Building of Informatics, Technology and Science (BITS)
Volume 6, No 1, June 2024 Page: 421-429
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i1.5363

Implementation of the Naïve Bayes Algorithm to Predict New Student Admissions

Aulia Salsabila*, Marnis Nasution, Irmayanti

Faculty of Science and Technology, Information System, Universitas Labuhanbatu, Rantauprapat, Indonesia
Email: [1,*]auliasalsabilanst02@gmail.com, [2]marnisnst@gmail.com, [3]irmayantiritonga2@gmail.com
Correspondence Author Email: auliasalsabilanst02@gmail.com
Submitted: **17/06/2024**; Accepted: **30/06/2024**; Published**: 30/06/2024**

Abstract−New student admissions are critical to the success of an educational institution because they determine the existence and financial sustainability of that institution. The number of prospective students who register changes every year. The school cannot anticipate the number of students who will come. Additionally, data on prospective students who enroll is collected annually without being analyzed to extract valuable information. The school must make predictions to estimate the number of new students in the next school year. Predictions are essential for effective planning, both in the long and short term. This research aims to apply the Naïve Bayes algorithm with Gaussian type to predict new student admissions. To find out whether the Naïve Bayes algorithm works well, an evaluation matrix is used. The methods applied include the dataset collection process, data preprocessing, split data training and testing, feature engineering, the implementation of Naïve Bayes, and results evaluation. The dataset is divided into 70% training data and 30% testing data. The research results show an accuracy score of 86.11% during training and an accuracy score of 90.62% during model testing, with an increase of 4.51%. These results show that there is no indication of overfitting in the machine learning algorithm used. The evaluation matrix produces an accuracy score of 90.62%, precision of 100%, recall of 90.62%, and f1-score of 95.08%. From the results of the evaluation matrix scores, it can be concluded that the naive Bayes algorithm with Gaussian type succeeded in predicting new student admissions well.

Keywords: Admission; Gaussian; Naïve Bayes; New Student; Prediction

1. INTRODUCTION

Registration and admissions of new students are important parts of the success of an educational institution because they determine its existence and financial sustainability. Some educational institutions, especially those that are not fully funded by the government or that do not have other sources of income such as endowments and grants, may wish to accept more students while still providing sufficient resources [1]. Along with progress in the field of education, competition between schools is also getting tougher. Schools that can increase the number of students will be the main goal [2]. Every year, the number of prospective students who register fluctuates. The school cannot predict how many prospective students it will have in the future. Apart from that, data on prospective students who register accumulates every year without ever being analyzed to obtain useful information [3].

SMP Negeri 2 Bilah Hilir always accepts new students every year. The activity of accepting new female students is the starting point for determining the smooth functioning of a school, with the assistance of teaching staff and equipped with optimal facilities and infrastructure for teaching and learning activities, producing students who are skilled and broad-minded. The number of prospective students who register at SMP Negeri 2 Bilah Hilir fluctuates every year. Schools cannot predict the number of prospective students they will have in the future based on previously recorded data. These predictions greatly influence the decision to determine the number of facilities and infrastructure that must be provided. Effective planning for both the long and short term relies on predictions. If this prediction is implemented in the planning process, the school will be more helpful in scheduling and meeting activity needs because this prediction can provide the best output, so it is hoped that the risk of errors caused by planning errors can be reduced to a minimum.

To find out the number of new students in the coming year, it is necessary to make predictions or forecasts based on these conditions. The new student admissions team not only has to carry out promotional activities, but they also have to be able to estimate how many new students there will be in the upcoming new school year [4]. By knowing the number of new students, schools can know the policies used to add students [5]. These predictions greatly influence the decision to determine the number of facilities and infrastructure that must be provided. Effective planning for both the long and short term relies on predictions.

Machine learning is growing rapidly in the education industry and has the potential to improve many important aspects of teaching and learning, research, and decision-making [6]. Over the past few decades, a number of researchers and scientists have shown interest in the use of machine learning in education [7], [8], [9]. Recent developments provide us with valuable tools by leveraging machine learning to explore and exploit education data [10]. Machine learning algorithms actually help organizations, including educational institutions, improve operational performance and decision-making processes and reduce forecasting errors [11].

