with accuracy results of 80%, precision of 82.3%, and recall of 93.3% [4]. The following research is

the air quality classification with the Naïve Bayes algorithm with an accuracy of 88%, 96% recall, and f1-score of 90% [8].

Classification has various methods, including XGBoost (Extreme Gradient Boosting). XGBoost is a framework developed from the Gradient Decision Tree (GDBT), a highly efficient and precise machine-learning algorithm. This GDPT can perform multiple machine learning processes, such as multi-category classification, click prediction, and sort learning [9]. Another advantage of XGBoost is its regularization, parallelization, flexibility, and good classification results, so the advantages of the XGBoost algorithm produce good performance [10]. So, using the XGBoost algorithm for air quality classification can produce higher accuracy results than other classification algorithms based on a comparison of algorithms by [11][12].

Because XGBoost can handle complex and diverse data and optimize models quickly and efficiently. The main advantages of using XGBoost for air quality classification are high accuracy for taking large amounts of data by using gradient enhancement techniques that build decision tree ensembles to make predictions, handle many features, and process complex relationships between air quality data, which can help improve accuracy. And to help improve higher accuracy, this research added a parameter, namely (PM2,5). This parameter is an additional parameter that influences air quality because (PM2,5) has an air particle size of 2.5 micrometers. These airborne particles are hazardous because of the pollutant substances produced from vehicle fumes, which can harm health. [13].

Then, the dataset used has an imbalance class or class imbalance. To overcome this, this study will compare three imbalance class methods. The methods used are SMOTE (Synthetic Minority Over-sampling Technique), Random Over Sampling and Random Under Sampling [14]. Based on the explanation above, the research conducted will determine the Extreme Gradient Boosting algorithm for the classification of determining air quality with parameters (PM_{10}), $(PM_{2.5})$, (SO_2), (CO), (O_3) and (NO_2) for the case of DKI Jakarta. The classification is based on three categories: good, moderate, and unhealthy.

2. RELATED WORK

In research [4] regarding air quality classification for the city of Palembang using the K-Nearest Neighbor (K-NN) method. The processed data consists of (PM_{10}), (SO_2), (CO), (O_3) and (NO_2). and air quality status. Wthe index categories are good, moderate, unhealthy, very unhealthy and dangerous. In this study, of the 20 data that had been trained and tested, only four data were inaccurate because the source data had unbalanced classes, with an accuracy of 80%. Research [15] Classification of air pollution levels with the method used, namely Artificial Neural Network. Data was obtained using IoT sensors. Then, the category will produce three pollution levels: Good, Moderate, and Unhealthy. The built model collects data, initial data processing, modeling, and model evaluation metrics. In addition to looking at the results of model accuracy, the model used is seen from the sensitivity and specificity of the bag. From this research, the experiments' results obtained sensitivity and specificity values above 90%, and the resulting accuracy by setting the network requirements was 96.61%. This accuracy can still be improved using other parameters affecting air quality. Research [16] Air quality classification uses Fuzzy Logic with two substances used to consider air quality: carbon monoxide (CO) and nitrogen dioxide (NO2). Then, the classification results are six categories: very low, low, moderate, high, very high, and extremely high. Only using two types of pollutant substances as parameters used for air quality classification is still not practical because many other types of pollutant substances impact the air. Research [17] classifies air quality using the Random Forest method with air pollutant parameters used, namely: (PM_{10}), (SO_2), (CO), (O_3) and (NO_2). The datasets used are Dataset1 (Beijing), Dataset2 and Dataset3 (Fangchenggang) and Dataset4 (Beijing and Fangchenggang). This study found that the distribution of air data was unbalanced, causing the random forest classification results to decrease. Research [18] Prediction of air quality using Extreme Gradient Boosting (XGBoost) combined with the SMOTE method based on the Air Pollution Standard Index (ISPU), the dataset used is air quality for the last five years based on the Jakarta Environment Agency using a publication frequency of 1-month measurement with the parameter used being PM10, SO2, CO, O3, NO2, and labels, the dataset experiences a class imbalance. Hence, it uses the SMOTE technique to overcome it, and the accuracy results produced by the confusion matrix model are accuracy, precision, recall, f1-score, and ROC AUC.

