Resampling Technique for Imbalanced Class Handling on Educational Dataset

Anief Fauzan Rozi[1], Adi Wibowo[2], Budi Warsito[3]

[1,3]School of Postgraduate Studies Diponegoro University, [2]Dept. Informatics Diponegoro University, Indonesia

[1]anieffauzanrozi@students.undip.ac.id, [2]bowo.adi@live.undip.ac.id, [3]budiwrst2@gmail.com

Abstract - Educational data mining is an emerging field in data mining. The need for accurate in identifying student accomplishment on a course or maybe an upcoming course can help the institution to build technology-aided education better. Educational data mining becoming a more important field to be studied because of its potential to produce a knowledge base model to help even the teacher or lecturer. Like another classification task, educational data mining has a common and frequently discovered problem. The problem that occurred in educational data mining specifically and classification tasks generally is an imbalanced class problem. An imbalanced class is a condition where the distribution of each class is not in the same proportion. In this research, it is found that the class distribution is severely imbalanced and it is a multiclass dataset that consists of more than two class labels. According to the problem stated beforehand, this paper will focus on the imbalanced class handling and classification with several methods on both of it such as Linear Regression, Random Forest and Stacking for classification and SMOTE, ADASYN, and SMOTE-ENN for the resampling algorithm. The methods are being evaluated using a 10-fold cross-validation and an 80-20 splitting ratio. The result shows that the best performance coming from the Stacking classification on ADASYN resampled dataset evaluated using an 80-20 splitting ratio with a 0.97 F1 score. The result of this study also shows that the resampling technique improves classification performance. Even though the no-resampling classification result produced a decent result too, it can be caused by several things such as the general pattern of the data for each class is already been good from the start. Thus, there is no real drawbacks if the original data is processed.

Keywords: Imbalanced class, ADASYN, SMOTE, SMOTE-ENN, educational data mining

I. INTRODUCTION

Educational data mining is an emerging field in data mining. The need for accuracy in identifying student accomplishment on a course or maybe an upcoming course can help the institution to build technology-aided education better[1]. Educational data mining becoming a more important field to be studied because of its potential to produce a knowledge base model to help even the teacher or lecturer [2]. Several studies have been conducted in the educational data mining field such as [3] which predicts the drop-out potential of a student using the Random Forest classification method, and another study conducted by [2] which measures the potential of an engineering student success based on their first three-year study performance. Other studies such as [1], [4]–[7] are several studies that focus on educational data mining discussion.

Like another classification task, educational data mining has a common and frequently discovered problem. The problem that occurred in educational data mining specifically and classification tasks generally is an imbalanced class problem. An imbalanced class is a condition where the distribution of each class is not in the same proportion [8]. The existence of imbalanced data can disrupt the classification performance as it makes the classification result biased toward the majority class. To make a fair classification process for each of the class(es), the imbalanced class needs to be overcome first. There are two mechanisms to handle imbalanced data, data level approach and algorithm approach [9]. The data level approach allows the dataset to be altered in terms of volume, either it is trimmed or added using a resampling technique. The reduction and the addition is depending on which data resampling algorithm. The data-level approach has two kinds of resampling mechanisms that are oversampling and undersampling [10]. Undersampling works by reducing the amount of data from the majority class to be balanced with the minority class [11]. Otherwise, oversampling is adding synthetic data into the minority class to be balanced with the majority class [12]. While undersampling potentially information trimming from the data because of the data reduction behavior, oversampling preserves the general existing information without any information reduced from the data. Synthesized data of the minority class is generated to preserve the characteristics of the original dataset on the certain class that have less amount of data from other class.

Student performance dataset possible to have imbalanced data according to the student's grade distribution in a course class. To preserve the general pattern and to make sure the classification's result is appropriate, the dataset will be undergoing a resampling step if there are any imbalanced class problems on the dataset. In this research, it is found that the class distribution is severely imbalanced and it is a multiclass dataset that consists of more than two class labels. According to the problem stated beforehand, this paper will focus on the imbalanced class handling and classification with several methods on both of it such as Linear Regression, Random Forest and Stacking for classification and SMOTE, ADASYN, and SMOTE-ENN for the resampling algorithm. The expected outcome from this research is the understanding of how those methods interact and performs toward this dataset, and whether the resampling method can improve the accuracy of the classification process in comparison with the no-resampling classification.

II. METHOD

This research was conducted using a private dataset of students' performance in several courses at the university, especially in the Information System program. Each student activity is used to determine their performance grade as the class of the dataset. The grading mechanism has done manually based on their score on the final exam and the activeness of the student. Each student is graded into three categories named, High, Middle, and Low representing how well they perform in the class. According to the distribution of the class after being labeled, the imbalanced class occurrence indication is strong in this dataset, so the imbalance class handling is being used to provide a better and fair classification of the data. This study follows the schematic diagrams shown in Fig. 1.