The application of machine learning to predicting new student admissions has also been carried out by previous researchers by applying various types of algorithms. The results of research conducted by [12], shows that random forest regression is the most suitable machine learning algorithm for predicting university admissions. Based on the results of research conducted by [13], the random forest algorithm has a high ability to predict student acceptance at the high school level. The application of the Naïve Bayes algorithm carried out by [14] in recommending new student

Building of Informatics, Technology and Science (BITS)
Volume 6, No 1, June 2024 Page: 421-429
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i1.5363

admission strategies resulted in an accuracy of 72%. By applying the Naïve Bayes algorithm to the classification of prospective new students, an accuracy of 73% was obtained [15].

In predicting new student admissions, it is important to choose the right algorithm. Several studies have proven that the Naïve Bayes algorithm is superior to other algorithms in case studies in the field of education. Based on the results of comprehensive model experiments, it can be concluded that the Naïve Bayes algorithm has the best performance compared to the C4.5 algorithm. This is indicated by an Area Under Curve value of 92% and an accuracy of 94% [16]. According to research conducted by [17], the results show that the Naïve Bayes algorithm is more accurate than the random forest and C4.5 algorithms. The difference in accuracy between naïve Bayes and random forest is 2.84%, and the difference in accuracy between naïve bayes and C4.5 is 3.53%. The results of research conducted by [18] show the superiority of the Naïve Bayes algorithm compared to deep learning and random forest, with accuracy results of 99.79%. According to the results of research conducted by [19], the Naïve Bayes algorithm has the highest accuracy of 75% compared to random forests and decision trees. The Naïve Bayes algorithm also has several advantages, namely that it has high performance even on small amounts of data, performs analysis quickly, and is not affected by changes in the training data ratio [20].

In predicting new student admissions, the Naïve Bayes algorithm has also been applied in several previous studies. Research conducted by [21] aims to determine the application of the Naïve Bayes classification to the average of report card grades and national exam scores to the level of student acceptance in state high schools or vocational schools using the results of the classification model that was formed. In this research, the data used is data from new vocational school students. The data mining process was assisted by WEKA software using Naïve Bayes classification and 10-fold cross validation. Next, the Naïve Bayes classification model is used to process the prediction data. The results of testing the Naïve Bayes algorithm in predicting new student admissions to Vocational High Schools (SMK) on 196 student data tested in this research show that the Naïve Bayes algorithm has an accuracy rate of 86.22%. However, this research only uses the average report card score. Another study conducted by [5] aims to predict the rise and fall of the number of students registering using the Naïve Bayes method. Research data was obtained by randomly distributing questionnaires to 200 respondents (students) who were about to enter high school. The data was accumulated using Microsoft Excel. The results obtained were that the high-class precision prediction was 100%, while the low-class precision prediction was 94.23%. However, in this research, the extracurricular, cost, and distance criteria need attention and improvement. This is because no interest and low predictions are higher than interest with high predicted results. Research conducted by [22] aims to build a prediction system for new student admissions at MTS using the Naïve Bayes method using Python. The goal is to increase the accuracy of predictions of the number of new students who will enter each year. The student data used was 623 training data and 82 testing data used as a basis for predictions. The Naive Bayes method is used to classify data by calculating probabilities based on historical data. However, from this research, prediction results for the non-entering class were obtained with a precision percentage of 0%, recall of 0%, and f1-score of 0%. This means that the Naïve Bayes algorithm is unable to read the prediction results for classes that do not enter.

Based on the background of the problem that has been explained, this research aims to apply the Naïve Bayes algorithm to predict the admissions of new students at State Junior High School 2 Bilah Hilir, Labuhanbatu Regency. This research is different from the previous research that has been described; in this research, the Naïve Bayes algorithm with the Gaussian type is applied. The Gaussian Naive Bayes algorithm is easy to use, simple, and does not require a lot of training data. This highly scalable algorithm scales linearly with the number of features and data points, is insensitive to irrelevant features, and is very effective in dealing with missing data [23]. The performance of the Naïve Bayes algorithm will be measured based on the evaluation matrix accuracy, precision, recall, and f1-score. From the evaluation matrix, it can be seen whether the naïve Bayes algorithm with the Gaussian type works well in predicting new student admissions. This research is important to carry out to help schools in the planning process for accepting new students.

2. RESEARCH METHODOLOGY

2.1 Research Stages

This section presents the methodology used to predict student acceptance at SMP Negeri 2 Bilah Hilir based on the Naïve Bayes algorithm. The processes carried out are dataset collection, data preprocessing, split data training and testing, feature engineering, implementation of Naïve Bayes, and results evaluation. The research stages are shown in Figure 1. The method used in this research comes from research conducted by [5] and [24].

