# Accuracy Enhancement of Prediction Method using SMOTE for Early Prediction Student's Graduation in XYZ University

Ainul Yaqin*, Majid Rahardi, Ferian Fauzi Abdulloh
Computer Science Faculty, University of AMIKOM Yogyakarta, Yogyakarta, Indonesia

***Abstract*—According to the Minister of Education and Culture of the Republic of Indonesia's regulations from 2014, one of the essential elements in implementing higher education is the student's study duration. Higher education institutions will use early graduation prediction as a guide when developing policy. According to XYZ University data, the student study period is Grade Point Average (GPA), Gender, and Age are all aspects to consider. Using a dataset of 8491 data, the Prediction of Early Graduation of Students based on XYZ University data was examined by this study, particularly in the information systems and informatics study program. The aim is to find significant features and compare three prediction models: Artificial Neural Networks (ANN), K-Nearest Neighbor (K-NN) method, and Support Vector Machines (SVM). The Challenge in the development of a prediction model is imbalanced data. The Synthetic Minority Oversampling Technique (SMOTE) handles the class imbalance problem. Next, the machine learning models are trained and then compared. Prediction results increase. The best test accuracy value is on ANN with a data Imbalance of 62.5% to 70.5% after using SMOTE, compared to the accuracy test on the K-NN method with SMOTE 69.3%, while the SVM method increased to 69.8%. The most significant increase in recall value to 71.3% occurred in the ANN.**

***Keywords*—Prediction study period; SMOTE; neural network; k-nearest neighbors; support vector machine**

## I. Introduction

### A. Background

One of the criteria that can be used to determine a student's performance in higher education is the student's study duration. The information about the study session is recorded in the academic database in this case, and management can utilize it to assess decision-making. The Informatics and Information Systems Study Program at XYZ University is an example of a study program with the maximum number of active students and continues to increase every year. Management can identify early prevention strategies connected to Drop Out (DO) instances by using the prediction model for the student study period, which will have a domino effect on accreditation.

Undergraduate students must take at least 144-semester credits with a study period of four years or less to meet learning outcomes, as expressly stated in the Regulation of the Minister of Education and Culture of the Republic of Indonesia in 2014, a section on graduate competency standards article 17 [1]. Consequently, the study program manager is required to keep track of student progress to reduce the number of dropouts. The development of prediction models for students' study periods using machine learning models has been done using several parameters Gender[2], GPA [3][4], Number of semester credit units (credits), parents' job, high school major, high school city[2][5]. In this study, a prediction model for the study period was constructed utilizing semester 1 to 4 GPA values based on XYZ University's educational program, which highlights concentration selection in semester four lectures, ages, and Gender.

The K-Nearest Neighbor technique (K-NN), Artificial Neural Networks (ANN), and Support Vector Machines are some examples of popular machine learning methods for Prediction (SVM) [6]. The Challenge in the development of a prediction model is imbalanced data. This situation prompted us to use the Synthetic Minority Oversampling Technique (SMOTE) to improve prediction utilizing balance data [7] to handle the class imbalance problem. University XYZ datasets are used in an experimental study to validate the suggested approach, including selecting a prediction model and performance indicators. In the following areas, this study adds to the existing body of knowledge:

*1)* ANN, K-NN, and SVM were used to create an effective model.

*2)* To analyze the relevant features extracted from the dataset that has an impact on the machine learning algorithm's performance.

*3)* Performance of ANN, K-NN, and SVM models using Imbalance data is compared using SMOTE.

Thirty percent of machine learning methods used in predicting student performance are artificial neural networks and vector machines at 6,20%. Other methods used are decision tree, nave Bayes, and k-Nearest Neighbor 17% each [6].

### B. Related Work

Agustin [8] used class year, GPA, and grades from nine courses as parameters with the Naive Bayes Classification Technique, K-Nearest Neighbor, and Neural Network to predict student study period. His research divided predictions into three classes: fast, precise, and slow. And using the most influential attribute selection, namely, GPA, using the classifier attribute.

*Corresponding Author.

Prediction of students' graduation time using the K-Nearest Neighbor Algorithm with the parameter GPA of students during the seventh semester will be used as training data. Data of students who pass are used as sample data. K-Nearest Neighbor works according to the given data sample. The highest level of Accuracy can be achieved when k = 3 [9].

The fundamental purpose of SVM is to separate the data set into classifications to find the most negligible Hyperplane. The gamma value used varies from 0 to 1, but a good default gamma value is 0.1 and can be achieved Accuracy of 73% [10].