3. METHODOLOGY

In this study, air quality classification will be carried out using the extreme gradient boosting algorithm, with the first step being to obtain a dataset. Then, the dataset will be preprocessed for ready-to-use datasets, divided into 80%, 20% for test data, and training data. Because the dataset has an unbalanced class, it will use methods to overcome it, namely SMOTE, Random oversampling, and Random under sampling. Then, after the dataset is ready for use, it will be classified using the extreme gradient boosting algorithm, and the final step is evaluation to test the best parameters and their accuracy with the confusion matrix. The research method is shown in the Figure 1.

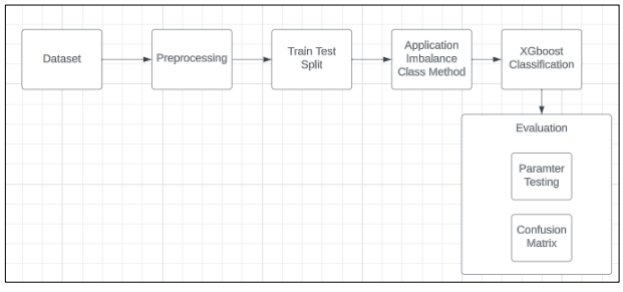

FIGURE 1. METHODOLOGY

3.1 Dataset

At this stage, the data to be used is air quality results data located in DKI Jakarta with a total of 1804 data records. The dataset used has similarities in data attributes with previous research regarding air quality classification [4] but added a new attribute in this study. The description of the dataset is shown in Table 1.

TABLE 1. DATASET DESCRIPTION

NO	Attribute Data	Description
1	Date	Date of obtaining air quality
2	Region	Location air quality measurement
3	PM10	Airborne particles with a size of 10 microns
4	PM 2,5	Airborne particles with a size of 2.5 microns or less
5	SO2	Pollutant gas caused by combustion which contains elements of sulfur
6	CO	Gas produced from combustion containing carbon
7	O3	Layer of air produced from oxygen caused by electricity or the influence of sunlight
8	NO2	Gas produced from vehicles, household and industrial activities
9	MAX	The highest value of one of the air quality parameters
10	Critical	Parameters with the highest measurement
11	Category	Good, Moderate and Unhealthy

The description of the data attributes in the table above explains the information on the dataset attributes, by having 11 data attributes used in this study. The research dataset that will be used is shown in the Table 2.

TABLE 2. RESEARCH DATASET

Attribute Data	Value	Value	…	Value
Date	2021-01-01	2021-01-02	…	2021-01-11
Region	DKI1	DKI1	…	DKI3
Station	Bunderan HI	Bunderan HI	…	Jagakarsa
PM10	38	27	…	84
PM2,5	53	46	…	112
SO2	29	27	…	20
CO	6	7	…	33
O3	31	47	…	57
NO2	13	7	…	10
MAX	53	47	…	112
Critical	PM2,5	O3	…	112
Category	Moderate	Good	…	Unhealthy

3.2 Preprocessing Data

The preprocessing stage is the stage for converting data into data ready to be tested. At this stage, data is processed through data cleaning, feature selection, and encoding stages. Data cleaning is cleaning data from noisy and inconsistent data to eliminate data quality problems affecting the analysis results so that the data is ready for use [19]. In the dataset obtained, there needs to be value-added. Checking data that has a missing value is shown in Figure 2.

```
pm10          0
pm25         60
so2           0
co            0
o3            0
no2           0
categori      1
dtype: int64
Total missing values: 61
```

FIGURE 2. CHECKING THE NUMBER OF MISSING VALUES

A value in the pm2.5 attribute 60 and a category 1 must exist. To overcome this missing value, the pm2.5 attribute will be filled in with the mean of the pm2.5 attribute. This is because research [20] Classified using an AdaBoost algorithm compares to fill in the missing value using the mean, median & mode. The classification results get greater accuracy when filling in the missing value with the mean. For the Category attribute, delete rows or columns containing missing values. The dataset that has been above the missing value is shown in Figure 3

```
pm10          0
pm25          0
so2           0
co            0
o3            0
no2           0
categori      0
dtype: int64
```

FIGURE 3. AFTER CLEANING THE DATA

Furthermore, the feature selection process is carried out, namely removing features that do not provide significant information. Dataset before feature selection shown in Figure 4.

