A Cost-Sensitive K-Nearest Neighbors (K-NN) Classification in Handling Multi-class Imbalance Problem

Adli A Nababan, Erna B Nababan, Muhammad Zarlis and Sutarman Wage

***Abstract*— The problem of unbalanced multiclass data greatly affects the classification process in machine learning. Unbalanced multiclass data is an interesting case to study until now. Several studies have shown that the minority class in the dataset is often considered unimportant or less influential than the majority class. This data imbalance problem arises when the minority class is misclassified. Misclassification causes low accuracy values and affects classifier performance. In this research, feature selection using Gain Ratio (GR) and Principal Component Analysis (PCA) were used to determine the selection of the most relevant features in the dataset. The classifier model is made by remodeling K-Nearest Neighbors (K-NN) with minimal cost using the MetaCost method. Then validate by calculating the value of accuracy and total cost of the proposed method in solving the problem of data imbalance in multiclass. The results of testing the K-NN algorithm using feature selection are proven to be able to improve performance, especially feature selection using the Gain Ratio method. The MetaCost method is also proven to be able to provide better performance. The combination of MetaCost and GR has better performance, wherefrom the two datasets there is an increase of 0.0295 in the Accuracy value, 0.1132 in the Precision value, 0.0897 in the Recall, and 0.1021 in the F1-Score.**

Keywords— cost-sensitive, k-nearest-neighbors, multiclass, classification, gain ratio, pca, metacost

I. INTRODUCTION

WITH the development of Big Data in the current Industrial Age 4.0, the data availability is increasing from time to time. A lot of stored data has information knowledge, but not all data can be used, processed, and used as new knowledge.

A data object is a collection of entities from data which is also called a dataset. Datasets based on class distribution are classified into binary data (has two classes) and multi-class data (more than two classes) [1], [2].

So far, the problem of data imbalance has focused on the literature on the problem of imbalance between two classes, while many problems arise in data that has more than two classes (multi-class) and is a problem that must be faced until now [3]. The class with a ratio of the number of instances that is more dominant than other classes is called the majority class, while the class with only a few or less dominant ratio of the number of instances is called the minority class. Because it only has a few instances, this minority class is often considered unimportant or less influential in machine learning. But in reality, in the real world, data with this minority class contains very useful information and has an impact that is no less important than the majority class. [2].

This class imbalance causes problems in the classification process in machine learning. The skewed data distribution in the classification process causes many conventional methods in machine learning to be less effective in predicting minority classes. When the data has a class imbalance, the classifier tends to be more dominant over the majority class which causes the minority class classification to be poor. [4]. Some research has imbalance problems such as coronary artery diagnosis [5], predict bank creditworthiness [6], and prediction of telecommunication subscriber churn [7].

Many researchers have proposed several techniques to solve the problem of skewed data distribution on binary data [8], however, the problem that arises and has a more difficult level to solve in the classification process is the problem of skewed data distribution in multi-class [9].

The problem of imbalance in multi-class data has been proposed by many researchers using techniques and ways to solve it [10]. However, it is necessary to develop new techniques to obtain a classifier that will provide optimal accuracy. In particular, classification techniques consider and provide high accuracy for the minority class without ignoring the majority class.

One of the effective techniques to solve the data imbalance problem is the cost-sensitive K-Nearest Neighbors (K-NN) [5], [10], [11]. The proposed technique is a combination of two methods, namely cost-sensitive learning and K-Nearest Neighbors. This technique works by creating a K-Nearest Neighbors model, which considers the minimal cost of misclassification [10-12]. The proposed K-Nearest Neighbors

Adli A Nababan, Doctoral Program, Department of Computer Science, Faculty of Computer Science and Information Technology Universitas Sumatera Utara, Medan, Indonesia (e-mail: adli.nababan@students.usu.ac.id).