A. Dataset

This research use students' course results from three different courses at our university. The dataset has been labeled as "Low", "Medium", and "High" according to their class performance on each course that represents their class activities and exam scores. The dataset contains 298 records along with 7 predictor variables such as Course, Attendance(%), MidTerm, 1stAssessment, 2ndAssessment, Class_Activity, and one class attribute. The sample of the dataset can be seen in Table I.

From Table I can be seen that the dataset is a combination of numerical and nominal features. It needs to be encoded before being processed by any data mining method. The encoding steps will be carried out in the pre-processing steps by changing all of the nominal attributes into numerical using unique integers value such as "Yes - No" into 1 – 0 and so on. Before undergoing any of the data mining steps, exploratory data analysis will be carried out on this dataset to get a better understanding of the data characteristics and probably find something important to be addressed. First, the distribution of each class will be visualized to know how the dataset is classified and how many records belong to each class. The distribution of the class can be seen in Fig. 2.

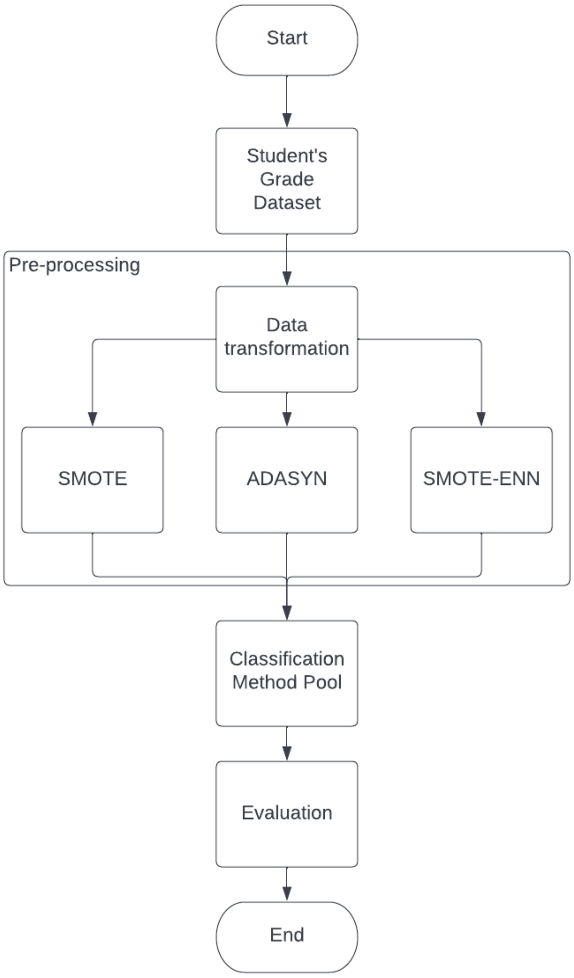

Fig. 1 Research design

TABLE I
STUDENT'S DATA

No	Course	Attendance(%)	MidTerm	1stAssessment	2ndAssessment	Class_activity	class
1	PSI	29	16.67	No	Yes	No	Low
2	PSI	21	83.33	Yes	No	No	Middle
3	PSI	86	40.00	Yes	Yes	Yes	Middle
4	PSI	0	1.00	Yes	No	No	Low
5	PSI	29	90.00	Yes	Yes	No	High
6	PSI	29	73.33	No	No	No	Low
7	PSI	21	43.33	Yes	No	No	Low
8	PSI	0	36.67	No	No	Yes	Low
9	PSI	0	33.33	No	No	No	Low
10	PSI	100	93.33	Yes	Yes	Yes	High
11	PSI	100	90.00	Yes	Yes	Yes	High
12	PSI	93	66.67	Yes	Yes	Yes	High
13	PSI	93	60.00	Yes	Yes	Yes	High
14	PSI	100	60.00	Yes	Yes	Yes	High
15	PSI	100	70.00	Yes	Yes	Yes	High
…	…	…	…	…	…		…
269	eGov	29	45.00	Yes	Yes	No	Low