	tanggal	stasiun	pm10	pm25	so2	co	o3	no2	max	critical	categori
0	2021-01-01	DKI1 (Bunderan HI)	38	53	29	6	31	13	53	PM25	SEDANG
1	2021-01-02	DKI1 (Bunderan HI)	27	46	27	7	47	7	47	O3	BAIK
2	2021-01-03	DKI1 (Bunderan HI)	44	58	25	7	40	13	58	PM25	SEDANG
3	2021-01-04	DKI1 (Bunderan HI)	30	48	24	4	32	7	48	PM25	BAIK
4	2021-01-05	DKI1 (Bunderan HI)	38	53	24	6	31	9	53	PM25	SEDANG

FIGURE 4. DATASET BEFORE FEATURE SELECTION

	pm10	pm25	so2	co	o3	no2	categori
0	38	53	29	6	31	13	SEDANG
1	27	46	27	7	47	7	BAIK
2	44	58	25	7	40	13	SEDANG
3	30	48	24	4	32	7	BAIK
4	38	53	24	6	31	9	SEDANG

FIGURE 5. DATASET AFTER FEATURE SELECTION

In this study, the dataset used will be cleaned and selected to determine the attributes that the system will operate. The data attributes used from this process will carry out the cleaning and selection process. The selected attributes are PM_{10}, $PM_{2.5}$, SO_2, CO, O_3, NO_2, Category. This is because the attributes obtained with the ISPU are adjusted. Dataset after feature selection shows in Figure 5 above.

3.3 Train Test Split

Splitting data into training and testing sets yields more precise outcomes when applied to new or unfamiliar data [21]. Training and test data will be divided into 80% and 20% by implementing unbalanced class methods, namely SMOTE, Random Over Sampling, and Random Under sampler.

3.4 Application Imbalance Class Method

Class reference Refers to a situation where the number of instances of one class exceeds the other types in the data set. The class with the most samples is the majority class, while the class with fewer examples is the minority class [22]. This research will use imbalance class methods, namely SMOTE, Random oversampling, and Random under sampling.

SMOTE is an oversampling technique used to overcome class imbalance by increasing the number of samples in the minority class. The Minority Over-Sampling Synthetic Sampling (SMOTE) technique provides effective results. It helps overcome the class imbalance problem by reducing the weakness of the over-sampling method that is excessive on minority classes. SMOTE generates synthetic examples of minority classes that behave in the feature dimension instead of the data space. The technique creates new synthetic examples by extending the minority sample range using random samples from k nearest neighbors, mimicking the sample pattern of the minority class. By generating synthetic examples of minority class cases, the technique expands the scope of decision making for the minority class [23]. SMOTE visualization is shown in the Figure 6.

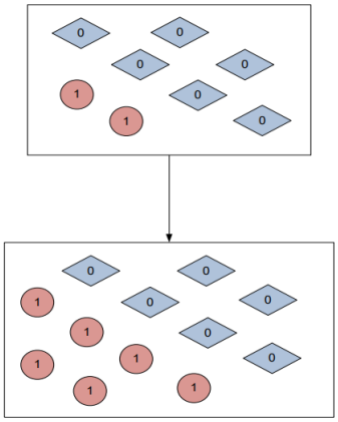

FIGURE 6. SMOTE VISUALIZATION

Under sampling is reducing the number of samples from the majority class. Some standards under sampling methods include tomes link, cluster centroid, and others. Under sampling can eliminate valuable data for classification models but is useful when the data is very large. Oversampling is increasing the number of samples from the minority class. Random under sampling is a technique used to overcome class imbalance in a dataset by reducing the number of samples from the majority class. This technique is suitable for datasets with class imbalance where the majority class has a significantly larger sample than the minority class. Oversampling methods can generate new examples or repeat some examples. An example of an oversampling method is Borderline-SMOTE. Oversampling can improve model performance by providing more information about the minority class. Thus, under sampling reduces the number of samples from the majority class, while oversampling increases the number of samples from the minority class [24]. the difference between under sampling and oversampling is shown in Figure 7.