Erna B Nababan, Department of Computer Science, Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Medan, Indonesia (e-mail: ernabrn@usu.ac.id)

Muhammad Zarlis, Department of Computer Science, Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Medan, Indonesia (e-mail: m.zarlis@usu.ac.id)

Sutarman Wage, Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Sumatera Utara, Medan, Indonesia (e-mail: sutarman@usu.ac.id)

model is expected to minimize the misclassification of unbalanced data, especially in multi-class data by considering cost-sensitive learning techniques. Therefore, developing new techniques in making the K-Nearest Neighbors model is necessary.

In this study, we focus on handling the problem of multiclass unbalance data using K-Nearest Neighbors. Predictive performance is measured comparatively based on accuracy using feature selection methods such as Gain Ratio (GR) and Principal Component Analysis (PCA). After feature selection, the MetaCost method is used in cost-sensitive learning.

The minimal re-cost model built by labeling and modifying the K-Nearest Neighbors model is expected to affect the performance and rate of the classification process. The learning objectives are carried out to compare the performance of K-Nearest Neighbors that are sensitive to costs, cost-sensitive K-Nearest Neighbors with Gain Ratio (GR), sensitive to costs of K-Nearest Neighbors with Principal Component Analysis (PCA) in designing the K-Nearest Neighbors model.

II. METHODOLOGY

A. Feature Selection

- *Gain Ratio (GR)*

  Gain Ratio (GR) is a development method of Gain Information by eliminating existing bias. GR is the determining parameter in selecting the features that are most relevant to the results by considering the intrinsic information from the data in the Decision Tree method. [13] explain the GR calculation steps as follows:

  Step-1: Calculate the entropy value using the equation (1)

  $$Entropy\ (S) = \sum_{i=1}^{n} - pi * log_2 pi \qquad (1)$$

  Step-2: Calculate the gain information with the equation (2)

  $$Information\ Gain\ (S, A) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} \times Entropy(S_i) \qquad (2)$$

  Step-3: Calculate split info with equation (3)

  $$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times log_2(\frac{D_j}{D}) \qquad (3)$$

  Step-4: Step-4: Calculate GR with equations (4)

  $$Gain\ Ratio\ (A) = \frac{Gain\ (A)}{SplitInfo(A)} \qquad (4)$$

- *Principal Component Analysis (PCA)*

  PCA can be used and applied to correlated attributes [15]. PCA can determine patterns in a data set, and find the same or different patterns between attributes. The data is selected and calculated using a covariance matrix then the results of these calculations can be used to calculate the eigenvectors and eigenvalues [16]. The main components of the data set selected based on the eigenvectors with the highest eigenvalues indicate a significant relationship between each attribute of the data set. The data were selected based on the most significant eigenvalues and the least significant data. The results of sorting the data based on the eigenvalues can reduce high-dimensional data to lower dimensional data.[17]. To determine the variance of the distribution of data on a dataset that has a data imbalance, it can be calculated to determine the deviation of the data in the sample dataset using the equation (5) [18].

  $$\mathrm{Var(x)} = \sigma^2 = \frac{1}{n} \sum_{i=1}^{n} (\tilde{z}_{ij} - \mu_j)^2 \qquad (5)$$

  Then, to find the relationship between each class, the covariance is calculated to indicate whether or not there is a relationship between the two dimensions. A zero value in the covariance indicates that the dimension is not related. [19] suggested calculating Covariance using the equation (6).

  $$\mathrm{Cov(x,y)} = \frac{1}{n-1} \sum_{i=1}^{n} (x_{ij} - \mu_{xj})(y_{ij} - \mu_{yj}) \qquad (6)$$

  The covariance matrix is calculated in the next step based on the eigenvalues and eigenvectors. The calculation results of the eigenvalues are transformed or called orthogonal varimax rotation using the equation (7)[19].