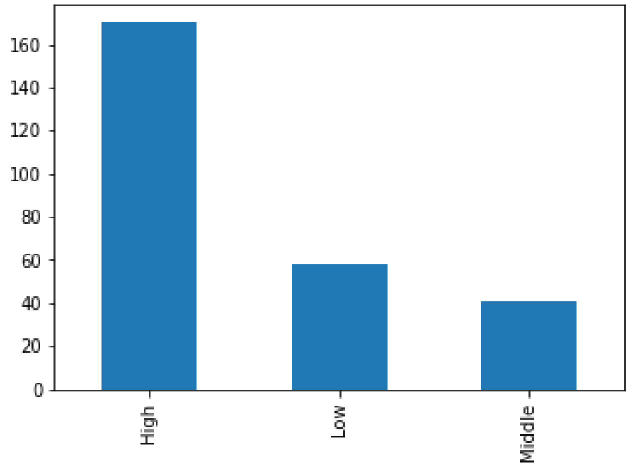

Fig. 2 Class distribution

B. *Preprocessing*

According to Fig. 2, it can be seen that the class distribution is skewed or can be described as imbalanced distribution among the classes. Under the imbalanced situation, any classification method has the probability to be biased toward the majority class because it has more data that the other class which is less than the majority class. The imbalanced problem needs to be addressed before the dataset being process further into the classification step because the ratio of the majority and minority classes is significantly different. The class data counts for each class are High: 170, Low: 58, and Middle: 41. Even though the imbalanced condition can be harmless to the classification performance. It is important to make sure that the classification result gives its optimum result. Therefore, the imbalanced condition will be handled first using an appropriate mechanism which is resampling.

Resampling is a mechanism to relatively "balance" the dataset by either reducing the majority sample or synthesizing the minority sample to make the distribution of each class become relatively the same. the name of both mechanisms is undersampling and oversampling respectively. Both of the resampling mechanisms have their own advantages and disadvantages. But, oversampling is a more widely-used resampling mechanism according to studies on imbalanced data or even a hybrid one (oversampling method + undersampling method).

As for this study, three resampling techniques will be implemented namely, SMOTE[13], ADASYN[14], and SMOTE-ENN[13]. The first two methods are oversampling and the last one is a hybrid method. By using different resampling methods, a comparison of which method performs better on the classification method can be expected. So that the justification of why one's method is preferably used on certain datasets' conditions can be supported.

SMOTE works by synthesizing the minority sample of the data considering the neighborhood sample data of minority class(es). New data points are generated using (1).

$$X_{new} = X_i + (\hat{X}_i - X_i) \times \delta \qquad (1)$$

where X_i is the random sample from k neighbor and δ is a random number in the interval [0,1].

ADASYN is also an oversampling method, it is a further development of SMOTE which works more adaptively to make the dataset relatively balanced. The steps of the ADASYN method are as follows [14].

1. Evaluate the degree of imbalance of all classes, $d = m0/m1\,, d \in \{0,1\}$.
2. Calculate the number of the sample produced $G = (m1 - m0) \times\ \beta, \beta \in [0,1]$ represents the expected level of unbalance after data generation. If $\beta = 1$ means that the class sample is fully balanced after data generation.
3. For each sample from the minor class, find the deepest K in n-dimensional space. Calculate $Ii = \Delta i/k\ \ (i = 1,2,\dots,m). Ii \in [0,1]$ where is the sample number that belongs to the main class and also the k-nearest neighbor of xi.
4. Regularize Ii according to $Ii = Ii/\sum_i^m Ii$, then Ii is the probability distribution, and $\sum Ii = 1$.
5. Count the number of samples *xi* in the minor class to produce $gi = Ii \times G$.
6. Choose a sample of the k-nearest neighbors from the small class. Synthesize a new sample S_z, where $S_z = (xj + xi) \times \lambda, \lambda \in [0,1]$ is a random number.
7. Repeat step 6, *gi* times to get the sample *gi xi*.

A combination of SMOTE with an undersampling gives better performance [15]. One of the popular hybrid methods of oversampling-undersampling technique is SMOTE-ENN which was developed by [13]. This method combines SMOTE's ability to generate a synthetic data sample of minority classes and the ability of ENN to delete some records from all classes that are identified as having different classes between the records and its k-nearest neighbor majority class. The process of SMOTE-ENN is as follows:

1. (Start of SMOTE) Choose random data from the minority class.
2. Calculate the distance between the random data and its k-nearest neighbors.
3. Multiply the difference with a random number between 0 and 1, then add the result to the minority class as a synthetic sample.
4. Repeat steps 2-3 until the desired proportion of minority class is met. (end of SMOTE)
5. (Start of ENN) determine K, as the number of k-nearest neighbors. If not determined, then K=3.
6. Find the k-nearest neighbor of the record among the other record in the dataset, then return the majority class from the k-nearest neighbor.
7. If the class of the observation and the majority class from the record's k-nearest neighbor is different, then the record and its k-nearest neighbor are deleted from the dataset.
9. Repeat steps 6 and 7 until the desired proportion of the dataset is met.