The rest of the paper is organized as follows: Section II outlines the proposed approach, the dataset, preprocessing, and data visualization to uncover the dataset's hidden pattern. It also explains the various algorithms employed in this study. The discussion and analysis of the results, performance metrics, and experimental study design are all described in Section III. In Section IV, the findings are discussed. Section V concludes with a discussion of future work.

## II. Research Methodology

This research will use a method divided into three stages: preprocessing, predictive Process, and evaluation. The research flow is shown in Fig. 1.

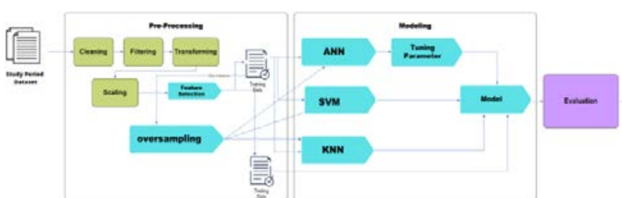


Fig. 1. Research Methodology.

### A. Synthetic Minority Oversampling Technique (SMOTE)

Several approaches to dealing with data imbalances include *undersampling*, *oversampling*, and *hybrid* methods [11]. Oversampling technique is often adopted to overcome the problem by generating new synthetic sample data [12][7][13]. In comparison, the *Undersampling technique* has an approach of removing data from the majority class to balance the class distribution [14].

### B. Artificial Neural Networks (ANN)

A Neural Network (ANN) is a group of small processing units based on a human neural network in general [15]. The learning system is a never-ending process of adding to the NN's knowledge (continuity). When used to recognize an object, the inside will be fully utilized. The anatomy of neural networks [16] is shown in Fig. 2.

The following data is provided by the ANN Structure shown in Fig. 2. The "Input Layer" is made up of input nodes that transmit data from the outside world to the network. A "hidden layer" is formed when a group of hidden nodes is combined [17]. The hidden layer nodes of the ANN have an activation function that allows for a curvilinear fit between the input and output units (layers). The activation function you choose has a big impact on how well your network works. [18].

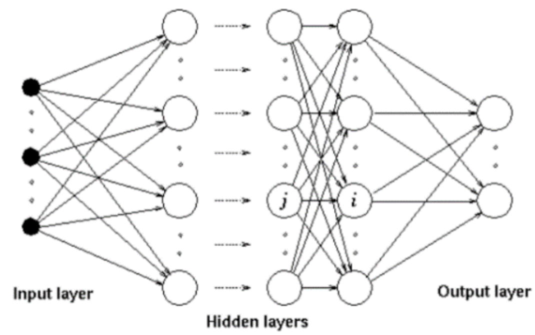


Fig. 2. Arsitektur ANN.

### C. K-Nearest Neighbor method (K-NN)

K-NN is a statistical classification technique that uses the closest training samples in Feature Space to classify objects. It's a lazy learning algorithm in which the K-NN function is approximated locally and all calculations are postponed until classification is completed [19]. The K-NN algorithm generally uses the Euclidean distance formula, such as the formula 1:

$$D_{xy}\sqrt{\sum_{i=1}^{n} \quad (X_i - y_i)^2} \tag{1}$$

The transition from distances to weights should follow some kernel functions. Typical examples of this function include rectangular kernel, triangular kernel, cosine kernel, Gauss kernel, and inversion kernel [20].

### D. Support Vector Machines (SVM)

Support Vector Machines (SVM) is a supervised learning algorithm that uses a learning algorithm that analyzes data related to classification and regression analysis [21]. In SVM modeling, there are several different kernels, including Radial Base (RBF), linear (LIN), sigmoid (SIG), and polynomial (POLM melakrefrefPOL) [22]. SVM reformats non-linear into linear by generating a Hyperplane and converting it to a straightforward form that can be processed [23].

The support vector is the closest training point to the Hyperplane, so maximizing the margin is considered a rule by minimizing the value of *w* [22]. The Decision Function Formula of SVM as in formula 2 and the SVM Scheme is shown in Fig. 3.

$$\mathrm{f}(x) = sign\ (w.x + b) \tag{2}$$

### E. Evaluation

At this stage, an Evaluation or Model Evaluation is carried out on each model based on the accuracy score using the *Confusion Matrix* to determine the Accuracy of Prediction, *recall*, *precision*, and *F-Measure* [24].