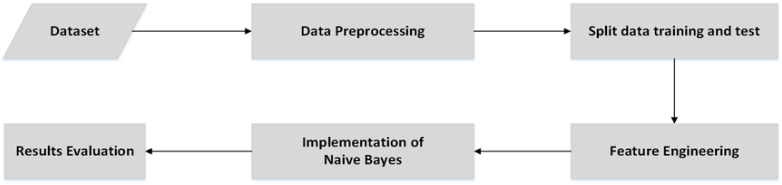

Figure 1. The Research Stages

Building of Informatics, Technology and Science (BITS)
Volume 6, No 1, June 2024 Page: 421-429
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i1.5363

2.1.1 Dataset

In this research, the initial stage carried out was preparing data, where the data obtained came from the operator of SMP Negeri 2 Bilah Hilir. The data used in this research is student registration data at SMP Negeri 2 Bilah Hilir in 2023, which is used as a basis for predicting the acceptance of new students.

Table 1. Sample of dataset

Students	Sex	School Origin District	National Examination Score	Distance (meter)	Results
student001	Female	Panai Hulu	251.67	4950	Accepted
student002	Female	Bilah Hilir	296.33	1500	Accepted
student003	Female	Panai Tengah	249.00	7000	Accepted
student004	Female	Bilah Hilir	240.67	9240	Not accepted
student005	Male	Panai Tengah	234.67	11288	Not accepted
student006	Female	Bilah Hilir	281.00	9240	Not accepted
student007	Female	Bilah Hilir	265.00	509	Accepted
student008	Male	Bilah Hilir	241.00	700	Not accepted
student009	Male	Panai Hulu	226.33	4950	Not accepted
student010	Mele	Bilah Hilir	270.33	300	Accepted

Table 1 shows a sample research dataset. The dataset used is data on prospective students who register in 2023, with a total of 104 rows of records. The research dataset consists of the sex attribute, the school district of origin, national exam scores, distance from home to school, and the results attribute, namely whether students are accepted or not accepted at the school.

2.1.2 Data Preprocessing

The goal of preprocessing is to clean and convert raw data into a format that can be used successfully by the selected algorithm [25]. Data pre-processing in this research includes the processes of handling missing values and dividing data into categorical data and numerical data. In the process of handling missing values, missing values will be filled with the value 0 or values that appear frequently [26]. In this section, the dataset is separated into categorical and numeric variables.vThen an exploration of categorical and numerical variables was carried out to see whether there were missing values in the data and their cardinality. This stage also checks for missing values in the variables used..

2.1.3 Split Data Training and Test

For model validation, data splitting is commonly used. This method divides a given data set into two different sets: training and testing. Statistical and machine learning models are then fitted to the training set and validated against the test set. By providing a data set for validation that is different from training, we can evaluate and compare the predictive performance of different models without worrying [27]. In this section, we will divide the amount of training data and testing data into the dataset. This research applies a distribution of training data of 70% and test data of 30% [28].

2.1.4 Feature Engineering

Before applying any machine learning algorithm, these raw data sources need to be transformed into applicable and meaningful features that represent specific observational properties. Features typically appear as columns in a data matrix provided to a machine learning algorithm. This important step, often referred to as feature engineering, is the most important step in the machine learning process [29]. This stage is the process of turning raw data into useful features. This process helps in understanding the model better and increases its predictive power. At this stage, feature engineering will be carried out on various types of variables. Feature engineering was done using library OneHotEncoder in Python. One-hot encoding is used for categorical data types that only consist of two options [26]. Then, all feature variables are put on the same scale. To map variables to the same scale, the RobustScaler library in Python is used. RobustScaler is a method from Scikit-Learn that is used to normalize or scale data with the aim of improving the performance of machine learning models. RobustScaler is designed to handle outliers better than other scaling methods, such as StandardScaler or MinMaxScaler. This is done by using the median and interquartile range (IQR) of the data rather than the mean and standard deviation. RobustScaler normalizes individual features so that each feature falls within the same range. This helps machine learning models to converge faster and better because features that are in the same range do not dominate other features [30].