C. Classification

Several classification methods are used in this research to show how each of them performs toward the dataset with different resampling mechanisms. The classification method used is Logistic Regression, Random Forest, and ensemble technique Stacking.

Logistic Regression is a classification model that works on the categorical type of dependent variable; can be 0 and 1, true and false, etc. The logistic regression equation is stated as (2) and (3).

$$Ln\left(\frac{p}{1-p}\right) = B_0 + B_1X \qquad (2)$$

where:
B_0 = constant
B_1 = the coefficient of each variable

The p value or probability (Y = 1) can be found by the equation:

$$p = \frac{e^{(B_0+B_1X)}}{(1+e^{(B_0+B_1X)})} \qquad (3)$$

This equation can be used to calculate the probability of a respondent having a variable value that has been defined in the equation, the final result of the p-value will of course be in the range of 0 -1 [1], [16], [17].

Random Forest is a classification method that consists of a large number of individual decision trees that operate as an ensemble (Fig. 3). Each tree in the random forest produces a class prediction and the class with the most votes becomes the model's prediction [1], [18], [19].

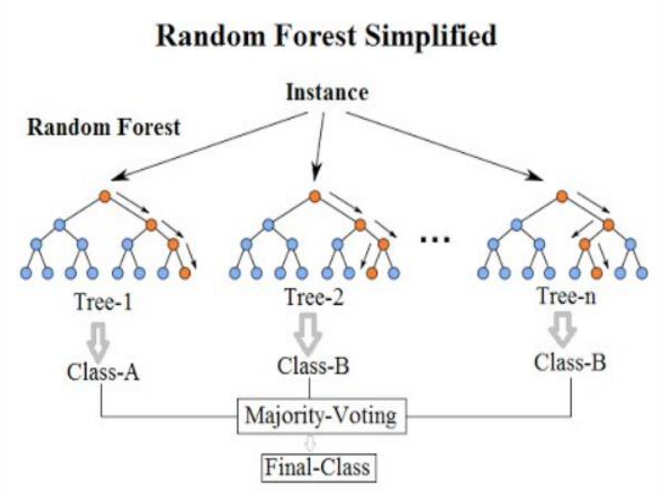

Fig. 3 Random Forest flow simplification [18]

The stacking technique allows several classification methods to work together and produce a better and stronger classification model. It learns in parallel manners, then combines them by training a meta-learner to output a prediction based on the other learner's prediction. A meta learner inputs the prediction as the features and the target being the ground truth values. It attempts to learn how the best combine the input predictions to make better output predictions[20]–[23]. The illustration of how the stacking technique works can be seen in Fig. 4.

III. RESULT AND DISCUSSION

Experiments were carried out on the dataset that had been set beforehand and will be explained in this section. The experimental scenario is described in the following section.

A. *Experimental setup*

The validation model used in this study is 10-fold cross-validation and 80-20 random splitting ratio. Each of the combinations of classification and resampling methods will be measured by both evaluation metrics. The experiment schema will be shown in Table II.

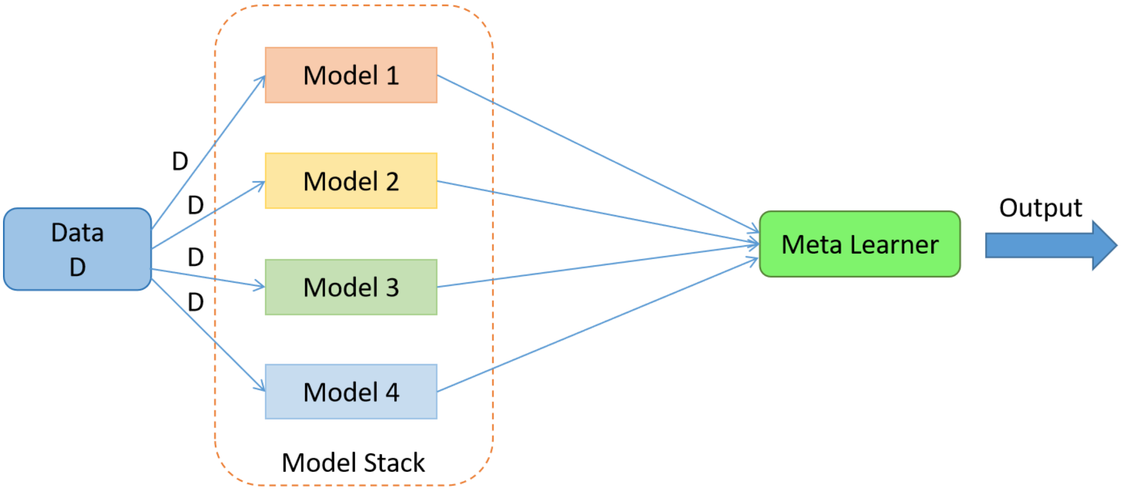

Fig. 4 Stacking algorithm flow [24]

TABLE II
EXPERIMENT SCHEMAS

Classifier	Resampling	10-Fold	80:20 Splitting
Logistic Regression	SMOTE ADASYN *SMOTE-ENN*	F1 Score	F1 Score
Random Forest	SMOTE ADASYN *SMOTE-ENN*	F1 Score	F1 Score
Stacking	SMOTE ADASYN *SMOTE-ENN*	F1 Score	F1 Score

The overall performance is expressed by the F1 score. Accuracy is not used as a performance measurement due to bias to the class imbalance problem.