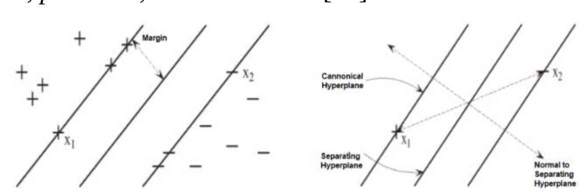


Fig. 3. Skema SVM.

## III. Experimental Study

To test the proposed technique, this experiment chooses a diverse range of prediction algorithms, datasets, and performance metrics. Data sources, classification methods, performance measurements, and experimental design are all discussed in this subsection.

### A. Data Source

The data was obtained by using query study data for students of the informatics and information systems study program in the class of 2004 to 2015 from the XYZ University database with a total data of 10,274. The dataset obtained consists of NIM, Gender, SMA, Date of Birth, Date of graduation, and GPA 1 to GPA 4. Summary dataset is shown in Table I.

TABLE I. Summary Dataset

| COLUMN | TYPE | DATA NULL/ISNA() |
|---|---|---|
| NIM | object | 0 |
| GENDER | object | 0 |
| Senior High School | object | 569 |
| Birth Date | datetime64[ns] | 38 |
| Graduation Date | datetime64[ns] | 458 |
| GPA 1 | float64 | 8 |
| GPA 2 | float64 | 258 |
| GPA 3 | float64 | 406 |
| GPA 4 | float64 | 766 |

### B. Prediction Model

This section discusses the algorithms used in this study. The prediction Model use ANN, K-NN, and SVM algorithm. The feature selection phase used the ONE-HOT algorithm, and an enhanced version of SMOTE was used in the rebalancing phase.

The first architectural model will be built with the ANN Algorithm in this study uses 1 hidden layer [25] with 1 to 100 neurons and a learning rate of 0.1 to 0.5, and a maximum iteration of 1000. The second model uses the K-NN algorithm by comparing and testing various K values from 2 to 1000. The third model uses SVM Algorithm by choosing the kernel between RBF and Linear [26].

### C. Performance Measures (Evaluation)

Accuracy, true positive rate (Recall), true negative rate, precision, G-Mean, F-measure, and computation time are all popularly used performance indicators in classification. The following are the definitions for these metrics.

*1)* True Positive (TP): TP is the number of instances that are correctly positive.

*2)* True Negative (TN): TN stands for the number of instances that are correctly negative.

*3)* False Positive (FP): The number of positive events that have been misclassified as negative.

*4)* False Negative (FN): the number of negative instances that have been misclassified as positive.

*5)* TP rate (TPR) is also called sensitivity measure or Recall like formula 3.

$$sensitivity = TP/(TP + FN) \quad (3)$$

*6)* TN rate is called specificity measure like the formula 4.

$$specificity = TN/(TN + FP) \quad (4)$$

*7)* FP rate is called Type-I error like the formula 5.

$$Type - I\ error = FP/(FP + TN) \quad (5)$$

*8)* Type-II error is another term for the FN rate. Like the formula 6

$$Type - II\ error = FN/(FN + TP) \quad (6)$$

*9)* Overall Accuracy: The percentage of accurately anticipated instances, such as formula 7, is referred to as accuracy.

$$Accuracy = (TN + TP)/(TP + FP + TN + FN) \quad (7)$$

*10)*Precision: This is the Number of fault-prone modules that are fault-prone modules like formula 8.

$$Precision = TP/(FP + TP) \quad (8)$$

*11)*G-Mean: it is the geometric mean of sensitivity and precision like the formula 9.

$$G\ Mean = \sqrt{sensitivity * Precision} \quad (9)$$

*12)*The harmonic mean of precision and recall is known as the F-Measure. The F-measure, like the formula 10, has been frequently utilized in information retrieval.

$$F\ Measure = (2 * Precision * TPR)/(Precision + TPR) \quad (10)$$

### D. Experimental Design

The following procedure was used to carry out the experiment:

**Input**: Dataset University XYZ
**Tool:** Jupyter - Python Base
**Step 1**: Pre-Processing Data
In this Phase, several processes are carried out, namely:

1. The Process of cleaning null data from 10,274 data to 8491 data
2. The filtering process is carried out by:
   a. The value of the study period is obtained through the difference between the year of study and the date of graduation by labeling the study period on time if the study period is <=48 months with a label 1 and late study with a label 0 if the study period > 48 months
   b. Get the student's age (years) during semester 4 by calculating the difference between the start date of college and the date of birth plus 2 years.
3. The transformation process is carried out by:
   a. Categorize data from school and age
   b. Combining age group data with Gender

4. The scaling process is carried out on Semester 1 GPA data into Semester 4 GPA data using MinMax Scaler(MMS)[27] like formula 11 [28]:

$$MMS = (X - X_{min})/(X_{min} - X_{max}) \quad (11)$$

**Step 2**: Correlation Analysis

Correlation analysis was used to determine the degree of relationship between GPA 1, GPA 2, GPA 3, GPA 4, Age, Gender, Year of Class, and Senior High School during the study period.