2.1.5 Implementation of Naïve Bayes

In this process, classification is carried out using Gaussian Naïve Bayes. The first thing to do is train the model, and after that, the prediction process is carried out. Then, the accuracy of the model score is checked and compared with the accuracy of the test set to check for overfitting. Berikut persamaan dari Naïve Bayes.

$$P(C) = \prod_{i}^{n} = 1p(X_i|C) \quad (1)$$

Building of Informatics, Technology and Science (BITS)
Volume 6, No 1, June 2024 Page: 421-429
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i1.5363

The Gaussian Naïve Bayes algorithm is shown in the following steps [31].

a. To train the date splitting the dataset into 70%, the remaining 30% is used for testing.
b. Training phase:
   1. Total = all instances in the training dataset
      C_j is class in the dataset of training
   2. Probability of every single class is calculating
      $P(C_j)$ = frequency (C_j)/total
   3. Calculate the mean (μ) as well as the standard deviation (σ) values of each of the training dataset class attributes. Note down the result.
c. Testing phase:
   1. In testing DSX is an instance
   2. By applying equation (1), the probability density function value of X is calculated at C_p for values of X attributes remain in S, $p\ (X_i|C_j)$
   3. By using the equation, $P\ (X|Cj) = \prod_i^n = 1p\left(f_j|C_i\ \right)$, for the values, resulting from step of 3.2, the conditional probability value of X is calculated at C_j
   4. By using an equation, $P(X) = P\left(c_j\right).p\left(C_j\right)$, here, $p\left(C_j|X\right)$ represents the probability value of instance at C_j, and then, the posterior probability of X can be calculated
   5. By selecting maximization $P\left(C_j|X\right)$, assign a X class label.
d. Return the class name

2.1.6 Results Evaluation

The final process of this research uses various evaluation metric criteria to measure the effectiveness of the model created. In this section, the results of the confusion matrix are also seen. The confusion matrix is a visualization tool commonly used in supervised learning. Each column in the matrix is an example of the predicted class, while each row represents events in the actual class. This method only uses a matrix table in the process if the dataset has a class, namely a class that is considered positive and the other class is a negative class. Evaluation with this confusion matrix produces accuracy, precision, and recall values for the classification that has been carried out [32]. Confusion matrix provide a clear picture of how a classification model functions and the types of errors it produces. It also provides a summary of correct and incorrect predictions, grouped by category. This summary is presented in image form. To assess the performance of the classification model, scores for accuracy, precision, recall, and f1-score will be displayed. It has the following definitions formally [33]:

$$Accucary = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

$$Recall = \frac{TP}{TP+FN} \quad (4)$$

$$F1-score = 2 * \frac{(Precision * Recall)}{(Precision + recall)} \quad (5)$$

True positive are called TP, false positive are called FP, true negative are called TN, and false negative are called FN. True positives and false positives are the number of positive and negative records that are classified as positive, while false negatives and true negatives are the number of positive and negative records that are classified as negative. Then enter the test data, and after that, calculate the values that have been entered to calculate the sensitivity, specifications, precision, and accuracy. Based on the contents of the matrix in the table, it can be seen the amount of data from each class that was predicted correctly, namely (true positives + true negatives), and the data that was classified incorrectly was (false positives + false negatives) [34].

3. RESULT AND DISCUSSION

3.1 Result

In this research, the Gaussian naïve Bayes classifier model has been applied to predict new student admissions. Python is a programming language used from data preprocessing to the results evaluation process, using the Jupyter Notebook text editor on Google Colab. Naive Bayes classification is an easy and powerful machine learning algorithm for classification tasks. This research uses the Scikit-Learn library in Python to implement the Naive Bayes classification algorithm in this kernel. Information about the dataset used in this research is shown in Figure 2.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 1, June 2024 Page: 421-429
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i1.5363

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 104 entries, 0 to 103
Data columns (total 5 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   sex          104 non-null    object
 1   subdistrict  104 non-null    object
 2   test_scores  104 non-null    float64
 3   distance     104 non-null    int64
 4   results      104 non-null    object
dtypes: float64(1), int64(1), object(3)
memory usage: 4.2+ KB
```

Figure 2. Summary of Dataset

Figure 2 shows a summary of the dataset used in this research after the data preprocessing stage. Figure 2 shows that the dataset consists of 104 entries and 5 columns. The column consists of the attributes sex, subdistrict, test_scores, distance, and results, which are the target attributes. The sex, subdistrict, and results attributes have the object data type. Meanwhile, the test_scores attribute has a float data type, and the distance attribute has an integer data type. Next, data preprocessing is carried out by handling missing values, as shown in Figure 3.

	Total of missing	Percentage of missing
sex	0	0.0%
subdistrict	0	0.0%
test_scores	0	0.0%
distance	0	0.0%
results	0	0.0%

Figure 3. Handling Missing Values

Figure 3 shows a function to display features with the number and percentage of missing values by calculating the total number of missing values for each feature and calculating the percentage of missing values for each feature. From the results of the Handling Missing Values process, it is shown that of all the features used, there are no missing values with a percentage of 0%. The next process is to divide the dataset into categorical and numerical data types, as shown in Figure 4.