B. *Experiment Result*

In the pre-processing step, each of the nominal data will be transformed into a numerical will with a unique integer encoding type. Each data is transformed into numeric to enable the calculation process of the resampling and classification result. One of the results of the resampling steps can be seen in Fig. 5.

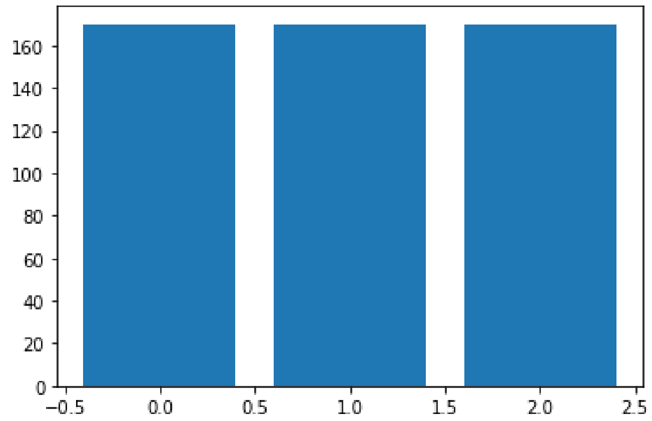

Fig. 5 Resampled data distribution using SMOTE

Fig. 5 shows the resampling result of SMOTE, which balances the class distribution for each class to the same amount of data (170 data records). The X-axis value represents the encoded classes 0 for High, 1 for Middle, and 2 for Low.

Table III provides information on the classification results of each resampling method along wing the evaluation metric used in this study.

As shown in Table III, each method combination produces a different result. As a comparison, a no-resampling classification is done to give a comparison to the usage of the resampling method toward the dataset. The most remarkable result was shown by the stacking algorithm along with the ADASYN resampling method on the 80:20 splitting evaluation metric with 0,97. The 10-fold stacking algorithm is on par with the Random Forest result at 0.90. the comparison between the resampled and no-resampling scores is not that far. It can be caused by several things such as the general pattern of the data for each class is already been good from the start. Thus, there is no real drawback if the original data is processed. The other reason is probably the method used in this research is not optimal yet, there is possibly a better resampling method that can be used for this kind of dataset. But one thing that shows is resampling technique improves the performance of the classification method to some extent.

To get a better understanding of the data, a feature importance calculation step is done under the Random Forest classification process. According to the feature importance calculation, 2nd Assessment (feature 4) has the biggest score among the others. The feature importance visualization can be seen in Fig. 6.

TABLE III
EXPERIMENT RESULT

Classifier	Resampling	10-Fold	80:20 Splitting
Logistic Regression	SMOTE	0,78	**0,88**
	ADASYN	0,75	0,83
	SMOTE-ENN	0,818	0,78
	No-resampling	0,83	**0,88**
Random Forest	SMOTE	0,90	**0,96**
	ADASYN	0,89	0,95
	SMOTE-ENN	0,89	0,86
	No-resampling	0,85	**0,96**
Stacking	SMOTE	0,90	**0,96**
	ADASYN	0,89	**0,97**
	SMOTE-ENN	0,87	0,86
	No-resampling	0,83	0,95

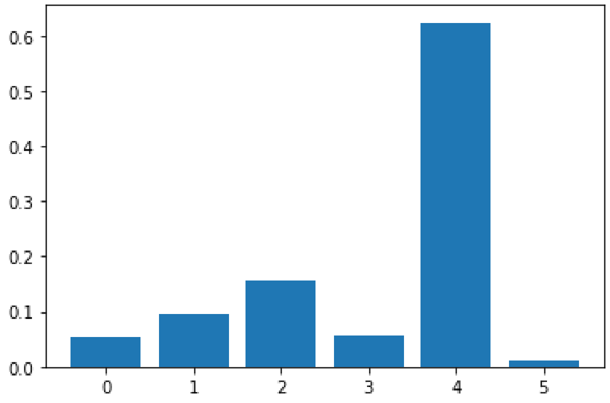

Fig. 6 Feature importance score

According to Fig. 6, can be seen that feature 4 (2nd Assessment) has the greatest importance score. It can be implied that the feature plays a great role to determine which class a student is included in. After the importance level is known. Further analysis of the data in terms of the relationship between the feature with a certain part of the class (Low Grade) is being done. The result of the analysis can be seen in Fig. 7.