FIGURE 7. DIFFERENCE BETWEEN UNDERSAMPLING AND OVERSAMPLING

3.5 XGBoost Classification

The Extreme Gradient Boosting (XGBoost) method is a boosting technique that uses a collection of decision trees. The trees in XGBoost are built sequentially, where the construction of each tree depends on the previous tree. The first tree in XGBoost has weak classification performance with user-defined initial probabilities. However, the weight of each tree built will be updated to produce a stronger set of classification trees with an objective function equation [25][26] as in the following equation:

$$obj(\theta) = \sum_{i}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \omega\,(f_k) \qquad (1)$$

Where $\sum_{i=1}^{n} l\,(y_i, \hat{y}_i)$ is a differentiable loss function to measure whether the model is suitable for the training dataset and $\sum_{k}^{K} \omega\,(f_k)$ is an item that defines the complexity of the model. As the complexity of the model increases, the corresponding score is reduced in value [27]. Gradient Boosting is an approach to regression and classification in predictive model building. This approach consists of a set of weak learnings combined to form a more robust model. In optimizing the model, evaluation is carried out using a loss function. The smaller the value of the loss function, the higher the model performance. Each iteration step builds weak learnings to provide more accurate predictions than the previous iteration. The development of more traditional gradient-boosting methods has resulted in more efficient implementations and more accurate predictions. A variant known as Extreme Gradient Boosting was also introduced to increase the model's performance even more[18],

The accuracy of classification results using XGBoost depends on setting specific parameters. Below are some parameters that can be adjusted in the XGBoost algorithm to improve classification accuracy. The parameters used in this study can be seen in Table 3.

TABLE 3. XGBOOST PARAMETERS

Parameter	Description
max_depth	Parameter to determine the maximum depth of each tree in the ensemble.
n_estimators	The parameter determines the number of trees to be built during the ensemble.
learning rate	The parameter for determining the maturity of the model's learning rate

3.6 Evaluation

The confusion matrix is a table that evaluates model performance in classification tasks. This table compares the predictions generated by the model with the actual table from the test data. The confusion matrix can assess how a classification model works, identify areas where the model can be improved, and make better decisions based on the model's performance in predicting classes. The Confusion Matrix consists of four central cells, namely True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) [23].

4. RESULT AND DISCUSSION

4.1 Experimental Testing

At this testing stage, experiments will be carried out to overcome the problem of class imbalance or class imbalance with the methods used, namely SMOTE, Random Undersampling, and Random OverSampling. After that, each method will be experimented with to find the best parameters. It aims to get the highest accuracy results for the three methods by looking for the best parameters. From the comparison of the three methods and parameter experiments carried out, the best accuracy results were obtained from each of the class imbalance methods. The results of the comparison of the three imbalance class methods can be seen in Table 5.

TABLE 4. COMPARISON OF ACCURACY RESULTS FROM THE IMBALANCE CLASS METHOD

Metode Imbalance Class				
	Accuracy	Precision	Recall	F1-Score
SMOTE	98,61%	99%	97%	98%
Random OverSampling	97%	99%	90%	94%
Random UnderSampling	98%	97%	97%	97%

4.1.1 Parameter Testing & SMOTE

The experiment used the SMOTE (Synthetic Minority Over-Sampling Technique) method with the number of labels shown in the Figure 8.

```
Jumlah label setiap kelas setelah SMOTE:
1    1073
2    1073
0    1073
```

FIGURE 8. NUMBER OF CLASSES AFTER SMOTE

Then, the best parameters will be searched for the best accuracy, as shown in the table 6.

TABLE 5. PARAMETERS AND SMOTE EXPERIMENTS

Train – Test 80% - 20%			
Max Depth	**n_estimators**	**Learning Rate**	**Accuracy**
6	30	0.01	98,61%
7	50	0.02	98,34%
8	75	0.03	98,34%
9	100	0.04	98,34%
9	125	0.05	98,06%

From the search results for the best parameters, namely Max Depth = 6, n_estimators = 30, Learning Rate = 0.01 for SMOTE to get 98.61% accuracy.

4.1.2 Parameter Testing & Random OverSampling

The experiment used the random oversampling method with the number of labels shown in the Figure 9.

```
Jumlah label setiap kelas setelah oversampling:
1    1073
2    1073
0    1073
Name: categori, dtype: int64
```

FIGURE 9. NUMBER OF CLASSES AFTER OVERSAMPLING

Then, the best parameters will be searched for the best accuracy, as shown in the Table 7.