  $$\mathrm{Det\ (A - \lambda I)} = 0 \qquad (7)$$

B. K-Nearest Neighbors

K-Nearest Neighbors (K-NN) is one of the most popular supervised algorithms or methods in machine learning that is used to classify data, even [20] in his book put K-NN into the top ten algorithms in data mining. K-NN is also an algorithm that belongs to the type of distance-based algorithm. This algorithm in the classification process will calculate the similarity of the data based on the distance between the data and make the majority class the class for new data [13], [21]. In the K-Nearest Neighbors method to calculate the distance between the test data to each training data using the Euclidean distance, it can be seen using the equation (8)

$$D\ (x, y) = ||x - y||_2 = \sqrt{\sum_{j-1}^{N} |\mathrm{x} - \mathrm{y}|^2} \qquad (8)$$

C. Cost-sensitive Learning

- *MetaCost*

  One of the cost-sensitive methods used in the meta-learning thresholding approach is the MetaCost algorithm. MetaCost is used to minimize costs in the classification process [15]. The basic principle of MetaCost in classification is to calculate the probability value of each class in the data [16]. When the class probability value is > 1, then each class will be re-learned by trimming or re-labeling each data until the minimum cost is obtained in the classification process. The cost matrix is defined as C(i,j), where i is the actual class and j is the predicted class [22]. The cost with the minimum value will be used as a prediction class or a new leaf node to get optimal performance. Below is the procedure for implementing the MetaCost algorithm.

For each node *x* in *S*
For each class j

Let $P(j|x) = \frac{1}{\sum_i 1} \sum_i P(j|x, M_i)$

if $P(j|x, M_i) = 0$ is produced by M_i

$$Else\ P(j|x, M_i) > 1$$

$$\text{Let } x's\ class = argmin_i \sum_j P(j|x)C(i,j)$$

Let *M* = Models generated by applying training (*L*) to classification processes (S).

Return M

Where S is Training set, L is Classification learning algorithm, C is Cost matrix, M is The result of the number of leaf nodes, N is Number of nodes in each leaf node.

D. Training Process

n this study, we carried out several stages of research. The first stage is collecting data (i), the second stage is preprocessing the data (ii), then the third stage is selecting the data (iii), the fourth stage is conducting the training and testing process (iv), and the last stage is validating the method used and proposed (v). We will propose the following stages of the method to solve the classification problem on unbalanced multiclass data.

Step-1: Input the dataset for processing.
Step-2: Replacing the missing value and remove the outlier from the dataset and perform the min-max normalization process.
Step-3: Selecting the attributes (using GR or PCA).
Step-4: Creating the classifier model using K-Nearest Neighbors algorithm.
Step-5: Performing a remodeling of the K-Nearest Neighbors by minimizing the cost of using the MetaCost method.
Step-6: Performing validation by calculating the value of accuracy and total cost.

III. PROPOSED METHOD

In general, the method proposed in this study can be seen in Figure 1.

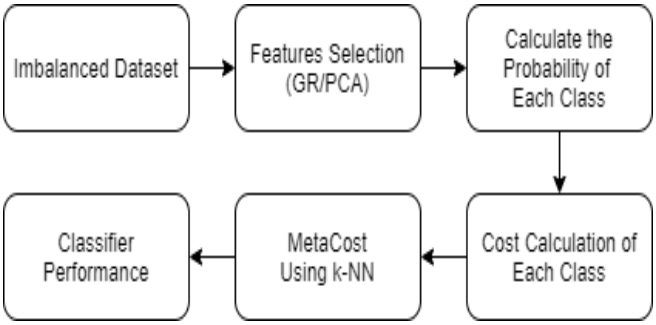

Figure 1. Proposed Method

Based on Figure 1, it can be seen that the proposed method will use Gain Ratio (GR) or Principal Component Analysis (PCA) feature selection to select features that are relevant to the data class. Furthermore, the calculation of the probability of data prediction for each class is carried out. Based on these probabilities, a cost matrix is calculated for each class and it is recommended to classify using MetaCost.

IV. RESULT AND DISCUSSION

In this study, several scenarios will be carried out. The first scenario will classify using a classifier without doing feature selection. the second scenario will classify with a classifier after feature selection. The dataset used is the red wine quality and e-coli dataset, for details on the dataset, see Table 1.