```
df[categorical].head()
```

	sex	subdistrict	results
0	Male	Bilah Hilir	Accepted
1	Male	Bilah Hilir	Accepted
2	Female	Bilah Hilir	Accepted
3	Male	Bilah Hilir	Accepted
4	Female	Bilah Hilir	Accepted

```
df[numerical].head()
```

	test_scores	distance
0	229.00	9240
1	244.00	2580
2	242.00	9240
3	238.67	1500
4	255.00	3800

Figure 4. Categorical and Numerical Data

Figure 4 shows the attributes in the dataset, which consist of two types of variables, namely categorical and numerical. Categorical variables have the object data type, and numeric variables have the float64 and int64 data types. There are three attributes that fall into this type of category, namely, gender, subdistrict, and results. Meanwhile, in the numeric type, there are two attributes, namely test scores and distance. In this data set, there are no missing values. The gender attribute consists of two labels, namely, male and female. The sub-district attribute consists of three labels, namely: downstream, upstream, and mid-panai. Meanwhile, the result attributes consist of Accepted and Not Accepted. The next stage is splitting the dataset into training data and testing data using the Scikit-Learn library in Python, as shown in Figure 5.

```
# split X and y into training and testing sets
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 0)
```

```
[ ] # check the shape of X_train and X_test
    X_train.shape, X_test.shape
```