**Step 3**: Features Selection

In this stage, feature determination is carried out, which will be used in the prediction model

**Step 4**: Split Data Train and Testing and Balancing using SMOTE

In this Step, Data distribution is divided into 80% training data and 20% testing data. Then, the training data is balanced with SMOTE and then stored in other variables

**Step 5**: Training Prediction Model

In this Phase, the prediction model was built with ANN, K-NN, and SVM trained using the Imbalanced training data and SMOTE training data. So it produces six models.

**Step 6**: Evaluation

In this Phase, evaluate the results of testing data as much as 20% of the dataset on six models (in the model training phase) such as Accuracy, recall, precision, G-Means, and F-Measure.

## IV. Results

### A. Dataset

The results of the preprocessing of the dataset are shown in Table II.

### B. Correlation Analysis

Fig. 4 shows that the GPA 1 to 4 has a correlation value of >0.3 with the Study Period. Then the Gender-Age value data with Study Period also indicates a negative relationship with a correlation value of -0.13

### C. Features Selection

At this stage, the features are determined by setting a GPA 1 to 4, and Gender-Age is the feature and the Study Period is the target. Then Categorical Features Encoding is carried out for Gender-Age data. So that the resulting features selection data includes as many as 12 features. The result of feature selection is shown in Table III.

### D. Split Data

In this Phase, the data was obtained from 6792 training data and 1699 testing data. The summary of training data is shown in Fig. 5. Training Balance data is shown in Fig. 6.

TABLE II. DATASET

| Num | Nim | Lblgender | Lblschool | GPA1 | GPA2 | GPA3 | GPA4 | YEAR | LblAge | LblGender-Age | Study Period |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 13.11 | 0 | 2 | 3.5 | 3.833333 | 4 | 3.5 | 2013 | 19-23 | 7 | 1 |
| 1 | 13.11 | 0 | 2 | 3.5 | 3.75 | 3.333333 | 3.25 | 2013 | 19-23 | 7 | 1 |
| 2 | 13.11 | 0 | 1 | 3.166667 | 3.083333 | 3.333333 | 3.25 | 2013 | 19-23 | 7 | 1 |
| 3 | 13.11 | 1 | 0 | 3.5 | 3.25 | 3.5 | 3.416667 | 2013 | 19-23 | 6 | 1 |
| 4 | 13.11 | 0 | 1 | 3.25 | 3.166667 | 3.25 | 3.166667 | 2013 | 19-23 | 7 | 1 |
| … | … | … | … | … | … | … | … | … | … | … | … |
| 8488 | 04.12. | 1 | 1 | 2.5 | 2.923077 | 3 | 2.923077 | 2004 | 19-23 | 6 | 1 |
| 8489 | 04.12. | 0 | 0 | 2.833333 | 2.923077 | 3 | 2.615385 | 2004 | 19-23 | 7 | 1 |
| 8490 | 04.12. | 0 | 1 | 3.5 | 3.285714 | 3.307692 | 3.153846 | 2004 | 19-23 | 7 | 1 |
| 8491 | 04.12. | 0 | 1 | 3.5 | 3.357143 | 3.538462 | 2.833333 | 2004 | 19-23 | 7 | 1 |

TABLE III. RESULT OF FEATURE SELECTION

| Num | GPA1 | GPA2 | GPA3 | GPA4 | L_19-23 | L_23-27 | L_<19 | L_>27 | P_19-23 | P_23-27 | P_<19 | P_>27 | STUDY PERIOD |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0.85 | 0.9575163 | 1 | 0.87209 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0.85 | 0.9362745 | 0.8300 | 0.80813 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 2 | 0.75 | 0.7663399 | 0.8300 | 0.80813 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 3 | 0.85 | 0.8088235 | 0.8725 | 0.85077 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 4 | 0.775 | 0.7875817 | 0.8088 | 0.78682 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| … | … | … | … | … | … | … | … | … | … | … | … | … | … |
| 8488 | 0.55 | 0.7254902 | 0.7450 | 0.72450 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 8489 | 0.65 | 0.7254902 | 0.7450 | 0.64579 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 8490 | 0.85 | 0.8179272 | 0.8235 | 0.78354 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 8491 | 0.85 | 0.8361345 | 0.8823 | 0.70155 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