TABLE 6. PARAMETERS AND OVERSAMPLING EXPERIMENTS

Train – Test 80% - 20%			
Max Depth	n_estimators	Learning Rate	Accuracy
6	30	0.01	96,25%
7	50	0.02	97,01%
8	75	0.03	97,34%
9	100	0.04	97,46%
9	125	0.05	96,37%

From the search results for the best parameters, namely Max Depth = 9, n_estimators = 100, and Learning Rate = 0.04 for the Random OverSampling, it gets an accuracy of 97.46%.

4.1.3 Parameter Testing & Random UnderSampling

The experiment used the random undersampling method with the number of labels shown in the Figure 10.

```
Jumlah label setiap kelas setelah undersampling:
0    154
1    154
2    154
Name: categori, dtype: int64
```

FIGURE 10. NUMBER OF CLASSES AFTER UNDERSAMPLING

Then, the best parameters will be searched for the best accuracy, as shown in the Table 8.

TABLE 7. PARAMETERS AND UNDERSAMPLING EXPERIMENTS

Train – Test 80% - 20%			
Max Depth	n_estimators	Learning Rate	Accuracy
6	30	0.01	94,94%
7	50	0.02	95%
8	75	0.03	96,96%
9	100	0.04	97,97%
9	125	0.05	98%

From the search results for the best parameters, namely Max Depth = 9, n_estimators = 125, Learning Rate = 0.05 for random undersampling to get 98% accuracy.

4.2 Accuracy Testing

At this stage, accuracy testing is carried out to test the accuracy of the results obtained from the model designed and used. This test uses the Confusion Matrix method to compare model predictions with actual tables on test data. Testing the classification results carried out using the XGBoost model was evaluated using the confusion matrix method of the three imbalance class methods. The value of the confusion matrix is shown in the Figure 11, Figure 12 and Figure 13.

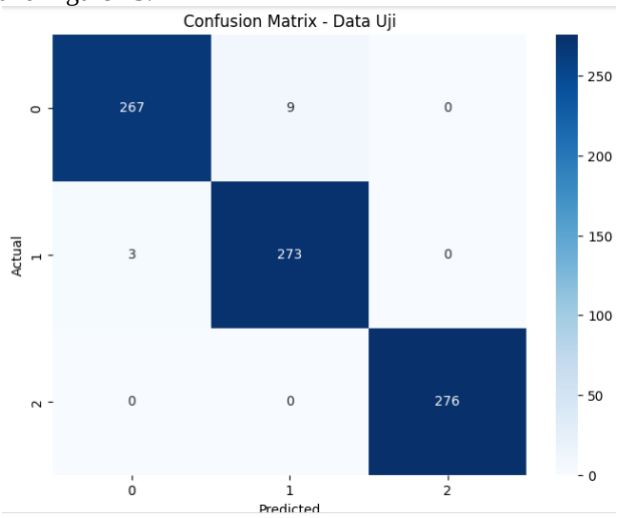

FIGURE 11. CONFUSION MATRIX SMOTE

Formula:

$$Accuracy = \frac{Total\ TP}{Total\ Data}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 - Score = \frac{2(Precision * Recall)}{Precision + Recall}$$

Results:

$$A = \frac{267 + 273 + 276}{267 + 9 + 0 + 3 + 273 + 0 + 0 + 0 + 276}$$

$$Accuracy = \frac{816}{828} = 98{,}61\%$$

$$Precision = \frac{267}{276} = 97\%$$

$$Recall = \frac{267}{266} = 100\%$$

$$F1 - Score = \frac{2(0.98.1)}{0.98 + 1} = 98\%$$

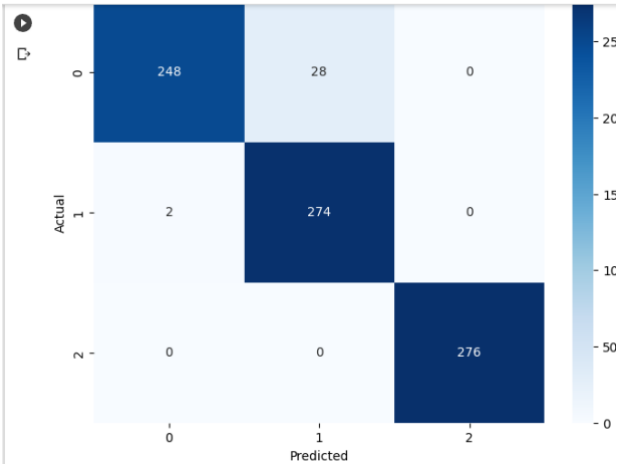

FIGURE 12. CONFUSION MATRIX RANDOM OVERSAMPLING

Formula:

$$Accuracy = \frac{Total\ TP}{Total\ Data}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 - Score = \frac{2(Precision * Recall)}{Precision + Recall}$$