From Fig 7 can be seen that the majority of the student in the "Low" grade class did not accomplish the 2nd Assessment. Referring to the feature importance score it can be implied that how a student performs in the class is most likely influenced by the accomplishment of the 2nd Assessment of the course. Another feature that is secondly important according to the feature importance is the MidTerm feature. The mean distribution of the MidTerm Score in each class is represented as shown in Fig. 8.

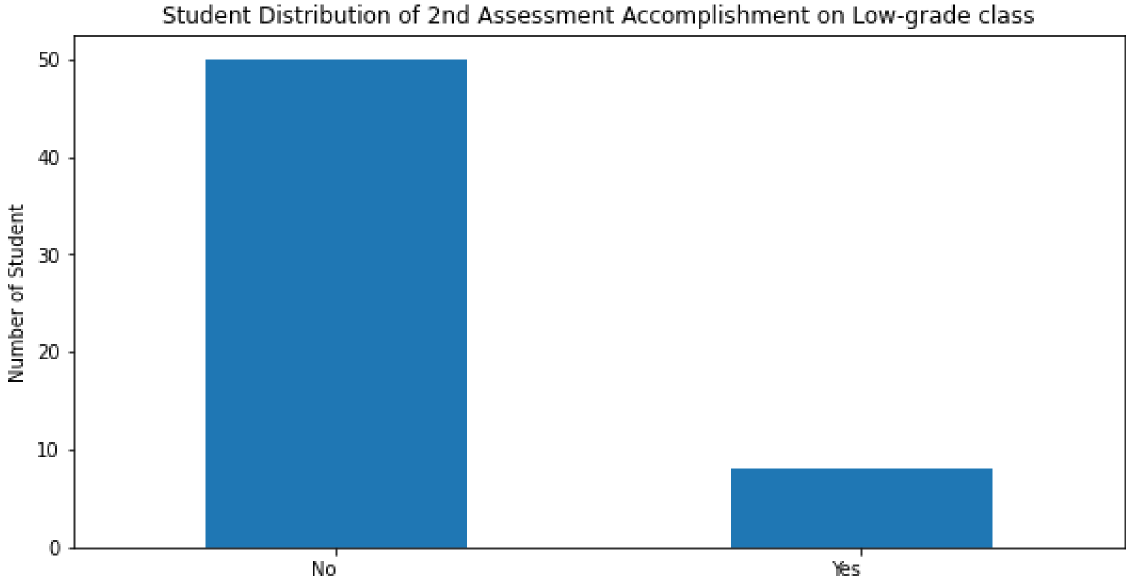

Fig. 7 Student distribution of 2nd assessment on low class

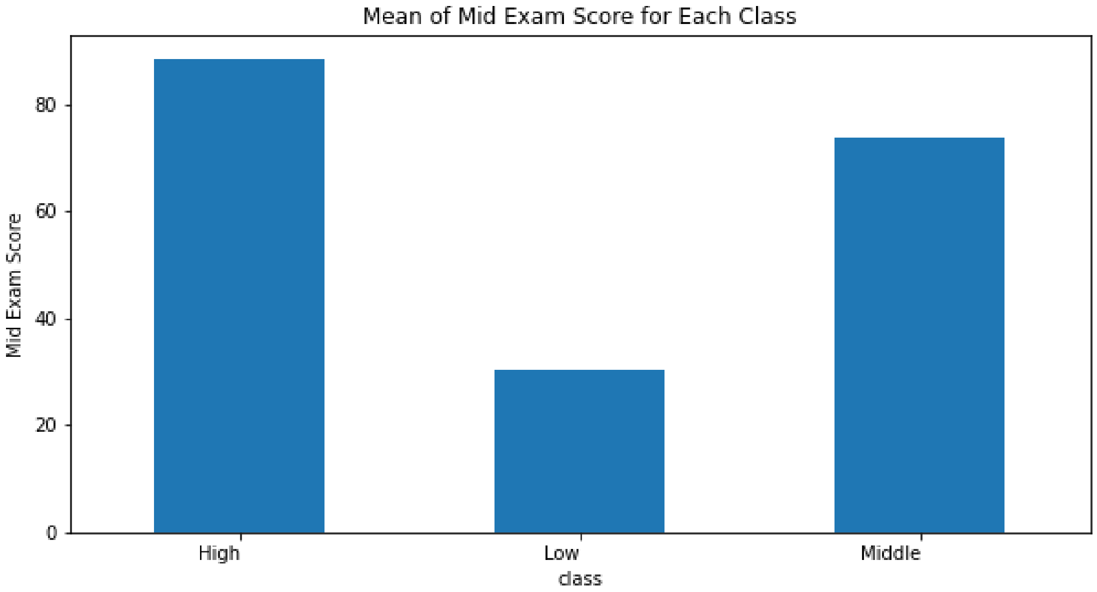

Fig. 8 Mean of midterm score on each class

According to Fig. 8, it can be seen that the Low-grade Class has the lowest mean of the MidTerm feature score. It implied that even if a student completes the 2nd Assessment but they have a low MidTerm Score, they are still included in the Low-grade Class. As an initial knowledge that if a student did not accomplish the 2nd Assessment and has a low MidTerm score, then the student is more likely will not have a successful period in that particular course or class. Otherwise, if a student accomplished the 2nd Assessment then more likely the student will succeed in the class or course as all of the students in the High-grade class are accomplished the 2nd Assessment.

IV. CONCLUSION

Based on the result produced in this research, the stacking classification algorithm performs better than the other two classifiers (Linear Regression and Random Forest) with a 0.97 F1 score as the measurement of how well the classification under an imbalanced class is. The difference resulting from the classification process could be happened because of the nature of the classifier itself.

Stacking is theoretically superior to the other two classifiers because the stacking algorithm combines several single classifiers as one more powerful classifier. It means that the stacking algorithm compromises the drawbacks of the classifiers and enhances the advantage of those classifiers. The result of this study also shows that the resampling technique improves classification performance. Even though the no-resampling classification result produced a decent result too, it can be caused by several things such as the general pattern of the data for each class is already been good from the start. Thus, there is no real drawback if the original data is processed. The other reason is probably the method used in this research is not optimal yet, there is possibly a better resampling method that can be used for this kind of dataset. According