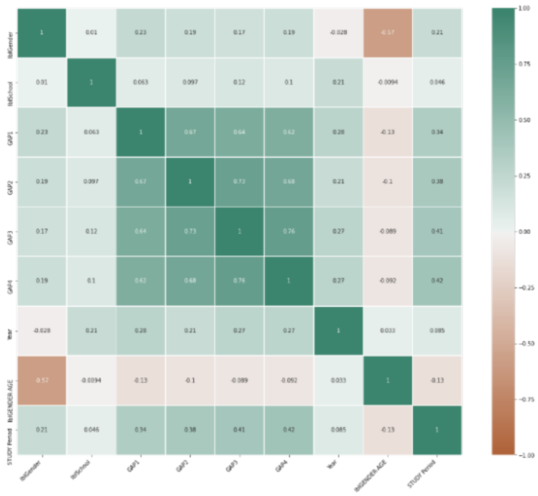

Fig. 4. Correlation Analysis.

```
X_train, X_test, y_train, y_test = train_test_split(dtfix[features], dtfix[target], test_size=0.20, random_state=42)
print("Training data", X_train.shape);
# print(y_train.shape)
print("Testing data", X_test.shape);
# print(y_test.shape);
```

```
Training data (6792, 12)
Testing data (1699, 12)
```

Fig. 5. Split Data.

```
y_train.STATUS.value_counts()
```

```
0    3792
1    3000
Name: STATUS, dtype: int64
```

Fig. 6. Imbalance Data.

Then, the training data is balanced with SMOTE. It is shown in Fig. 7.

```
print("Imbalance Training data : ",y_train.value_counts())
print("Balance Data using SMOTE:",y_train_smote.value_counts())
```

```
Imbalance Training data :  STATUS
0         3792
1         3000
dtype: int64
Balance Data using SMOTE: STATUS
0         3792
1         3792
dtype: int64
```

Fig. 7. SMOTE.

### E. *Results of Artificial Neural Network Training Model*

Network modeling is done by finding the best model with the GridSerachCV function in sklearn. Modeling is based on several parameters of the activation function, namely, tanh, relu, and logistics, with the Number of neurons in the hidden layer ranging from 1 to 100 neurons, a learning rate of 0.1 to 0.5, and a maximum iteration of 1000. The best model of ANN obtained activation function = "tanh", learning rate = 0.1 , alpha = 0.1 and neuron = 99 in the hidden layer. The result of the ANN Training Model is shown in Table IV.

### F. *Results of K-Nearest Neighbors (K-NN) Training Model*

K-NN modeling is carried out with uniform and distance and testing various K values from 2 to 1000. The Result of the Highest Accuracy is shown in Fig. 8 and Table V.

### G. *Results of Support Vector Machine (SVM) Training Model*

SVM modeling is done by choosing a kernel between RBF and linear, with the results shown in Fig. 9.

TABLE IV. RESULT OF ANN TRAINING MODEL

| Training Data | Study Period (Target) | | ANN training accuracy | ANN testing accuracy |
|---|---|---|---|---|
| | 0 | 1 | | |
| Imbalance | 3792 | 3000 | 0.7005 | 0.6250 |
| SMOTE | 3792 | 3792 | 0.6969 | 0.7080 |

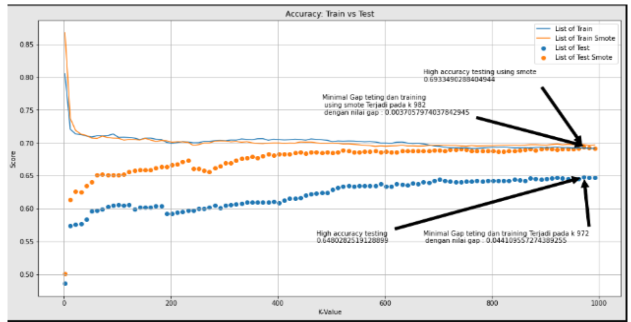


Fig. 8. Result of Highest K-NN Accuracy.