```
((72, 4), (32, 4))
```

Figure 5. Splitting of Dataset

Building of Informatics, Technology and Science (BITS)
Volume 6, No 1, June 2024 Page: 421-429
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i1.5363

Figure 5 shows the process of dividing data into a training set and a test set using Python. This division is an important step in implementing the Gaussian Naïve Bayes algorithm to predict new student admissions. The first step is to import the train_test_split function from the sklearn.model_selection module. This function is used to divide a dataset into two parts: a training set and a testing set. The train_test_split function is used to split the datasets X (features) and Y (labels) into a training set and a test set. X_train and y_train are training data. X_test and y_test are test data. Test_size = 0.3 means 30% of the data will be used as the test set, while 70% will be used as the training set. Random_state = 0 ensures that the data split will be consistent every time the code is run, so the results are reproducible. This line is used to check the shape (dimensions) of the training set and testing set; X_train.shape shows the shape of the training data, and X_test.shape shows the shape of the testing data. The results show that the training set X_train has 72 samples with 4 features, while the testing set X_test has 32 samples with 4 features. The dataset has been successfully divided into a training set and a test set with a ratio of 70:30. The use of random_state ensures that data distribution is consistent and reproducible. After this data sharing, the training set can be used to train a Gaussian Naïve Bayes model, and the test set can be used to evaluate the performance of the model. In the next stage, a feature engineering process was carried out on the dataset by applying OneHotEncoder in Python; the results are shown in Figure 6.

	sex_1	sex_2	subdistrict_1	subdistrict_2	subdistrict_3	test_scores	distance
26	1	0	1	0	0	239.67	3800
61	1	0	1	0	0	238.65	12279
2	1	0	1	0	0	242.00	9240
62	1	0	1	0	0	263.00	20000
85	1	0	1	0	0	283.00	9240

Figure 6. OneHot Encoder Results

Figure 6 shows the results of OneHot Encoder on the research dataset. The data that appears is five pieces of data in rows 26, 61, 2, 62, and 85. From the picture, it appears that the sex attribute is broken down according to the number of labels, namely sex_1 and sex_2. Sex_1 is male, while Sex_2 is female. Subdistricts are divided according to the number of labels, namely subdistrict_1, subdistrict_2, and subdistrict_3. Subdistrict_1 is for the Bilah Hilir subdistrict, Subdistrict_2 is for the Panai Hulu subdistrict, and Subdistrict_3 is for the Panai Tengah subdistrict. Then, all feature variables are converted to the same scale. To convert the data to the same scale, the RobustScaler library in Python was used. The results are shown in figure 7.

	sex_1	sex_2	subdistrict_1	subdistrict_2	subdistrict_3	test_scores	distance
0	0.0	0.0	0.0	0.0	0.0	-0.821449	1.026486
1	-1.0	1.0	0.0	0.0	0.0	-0.222029	-0.330103
2	-1.0	1.0	-1.0	1.0	0.0	-0.337971	0.934496
3	0.0	0.0	0.0	0.0	0.0	-0.608696	-0.000646
4	-1.0	1.0	0.0	0.0	0.0	0.183768	2.060078

Figure 7. Scaling Results from RobustScaler

Figure 7 shows the results of scaling the dataset using RobustScaler. In addition, all the values of the attributes have also been converted to the same scale. Apart from that, all attribute values have also been converted to the same scale, namely between -1 and 1. The dataset is ready for training and testing on the Gaussian Naïve Bayes model. The accuracy results of the training and testing process by applying the Gaussian Naïve Bayes algorithm are shown in Figure 8.

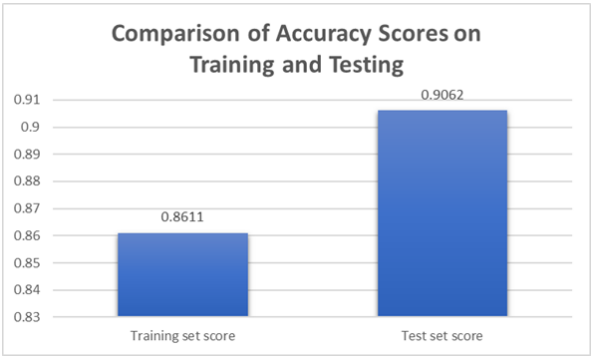

Figure 8. Accuracy Score

Building of Informatics, Technology and Science (BITS)
Volume 6, No 1, June 2024 Page: 421-429
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i1.5363

Figure 8 shows a comparison of accuracy scores between training data and testing data in the implementation of the Gaussian Naïve Bayes algorithm to predict new student admissions. The graph shows that the Naïve Bayes model has an accuracy of 86.11% when tested on training data. This means that the model was able to correctly predict 86.11% of the training data used to train the model. The graph also shows that the model's accuracy increased to 90.62% when tested on test data. This shows that the model performed better on never-before-seen data, being able to correctly predict 90.62% of the test data. There was an increase in accuracy scores during testing by 4.51%. Overall, this graph provides a clear picture of the model's performance in the training and testing stages and shows that the model has good generalization capabilities. However, we cannot say that the model results of this test are very good based on the above accuracy. We should compare it to zero accuracy. Zero accuracy is the accuracy that can be achieved by always predicting the most frequently occurring class. The evaluation of test results using a confusion matrix is shown in Figure 9.

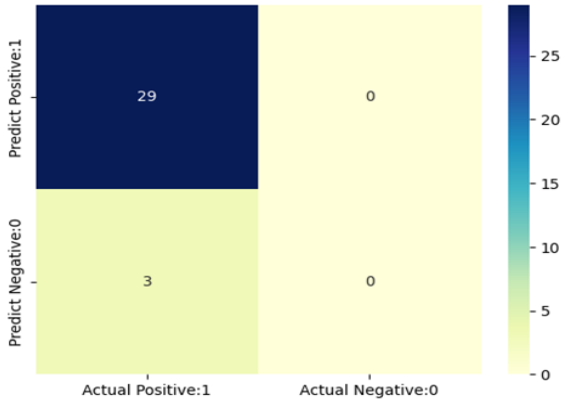

Figure 9. Confusion Matrix