to further data analysis after the data mining process, the feature importance process under the Random Forest classification method shows that the 2nd Assessment is the most important feature that mostly determines which class a student will be included. Referring to the finding, most of the students in the Low-grade class did not accomplish the 2nd Assessment and had a low score on the MidTerm feature. The MidTerm feature is the second important feature according to the feature importance process. Up to this point, how a student will perform in the class is only based on how they perform academically. It is possible to include their demographics or their daily life into consideration. It may give a better perspective of which kind of student will succeed in class and the one who will fail. The possible future research is adding more features from another aspect of the students.

REFERENCES

[1] R. Ghorbani and R. Ghousi, "Comparing Different Resampling Methods in Predicting Students' Performance Using Machine Learning Techniques," *IEEE Access*, vol. 8, pp. 67899–67911, 2020, doi: 10.1109/ACCESS.2020.2986809.

[2] A. I. Adekitan and O. Salau, "The impact of engineering students' performance in the first three years on their graduation result using educational data mining," *Heliyon*, vol. 5, no. 2, p. e01250, 2019, doi: 10.1016/j.heliyon.2019.e01250.

[3] M. Utari, B. Warsito, and R. Kusumaningrum, "Implementation of Data Mining for Drop-Out Prediction using Random Forest Method," *2020 8th Int. Conf. Inf. Commun. Technol. ICoICT 2020*, 2020, doi: 10.1109/ICoICT49345.2020.9166276.

[4] P. Dabhade, R. Agarwal, K. P. Alameen, A. T. Fathima, R. Sridharan, and G. Gopakumar, "Educational data mining for predicting students' academic performance using machine learning algorithms," *Mater. Today Proc.*, vol. 47, no. xxxx, pp. 5260–5267, 2021, doi: 10.1016/j.matpr.2021.05.646.

[5] M. Tsiakmaki, G. Kostopoulos, S. Kotsiantis, and O. Ragos, "Implementing AutoML in Educational Data Mining for Prediction Tasks," *Applied Sciences* , vol. 10, no. 1. 2020. doi: 10.3390/app10010090.

[6] E. A. Amrieh, T. Hamtini, and I. Aljarah, "Mining Educational Data to Predict Student's academic Performance using Ensemble Methods," *Int. J. Database Theory Appl.*, vol. 9, no. 8, pp. 119–136, 2016, doi: 10.14257/ijdta.2016.9.8.13.

[7] E. Buraimoh, R. Ajoodha, and K. Padayachee, "Importance of Data Re-Sampling and Dimensionality Reduction in Predicting Students' Success," *3rd Int. Conf. Electr. Commun. Comput. Eng. ICECCE 2021*, no. June, pp. 12–13, 2021, doi: 10.1109/ICECCE52056.2021.9514123.