TABLE V. RESULT OF THE K-NN TRAINING MODEL

| Training Data | Study Period | | K-NN training accuracy | K-NN testing accuracy |
|---|---|---|---|---|
| | 0 | 1 | | |
| Imbalance | 3792 | 3000 | 0.6921 | 0.6972 |
| SMOTE Balance | 3792 | 3792 | 0.6480 | 0.6933 |

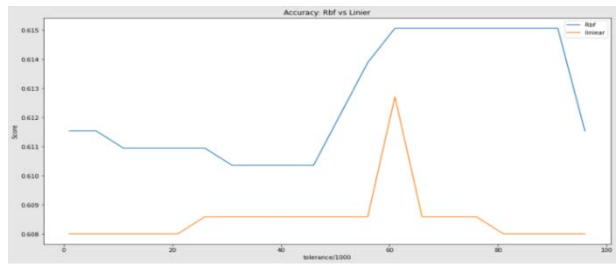


Fig. 9. Result of Comparison SVM Kernel.

The results of training and testing the SVM model with the RBF kernel using Balance data and SMOTE Balance data are shown in Fig. 10 and Table VI.

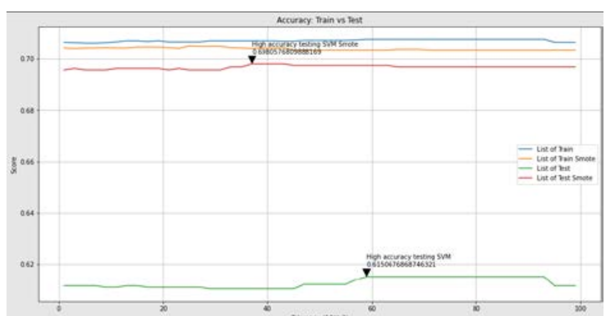


Fig. 10. Result of Highest Accuracy of SVM.

TABLE VI. RESULT OF THE SVM TRAINING MODEL

| Training Data | Study Period (Target) | | SVM training accuracy | SVM testing accuracy |
|---|---|---|---|---|
| | 0 | 1 | | |
| Imbalance | 3792 | 3000 | 0.7077 | 0.6150 |
| SMOTE | 3792 | 3792 | 0.7051 | 0.6980 |

TABLE VII. EVALUATION OF PERFORMANCE PREDICTION MODEL

| | Imbalance-ANN | SMOTE-ANN | Imbalance-K-NN | SMOTE-K-NN | Imbalance-SVM | SMOTE-SVM |
|---|---|---|---|---|---|---|
| **Training Accuracy** | 0.700530035 | 0.696993671 | 0.692137809 | 0.697257384 | 0.707744405 | 0.705168776 |
| **Testing Accuracy** | 0.625073573 | 0.708063567 | 0.648028252 | 0.693349029 | 0.615067687 | 0.698057681 |
| **TP** | 591 | 805 | 661 | 768 | 558 | 676 |
| **FP** | 471 | 398 | 440 | 410 | 483 | 452 |
| **FN** | 100 | 173 | 131 | 161 | 88 | 119 |
| **TN** | 537 | 323 | 467 | 360 | 570 | 452 |
| **Recall** | 0.52393617 | 0.713652482 | 0.585992908 | 0.680851064 | 0.494680851 | 0.59929078 |
| **Precision** | 0.8552822 | 0.823108384 | 0.83459596 | 0.826695371 | 0.86377709 | 0.850314465 |
| **G-Mean** | 0.657402861 | 0.705288613 | 0.671977086 | 0.699197609 | 0.646871565 | 0.688763242 |
| **F-Measure** | 0.649807587 | 0.764482431 | 0.688541667 | 0.746718522 | 0.629086809 | 0.703068123 |

### H. Evaluation

The evaluation phase, this study uses data testing as much as 1699 data on six models that have been trained. The results can be seen in Table VII.

Based on Table VII, the results of conformity testing are performed using a confusion matrix, and the test results are displayed in graphical form as follows in Fig. 11.

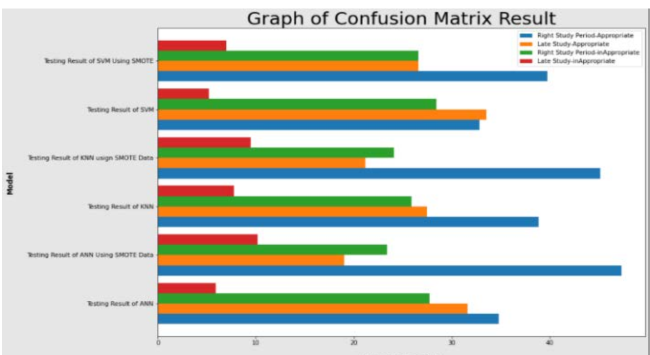


Fig. 11. Confusion Matrix Test Result.