Results:

$$Accuracy = \frac{248 + 274 + 276}{248 + 28 + 0 + 2 + 274 + 0 + 0 + 0 + 276}$$

$$Accuracy = \frac{798}{828} = 97\%$$

$$Precision = \frac{248}{276} = 89\%$$

$$Recall = \frac{248}{248} = 100\%$$

$$F1 - Score = \frac{2(0.89.1)}{0.89 + 1} = 94\%$$

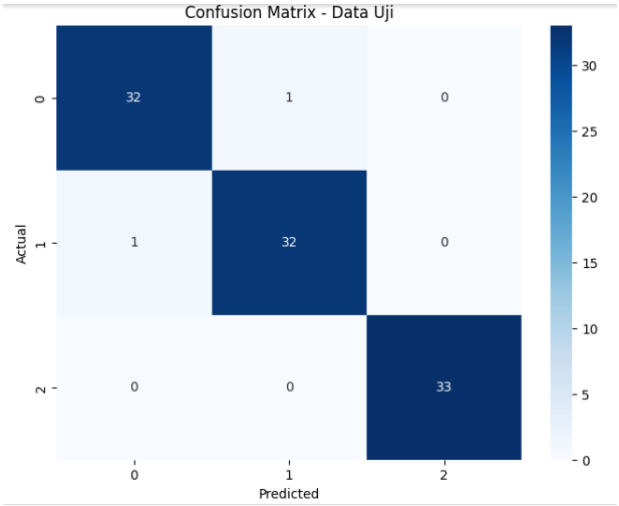

FIGURE 13. CONFUSION MATRIX RANDOM UNDERSAMPLING

Formula:

$$Accuracy = \frac{Total\ TP}{Total\ Data}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 - Score = \frac{2(Precision * Recall)}{Precision + Recall}$$

Results:

$$Accuracy = \frac{32 + 32 + 33}{32 + 1 + 0 + 1 + 32 + 0 + 0 + 0 + 33}$$

$$Accuracy = \frac{97}{99} = 98\%$$

$$Precision = \frac{32}{32 + 1}$$

$$Precision = \frac{32}{33} = 97\%$$

$$Recall = \frac{32}{32 + 1} = 100\%$$

$$F1 - Score = \frac{2(0.97.1)}{0.97} = 98\%$$

From the results of the discussion above, after comparing the imbalance class technique or unbalanced class and testing the best parameters, an evaluation was carried out with the confusion matrix. The best accuracy results were obtained with the confusion matrix with SMOTE 98.61%.

5. CONCLUSIONS

Based on this research, it was made using the Extreme Gradient Boosting algorithm. In this study, 80% of the data was used for training data, and 20% was used for test data. The dataset used experienced class imbalance, so a technique was used to overcome it, namely the SMOTE technique (Synthetic Minority Over-sampling Technique), Random Undersampling, and Random Oversampling and carried out experiments on the parameters used, namely the max_depth parameter to set the maximum depth of each tree in the ensemble, n_estimators to determine the total number of trees to be built during the ensemble and learning rate to determine which governs the model's learning rate. After experimenting with the extreme gradient boosting algorithm and overcoming unbalanced classes, 98.61% SMOTE, 97% Random Oversampling, and 98% Random Undersampling were produced. The best accuracy results were obtained using 98.61%, precision 97%, recall 100% and f1-score 98%. SMOTE. This study concludes that the Extreme Gradient Boosting algorithm can be applied to air quality classification measurements, and the best imbalance technique for air quality classification cases is SMOTE.

REFERENCES

[1] R. Satra and A. Rachman, "Pengembangan Sistem Monitoring Pencemaran Udara Berbasis Protokol ZIGBEE dengan Sensor CO," *Ilk. J. Ilm.*, vol. 8, no. 1, p. 17, 2016, doi: 10.33096/ilkom.v8i1.8.17-22.