[8] G. Kovács, "Smote-variants: A python implementation of 85 minority oversampling techniques," *Neurocomputing*, vol. 366, pp. 352–354, 2019, doi: 10.1016/j.neucom.2019.06.100.

[9] Y. Pristyanto, I. Pratama, and A. F. Nugraha, "Data level approach for imbalanced class handling on educational data mining multiclass classification," in *2018 International Conference on Information and Communications Technology, ICOIACT 2018*, 2018, vol. 2018-Janua. doi: 10.1109/ICOIACT.2018.8350792.

[10] V. S. Spelmen and R. Porkodi, "A Review on Handling Imbalanced Data," *Proc. 2018 Int. Conf. Curr. Trends Towar. Converging Technol. ICCTCT 2018*, pp. 1–11, 2018, doi: 10.1109/ICCTCT.2018.8551020.

[11] K. Borowska and J. Stepaniuk, "Imbalanced data classification: A novel re-sampling approach combining versatile improved SMOTE and rough sets," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9842 LNCS, no. 1, pp. 31–42, 2016, doi: 10.1007/978-3-319-45378-1_4.

[12] F. Sağlam and M. A. Cengiz, "A novel SMOTE-based resampling technique trough noise detection and the boosting procedure," *Expert Syst. Appl.*, vol. 200, no. April 2020, pp. 1–12, 2022, doi: 10.1016/j.eswa.2022.117023.

[13] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data," *SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 20–29, Jun. 2004, doi: 10.1145/1007730.1007735.

[14] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," *Proc. Int. Jt. Conf. Neural Networks*, no. 3, pp. 1322–1328, 2008, doi: 10.1109/IJCNN.2008.4633969.

[15] E. P. W. Mandala, E. Rianti, and S. Defit, "Classification of Customer Loans Using Hybrid Data Mining," *JUITA J. Inform.*, vol. 10, no. 1, p. 45, 2022, doi: 10.30595/juita.v10i1.12521.

[16] M. R. Romadhon and F. Kurniawan, “A Comparison of Naive Bayes Methods, Logistic Regression and KNN for Predicting Healing of Covid-19 Patients in Indonesia,” *3rd 2021 East Indones. Conf. Comput. Inf. Technol. EIConCIT 2021*, pp. 41–44, 2021, doi: 10.1109/EIConCIT50028.2021.9431845.

[17] B. A. Akinnuwesi *et al.*, “Application of intelligence-based computational techniques for classification and early differential diagnosis of COVID-19 disease,” *Data Sci. Manag.*, vol. 4, pp. 10–18, 2021, doi: https://doi.org/10.1016/j.dsm.2021.12.001.

[18] P. Cortez and A. Silva, “Using data mining to predict secondary school student performance,” *15th Eur. Concurr. Eng. Conf. 2008, ECEC 2008 - 5th Futur. Bus. Technol. Conf. FUBUTEC 2008*, vol. 2003, no. 2000, pp. 5–12, 2008.

[19] H. Jiawei and M. Kamber, *Data mining: concepts and techniques*. 2001. doi: 10.1002/1521-3773(20010316)40:6<9823::AID-ANIE9823>3.3.CO;2-C.

[20] J. Ling and G. Li, “A two-level stacking model for detecting abnormal users in Wechat activities,” *Proc. - 2019 Int. Conf. Inf. Technol. Comput. Appl. ITCA 2019*, pp. 229–232, 2019, doi: 10.1109/ITCA49981.2019.00057.

[21] C. F. Kurz, W. Maier, and C. Rink, “A greedy stacking algorithm for model ensembling and domain weighting,” *BMC Res. Notes*, vol. 13, no. 1, p. 70, 2020, doi: 10.1186/s13104-020-4931-7.

[22] K. Leartpantulak and Y. Kitjaidure, “Music genre classification of audio signals using particle swarm optimization and stacking ensemble,” *iEECON 2019 - 7th Int. Electr. Eng. Congr. Proc.*, pp. 1–4, 2019, doi: 10.1109/iEECON45304.2019.8938995.

[23] R. Sikora and O. Al-Laymoun, “A modified stacking ensemble machine learning algorithm using genetic algorithms,” *Artif. Intell. Concepts, Methodol. Tools, Appl.*, vol. 1, no. 1, pp. 395–405, 2016, doi: 10.4018/978-1-5225-1759-7.ch016.

[24] S. Džeroski and B. Ženko, “Is combining classifiers with stacking better than selecting the best one?,” *Mach. Learn.*, vol. 54, no. 3, pp. 255–273, 2004, doi: 10.1023/B:MACH.0000015881.36452.6e.