## V. CONCLUSION

Based on the results, this study can be formulated as follows. SMOTE can increase the recall value and test the accuracy of ANN, K-NN, and SVM models with relevant features extracted, namely GPA 1 to 4, and Encoding of Gender-Age based on correlation values. Prediction results have the best increase in the value of accuracy testing on ANN with a data Imbalance of 62.5% to 70.5% after using SMOTE, compared to the accuracy test on the K-NN method with SMOTE 69.3%. In comparison, the SVM method increased to 69.8%. The most significant increase in recall value to 71.3% occurred in the ANN.

## ACKNOWLEDGMENT

This research has been fully supported by grants from the University of AMIKOM (#RP-1615185701000).

## REFERENCES

[1] Kementerian Pendidikan dan Kebudayaan RI, “Peraturan Menteri Standar Nasional Pendidikan Tinggi,” Ban-PT, 2015. [Online]. Available: https://www.banpt.or.id/?page_id=80.

[2] A. Maulana, "Prediction of student graduation accuracy using decision tree with application of genetic algorithms," IOP Conf. Ser. Mater. Sci. Eng., vol. 1073, no. 1, p. 012055, Feb. 2021, DOI: 10.1088/1757-899X/1073/1/012055.

[3] A. Yaqin, A. D. Laksito, and S. Fatonah, "Evaluation of Backpropagation Neural Network Models for Early Prediction of Student's Graduation in XYZ University," Int. J. Adv. Sci. Eng. Inf. Technol., vol. 11, no. 2, pp. 610–617, 2021, DOI: 10.18517/IJASEIT.11.2.11152.

[4] M. Anwar, "Prediction of the graduation rate of engineering education students using Artificial Neural Network Algorithms," Int. J. Res. Couns. Educ., vol. 5, no. 1, pp. 23–31, Jun. 2021, DOI: 10.24036/00411ZA0002.

[5] M. Wati, W. H. Rahmah, N. Novirasari, Haviluddin, E. Budiman, and Islamiyah, "Analysis K-Means Clustering to Predicting Student Graduation," J. Phys. Conf. Ser., vol. 1844, no. 1, p. 012028, Mar. 2021, DOI: 10.1088/1742-6596/1844/1/012028.

[6] Y. Baashar, G. Alkawsi, N. Ali, H. Alhussian, and H. T. Bahbouh, "Predicting student's performance using machine learning methods: A systematic literature review," 2021 Int. Conf. Comput. Inf. Sci., pp. 357–362, Jul. 2021, DOI: 10.1109/ICCOINS49721.2021.9497185.

[7] D. Gonzalez-Cuautle et al., "Synthetic Minority Oversampling Technique for Optimizing Classification Tasks in Botnet and Intrusion-Detection-System Datasets," DOI: 10.3390/app10030794.

[8] D. Agustin, D. Zaenal Abidin, and E. Rasywir, “Kajian Komparasi Teknik Klasifikasi Naive Bayes, K-Nearest Neighbor, Dan Jaringan Saraf Tiruan Pada Prediksi Lama Masa Studi Mahasiswa Jurusan Teknik Informatika (Studi Kasus : Stikom Dinamika Bangsa Jambi).”

[9] R. Muliono, J. H. Lubis, and N. Khairina, "Analysis K-Nearest Neighbor Algorithm for Improving Prediction Student Graduation Time," Sink. J. dan Penelit. Tek. Inform., vol. 4, no. 2, pp. 42–46, Mar. 2020, DOI: 10.33395/SINKRON.V4I2.10480.

[10] I. Rodrigues, A. Parayil, T. Shetty, and I. Mirza, "Use of Linear Discriminant Analysis (LDA), K Nearest Neighbours (K-NN), Decision Tree (CART), Random Forest (RF), Gaussian Naive Bayes (NB), Support Vector Machines (SVM) to Predict Admission for Post Graduation Courses," SSRN Electron. J., 2020, DOI: 10.2139/ssrn.3683065.

[11] M. Koziarski, "CSMOUTE: Combined Synthetic Oversampling and Undersampling Technique for Imbalanced Data Classification."