[2] J. Abidin, F. Artauli Hasibuan, K. Kunci, P. Udara, and D. Gauss, "Pengaruh dampak pencemaran udara terhadap kesehatan untuk menambah pemahaman masyarakat awam tentang bahaya dari polusi udara," *Pros. Semin. Nas. Fis. Univ. Riau IV*, no. September, pp. 1–7, 2019, [Online]. Available: https://snf.fmipa.unri.ac.id/wp-content/uploads/2019/09/18.-OFMI-3002.pdf

[3] M. Rosyidah, "Polusi Udara dan Kesehatan," *J. Tek. Ind.*, vol. 1, no. 11, pp. 5–8, 2016.

[4] A. Sanmorino, J. Alie, N. Ariati, and S. V. Wulanda, "K-NN Based Air Classification as Indicator of the Index of Air Quality in Palembang," *SinkrOn*, vol. 7, no. 3, pp. 853–859, 2022, doi: 10.33395/sinkron.v7i3.11469.

[5] P. R. Peraturan Pemerintah No 41 Tahun 1999, "PP-No.41-th-1999-Pengendalian-pencemaran-Udara," no. 41, pp. 1–16, 1999.

[6] Peraturan Pemerintah RI, "Peraturan Menteri Lingkungan Hidup dan Kehutanan Republik Indonesia No 14 Tahun 2020 tentang Indeks Standar Pencemaran Udara," pp. 1–16, 2020.

[7] J. Wang, H. Li, and H. Lu, "Application of a novel early warning system based on fuzzy time series in urban air quality forecasting in China," *Appl. Soft Comput. J.*, vol. 71, pp. 783–799, 2018, doi: 10.1016/j.asoc.2018.07.030.

[8] A. Aziiz, H. Kirono, I. Asror, Y. Firdaus, and A. Wibowo, "Klasifikasi Tingkat Kualitas Udara Dki Jakarta Menggunakan Algoritma Naive Bayes," *eProceedings …*, vol. 9, no. 3, pp. 1962–1969, 2022, [Online]. Available: https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/18002%0Ahttps://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/18002/17631

[9] Y. Su, "Prediction of air quality based on Gradient Boosting Machine Method," *Proc. - 2020 Int. Conf. Big Data Informatiz. Educ. ICBDIE 2020*, pp. 395–397, 2020, doi: 10.1109/ICBDIE50010.2020.00099.

[10] Z. Qi, "The Text Classification of Theft Crime Based on TF-IDF and XGBoost Model," *Proc. 2020 IEEE Int. Conf. Artif. Intell. Comput. Appl. ICAICA 2020*, pp. 1241–1246, 2020, doi: 10.1109/ICAICA50127.2020.9182555.

[11] M. K. Nasution, R. R. Saedudin, and V. P. Widartha, "Perbandingan Akurasi Algoritma Naïve Bayes Dan Algoritma Xgboost Pada Klasifikasi Penyakit Diabetes," *e-Proceeding Eng.*, vol. 8, no. 5, pp. 9765–9772, 2021, [Online]. Available: https://journal.ubpkarawang.ac.id/mahasiswa/index.php/ssj/article/view/424/338%0Ahttps://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/15759

[12] M. R. Givari, M. R. Sulaeman, and Y. Umaidah, "Perbandingan Algoritma SVM, Random Forest Dan XGBoost Untuk Penentuan Persetujuan Pengajuan Kredit," *Nuansa Inform.*, vol. 16, no. 1, pp. 141–149, 2022, doi: 10.25134/nuansa.v16i1.5406.

[13] P. Studi and T. Lingkungan, "Analisa Deskriptif Pengelompokan Data Konsentrasi Pm2 , 5 Berdasarkan Hari Pada Titik Pemantauan," vol. 03, no. 01, pp. 42–48, 2022.

[14] M. Sahare and H. Gupta, "A review of multi-class classification for imbalanced data," *Int. J. Adv. Comput. …*, no. 3, pp. 1–5, 2012, [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi

=10.1.1.300.8687&rep=rep1&type=pdf

[15] F. Hamami and I. Fithriyah, "Classification of air pollution levels using artificial neural network," *2020 Int. Conf. Inf. Technol. Syst. Innov. ICITSI 2020 - Proc.*, pp. 217–220, 2020, doi: 10.1109/ICITSI50517.2020.9264910.