TABLE 1
DETAILS DATASET

No	*Dataset*	*Instances*	*Number of Classes*	*Class Distribution*
1	Red Wine Quality	1599	6	681/638/199/53/18/10
2	E-Coli	336	8	143/77/2/2/35/20/5/52

The cost for each class in the red wine quality data can be seen in Table 2.

TABLE 2
COST MATRIX FOR WINE QUALITY

Class	*3*	*4*	*5*	*6*	*7*	*8*
Cost	0.508	0.475	0.475	0.465	0.294	0.413

During feature selection, the 5 most dominant features will be selected based on the gain ratio and component model values. Performance for each scenario can be seen in Table 3.

TABLE 3
PERFOMANCE FOR RED WINE QUALITY

	Accuracy	*Precision*	*Recall*	*F1-Score*
K-NN	0.6692	0.4526	0.3338	0.3842
MetaCost	0.6829	**0.5691**	0.4141	**0.4794**
K-NN+GR	0.7117	0.5432	0.3872	0.4521
MetaCost+GR	**0.7361**	0.5449	**0.4143**	0.4707
K-NN+PCA	0.6629	0.4450	0.3301	0.3790
MetaCost+PCA	0.7148	0.5636	0.3916	0.4621

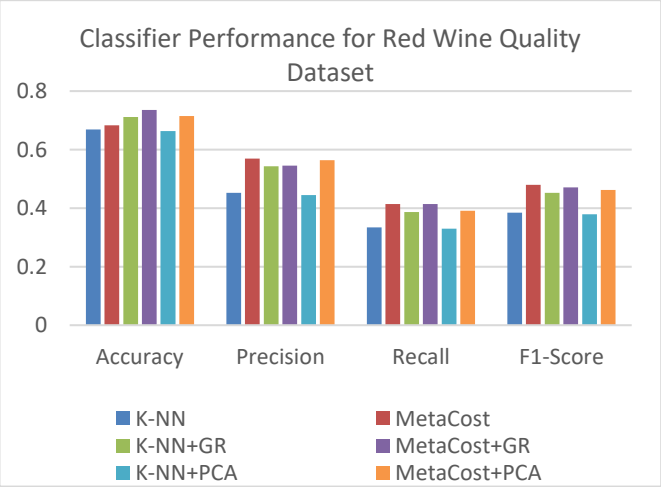

Figure 2. Classifier Performance for Red Wine Quality

Based on Table 3, it can be seen that feature selection with GR can provide a better performance, while feature selection with PCA makes K-NN performance decrease. While the specified cost is proven to be able to improve the performance of MetaCost. performance is seen by using MetaCost where the data is first selected for fit using GR, where the parameter increase occurs in the value of increasing accuracy and recall.

To obtain more convincing results, a second test will be carried out with the e-coli data set. The cost for each class in the e-coli data can be seen in Table 4, while the performance of each scenario can be seen in Table 5.

TABLE 4
COST MATRIX FOR E-COLI

Class	***cp***	***im***	***imS***	***imL***
Cost	0.2218	0.127	0.1	0
Class	***imU***	***om***	***omL***	***pp***
Cost	0.2898	0.1056	0.1085	0.2879

TABLE 5
PERFOMANCE FOR E-COLI

	Accuracy	*Precision*	*Recall*	*F1-Score*
K-NN	0.9018	0.6612	0.6423	0.6516
MetaCost	0.9018	0.6624	0.6648	0.6636
K-NN+GR	0.9048	0.6650	0.6673	0.6661
MetaCost+GR	**0.9167**	**0.9081**	**0.8303**	**0.8674**
K-NN+PCA	0.9018	0.6624	0.6648	0.6636
MetaCost+PCA	**0.9167**	**0.9081**	**0.8303**	**0.8674**