[12] S. Feng et al., "COSTE: Complexity-based OverSampling TEchnique to alleviate the class imbalance problem in software defect prediction," Inf. Softw. Technol., vol. 129, pp. 950–5849, Jan. 2021, DOI: 10.1016/j.infsof.2020.106432.

[13] F. Kamalov and D. Denisov, "Gamma distribution-based sampling for imbalanced data," Knowledge-Based Syst., vol. 207, p. 106368, Nov. 2020, DOI: 10.1016/j.knosys.2020.106368.

[14] T. Liu, X. Zhu, W. Pedrycz, and Z. Li, "A design of information granule-based under-sampling method in imbalanced data classification," Soft Comput., vol. 24, no. 22, pp. 17333–17347, 2020, DOI: 10.1007/s00500-020-05023-2.

[15] S.-C. Wang, "Artificial Neural Network," in Interdisciplinary Computing in Java Programming, Boston, MA: Springer US, 2003, pp. 81–100.

[16] B. S. Santoso, J. P. Tanjung, U. P. Indonesia, B. Gandum, and A. N. Network, "Classification of Wheat Seeds Using Neural Network Backpropagation," JITE ( J. Informatics Telecommun. Eng. ), vol. 4, no. January 2021.

[17] O. Ahmed and A. Brifcani, "Gene Expression Classification Based on Deep Learning," 4th Sci. Int. Conf. Najaf, SICN 2019, pp. 145–149, 2019, doi: 10.1109/SICN47020.2019.9019357.

[18] M. A. Mercioni and S. Holban, "Soft-Clipping Swish: A Novel Activation Function for Deep Learning," pp. 225–230, Jun. 2021, DOI: 10.1109/SACI51354.2021.9465622.

[19] R. Harikumar and K. P. Sunil, K-NN Classifier and K-Means Clustering for Robust Classification of Epilepsy ... - Harikumar Rajaguru, Sunil Kumar Prabhakar - Google Books. Hamburg: Anchor Academic Publishing, 2017.

[20] L. Cui, Y. Zhang, R. Zhang, and Q. H. Liu, "A Modified Efficient K-NN Method for Antenna Optimization and Design," IEEE Trans. Antennas Propag., vol. 68, no. 10, pp. 6858–6866, 2020, DOI: 10.1109/TAP.2020.3001743.

[21] J. Smith, Support Vector Machine: Examples With Matlab. CreateSpace Independent Publishing Platform, 2017.

[22] D. J. Armaghani, P. G. Asteris, B. Askarian, M. Hasanipanah, R. Tarinejad, and V. Van Huynh, "Examining hybrid and single SVM models with different kernels to predict rock brittleness," Sustain., vol. 12, no. 6, pp. 1–17, 2020, DOI: 10.3390/su12062229.

[23] M. N. Jebur, B. Pradhan, and M. S. Tehrany, "Optimization of landslide conditioning factors using very high-resolution airborne laser scanning (LiDAR) data at catchment scale," Remote Sens. Environ., vol. 152, pp. 150–165, 2014, DOI: 10.1016/j.rse.2014.05.013.

[24] J. Xu, Y. Zhang, and D. Miao, "Three-way confusion matrix for classification: A measure driven view," Inf. Sci. (Ny)., vol. 507, pp. 772–794, Jan. 2020, DOI: 10.1016/J.INS.2019.06.064.

[25] I. Diakonikolas, D. M. Kane, V. Kontonis, and N. Zarifis, "Algorithms and SQ Lower Bounds for PAC Learning One-Hidden-Layer ReLU Networks," Proceedings of Machine Learning Research, vol. 125. PMLR, pp. 1514–1539, 15-Jul-2020.

[26] B. Yekkehkhany, A. Safari, S. Homayouni, and M. Hasanlou, "A comparison study of different kernel functions for SVM-based classification of multi-temporal polarimetry SAR data," Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. - ISPRS Arch., vol. 40, no. 2W3, pp. 281–285, 2014, DOI: 10.5194/ISPRSARCHIVES-XL-2-W3-281-2014.

[27] H. Shaheen, S. Agarwal, and P. Ranjan, "MinMaxScaler Binary PSO for Feature Selection," Adv. Intell. Syst. Comput., vol. 1045, pp. 705–716, 2020, DOI: 10.1007/978-981-15-0029-9_55.

[28] T. T. Dien, S. H. Luu, N. Thanh-Hai, and N. Thai-Nghe, "Deep learning with data transformation and factor analysis for student performance prediction," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 8, pp. 711–721, 2020, DOI: 10.14569/IJACSA.2020.0110886.