[16] A. T. Teologo, E. P. Dadios, R. Q. Neyra, and I. M. Javel, "Air Quality Index (AQI) Classification using CO and NO2 Pollutants: A Fuzzy-based Approach," *IEEE Reg. 10 Annu. Int. Conf. Proceedings/TENCON*, vol. 2018-Octob, no. 2, pp. 194–198, 2019, doi: 10.1109/TENCON.2018.8650344.

[17] H. Yi, Q. Xiong, Q. Zou, R. Xu, K. Wang, and M. Gao, "A Novel Random Forest and its Application on Classification of Air Quality," *Proc. - 2019 8th Int. Congr. Adv. Appl. Informatics, IIAI-AAI 2019*, pp. 35–38, 2019, doi: 10.1109/IIAI-AAI.2019.00018.

[18] A. A. Nababan, M. Jannah, M. Aulina, and D. Andrian, "Prediksi Kualitas Udara Menggunakan Xgboost Dengan Synthetic Minority Oversampling Technique (Smote) Berdasarkan Indeks Standar Pencemaran Udara (Ispu)," *JTIK (Jurnal Tek. Inform. Kaputama)*, vol. 7, no. 1, pp. 214–219, 2023, doi: 10.59697/jtik.v7i1.66.

[19] H. Sulastri and A. I. Gufroni, "Penerapan Data Mining Dalam Pengelompokan Penderita Thalassaemia," *J. Nas. Teknol. dan Sist. Inf.*, vol. 3, no. 2, pp. 299–305, 2017, doi: 10.25077/teknosi.v3i2.2017.299-305.

[20] G. Abdurrahman, "Jurnal Sistem dan Teknologi Informasi Klasifikasi Penyakit Diabetes Melitus Menggunakan Adaboost Classifier," *JUSTINDO (Jurnal Sist. dan Teknol. Informasi)*, vol. 7, no. 1, pp. 59–66, 2022, [Online]. Available: http://jurnal.unmuhjember.ac.id/index.php/JUSTINDO

[21] T. Pan, J. Zhao, W. Wu, and J. Yang, "Learning imbalanced datasets based on SMOTE and Gaussian distribution," *Inf. Sci. (Ny).*, vol. 512, pp. 1214–1233, 2020, doi: 10.1016/j.ins.2019.10.048.

[22] S. M. Abd Elrahman and A. Abraham, "A Review of Class Imbalance Problem," *J. Netw. Innov. Comput.*, vol. 1, pp. 332–340, 2013, [Online]. Available: www.mirlabs.net/jnic/index.html

[23] M. H. Ariansyah, S. Winarno, E. Nur Fitri, and H. M. Arga Retha, "Multi-Layer Perceptron For Diagnosing Stroke With The SMOTE Method In Overcoming Data Imbalances," *Innov. Res. Informatics*, vol. 5, no. 1, pp. 1–8, 2023, doi: 10.37058/innovatics.v5i1.6565.

[24] R. Mohammed, J. Rawashdeh, and M. Abdullah, "Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results," *2020 11th Int. Conf. Inf. Commun. Syst. ICICS 2020*, pp. 243–248, 2020, doi: 10.1109/ICICS49469.2020.239556.

[25] M. Syukron, R. Santoso, and T. Widiharih, "Perbandingan Metode Smote Random Forest Dan Smote Xgboost Untuk Klasifikasi Tingkat Penyakit Hepatitis C Pada Imbalance Class Data," *J. Gaussian*, vol. 9, no. 3, pp. 227–236, 2020, doi: 10.14710/j.gauss.v9i3.28915.

[26] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, vol. 13-17-Augu, pp. 785–794, 2016, doi: 10.1145/2939672.2939785.

[27] I. Muslim Karo Karo, "Implementasi Metode XGBoost dan Feature Importance untuk Klasifikasi pada Kebakaran Hutan dan Lahan," *J. Softw. Eng. Inf. Commun. Technol.*, vol. 1, no. 1, pp. 11–18, 2020.

AUTHORS

First Author
Albi Mulyadi Sapari
Student at Universitas Jendral Achmad Yani
Informatics Study Program

Second Author
Asep Id Hadiana
Chairman Of Informatics Department at Universitas Jendral Achmad Yani

Third Author
Fajri Rakhmat Umbara
Lecture Informatics Departement at Universitas Jendral Achmad Yani