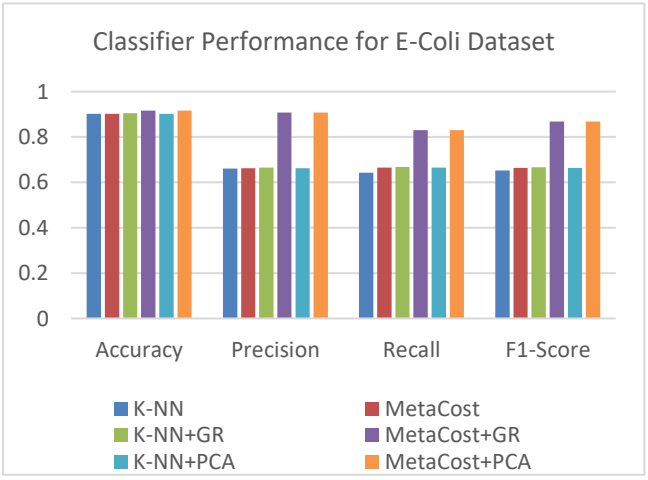

Figure 3. Classifier Performance for E-Coli

Based on Table 5, it can be seen that feature selection with GR provides better performance, while the specified cost is also proven to be able to improve the performance of MetaCost. The performance improvement is visible when using the MetaCost method where feature selection is carried out first, either with GR or with PCA. The highest increase parameters occur in the accuracy, precision, recall, and F1-Score values.

The highest performance was found in e-coli data, where the MetaCost method which has been feature selected was able to increase the average performance by 0.0142 for accuracy, 0.2453 for precision, 0.1705 for recall, and 0.2062 for F1-Score.

The results of this test based on Tables 3 and 5 it was found that the feature selection method was able to improve the by reducing irrelevant features. This test also proves that the MetaCost method has better performance on imbalanced data.

V. CONCLUSION

Feature selection is proven to be able to improve performance, especially feature selection using the Gain Ratio method. The MetaCost method is also proven to be able to provide better performance. From testing the two datasets, it can be seen that the combination of MetaCost and GR has better performance, where from the two datasets there is an increase of 0.0295 in the Accuracy value, 0.1132 in the Precision value, 0.0897 in the Recall, and 0.1021 in the F1-Score. For future work, we will focus on determining cost-matrix on cost-sensitive learning to improve classifier performance on unbalanced multiclass data.

REFERENCES

[1] H. Ali, M. N. M. Salleh, R. Saedudin, K. Hussain, and M. F. Mushtaq, "Imbalance class problems in data mining: A review," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 14, no. 3, pp. 1552–1563, 2019, doi:

10.11591/ijeecs.v14.i3.pp1552-1563.
[2] N. V Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "SMOTEBoost : Improving Prediction," *Eur. Conf. Princ. Data Min. Knowl. Discov.*, vol. 2838, pp. 107–119, 2003, [Online]. Available: https://link.springer.com/content/pdf/10.1007%2F978-3-540-39804-2_12.pdf.
[3] S. Wang and X. Yao, "Multiclass imbalance problems: Analysis and potential solutions," *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, vol. 42, no. 4, pp. 1119–1130, 2012, doi: 10.1109/TSMCB.2012.2187280.
[4] N. Santoso, W. Wibowo, and H. Himawati, "Integration of synthetic minority oversampling technique for imbalanced class," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 13, no. 1, pp. 102–108, 2019, doi: 10.11591/ijeecs.v13.i1.pp102-108.
[5] R. Alizadehsani, M. J. Hosseini, Z. A. Sani, A. Ghandeharioun, and R. Boghrati, "Diagnosis of coronary artery disease using cost-sensitive algorithms," *Proc. - 12th IEEE Int. Conf. Data Min. Work. ICDMW 2012*, pp. 9–16, 2012, doi: 10.1109/ICDMW.2012.29.
[6] A. Motwani, P. Chaurasiya, and G. Bajaj, "Predicting Credit Worthiness of Bank Customer with Machine Learning over Cloud," *Int. J. Comput. Sci. Eng.*, vol. 6, no. 7, pp. 1471–1477, 2018, doi: 10.26438/ijcse/v6i7.14711477.
[7] C. Wang, R. Li, P. Wang, and Z. Chen, "Partition cost-sensitive CART based on customer value for Telecom customer churn prediction," *Chinese Control Conf. CCC*, no. July, pp. 5680–5684, 2017, doi: 10.23919/ChiCC.2017.8028259.
[8] S. Barua, M. M. Islam, X. Yao, and K. Murase, "MWMOTE - Majority weighted minority oversampling technique for imbalanced data set learning," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 2, pp. 405–425, 2014, doi: 10.1109/TKDE.2012.232.
[9] Q. Gu, X. M. Wang, Z. Wu, B. Ning, and C. S. Xin, "An improved SMOTE algorithm based on genetic algorithm for imbalanced data classification," *J. Digit. Inf. Manag.*, vol. 14, no. 2, pp. 92–103, 2016.
[10] S. Zhang, "Cost-sensitive KNN classification," *Neurocomputing*, vol. 391, no. xxxx, pp. 234–242, 2020, doi: 10.1016/j.neucom.2018.11.101.
[11] Z. Qin, A. T. Wang, C. Zhang, and S. Zhang, "Cost-Sensitive Classification with k-Nearest Neighbors," pp. 112–131, 2013.
[12] A. N. Haldankar and K. Bhowmick, "A cost sensitive classifier for Big Data," *2016 IEEE Int. Conf. Adv. Electron. Commun. Comput. Technol. ICAECCT 2016*, pp. 122–127, 2017, doi: 10.1109/ICAECCT.2016.7942567.
[13] A. A. Nababan, O. S. Sitompul, and Tulus, "Attribute Weighting Based K-Nearest Neighbor Using Gain Ratio", *J. Phys. Conf. Ser.*, vol. 1007, no. 1, 2018, doi: 10.1088/1742-6596/1007/1/012007.
[14] N. Qazi and K. Raza, "Effect of feature selection, Synthetic Minority Over-sampling (SMOTE) and under-sampling on class imbalance classification," *Proc. - 2012 14th Int. Conf. Model. Simulation, UKSim 2012*, no. May 2014, pp. 145–150, 2012, doi: 10.1109/UKSim.2012.116.
[15] M. Z. F. Nasution, O. S. Sitompul, and M. Ramli, "PCA based feature reduction to improve the accuracy of decision tree c4.5 classification," *J. Phys. Conf. Ser.*, vol. 978, no. 1, 2018, doi: 10.1088/1742-6596/978/1/012058.
[16] Kotu V and Deshpande B, *Predictive Analytics and Data Mining*. Waltham: Morgan Kaufmann, 2015.
[17] R. Kavitha and E. Kannan, "An efficient framework for heart disease classification using feature extraction and feature selection technique in data mining," *1st Int. Conf. Emerg. Trends Eng. Technol. Sci. ICETETS 2016 - Proc.*, 2016, doi: 10.1109/ICETETS.2016.7603000.
[18] I. T. Jolliffe, "Principal Component Analysis, Second Edition," *Encycl. Stat. Behav. Sci.*, vol. 30, no. 3, p. 487, 2002, doi: 10.2307/1270093.
[19] J. F. Jr. Hair, W. C. Black, B. J. Babin, and R. E. Anderson, *Pearson New International Edition*. 2014.
[20] X. Wu *et al.*, *Top 10 algorithms in data mining*, vol. 14, no. 1. 2008.
[21] K. U. Syaliman, E. B. Nababan, and O. S. Sitompul, "Improving the accuracy of k-nearest neighbor using local mean based and distance weight", *J. Phys. Conf. Ser.*, vol. 978, no. 1, pp. 1–6, 2018, doi: 10.1088/1742-6596/978/1/012047.