Figure 9 is a confusion matrix resulting from predictions using the Gaussian Naïve Bayes algorithm to predict new student admissions. In this confusion matrix, there are four quadrants that show the relationship between predictions and actual results: Predict Positive: 1 (Line 1): Prediction that the student will be accepted. Actual Positive: 1 (Column 1): Students who are actually accepted. Actual Negative: 0 (Column 2): Students who were not actually accepted. Predict Negative:0 (Line 2): Predicts that the student will not be accepted. Actual Positive: 1 (Column 1): Students who are actually accepted. Actual Negative: 0 (Column 2): Students who were not actually accepted. True Positives (TP): The model predicted that 29 students would be admitted, and they were actually admitted. This is a correct prediction for positive cases. False Negatives (FN): The model predicts that 3 students will not be accepted, when in fact they were. This is the wrong prediction for positive cases. False Positives (FP): The model predicts that no students will be admitted, when in fact they are not. This is the wrong prediction for negative cases. In this case, no students were incorrectly predicted to be accepted. True Negatives (TN): No students were actually not accepted. Thus, true negatives are not detected in this matrix. Based on the values in the confusion matrix, the model correctly predicted that 29 students were accepted from a total of 32 cases (29 TP + 3 FN). Model precision cannot be calculated directly from this matrix because there are no false positives. Recall shows how well the model is at recognizing true positive cases; in this case, 29/32 = 0.9062, or 90.62%. The model has high recall, but the presence of false negatives (3 cases) shows that there were some students who were accepted but were not predicted by the model. The score from the performance evaluation matrix of the Gaussian Naïve Bayes algorithm, which includes accuracy, precision, recall, and f1-score, is shown in Table 2.

Table 2. the Results of Evaluation Matrix

Evaluation Matrix	Score
Accucary	90.62%
Precision	100%
Recall	90.62%
F1-score	95.08%

Table 2 shows the scores from the evaluation matrix from prediction testing on the Gaussian Naïve Bayes algorithm. From the table, it appears that the accuracy score obtained in making classification predictions is 90.62%. Accuracy measures the proportion of correct predictions over the total predictions made by the model. In this context, the Naïve Bayes model has an accuracy of 90.62%, which means that of all the data evaluated, 90.62% of the predictions made by the model are correct. This shows that the model is quite good at predicting overall new student admissions. The resulting precision in identifying the proportion of positive results that were predicted correctly was 100%. Precision measures the proportion of correct positive predictions over all positive predictions made by the model. With a precision value of 100%, this means that all student admission predictions made by the model are

Building of Informatics, Technology and Science (BITS)
Volume 6, No 1, June 2024 Page: 421-429
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i1.5363

correct. There were no cases where the model predicted a student would be admitted but was not actually admitted. The resulting recall in identifying the actual positive proportion that was predicted correctly was 90.62%. Recall (or sensitivity) measures the proportion of correct positive predictions over all positive actual cases in the data. A recall of 90.62% indicates that of all the students actually admitted, the model managed to identify 90.62% of them correctly. This shows the model's ability to recognize truly admitted students is quite good. The F1-score resulting from the harmonic average of precision and recall is 95.08%. The F1-score is the harmonic average of precision and recall. This provides a measure of balance between precision and recall. An F1-score of 95.08% indicates that the model has a very good balance between the ability to recognize all accepted students (recall) and ensuring that all admission predictions are correct (precision).

3.2 Discussion

The application of the Naïve Bayes algorithm with the Gaussian type in predicting new student admissions has resulted in an accuracy of 90.62%. However, based on this accuracy, we cannot say that this model is very good; we should compare it with zero accuracy, which is the accuracy that can be achieved by always predicting the most frequently occurring classes. After making a comparison with zero accuracy, the accuracy results obtained remained the same, namely, 90.62%. Therefore, we can draw the conclusion that the Gaussian Naive Bayes classification model works well to predict class labels. The training accuracy score was 86.11%, while the testing accuracy score was 90.62%. There was an increase in accuracy scores during testing by 4.51%. This shows that there are no signs of overfitting in the machine learning algorithm applied.

This research certainly has limitations. The resulting F1-score was 95.08%, greater than the accuracy score. The f1 score is always lower than accuracy measures because it embeds recall and precision in the computation. The results of the confusion matrix on 32 test data obtained 29 true positive results, meaning that 29 students were indeed positively accepted as new students. Meanwhile, three students were declared false negative, meaning that there were fake negative scores. Of course, these problems can be developed to find solutions through further research.

4. CONCLUSION

The application of the Naïve Bayes algorithm with the Gaussian type in predicting new student admissions has been carried out by applying research steps that are in accordance with the method. A higher accuracy score on test data is a good indicator that the applied Gaussian Naïve Bayes model is able to predict new student admissions well, even when given never-before-seen data. The implementation of the Gaussian Naïve Bayes algorithm in predicting new student admissions shows positive results, with fairly high accuracy on test data. This shows the potential of this algorithm to be used in real applications to predict new student admissions with a high degree of accuracy. This confusion matrix shows that the Gaussian Naïve Bayes algorithm applied to predict new student admissions has good performance with high recall, although there are several false negatives. This means the model does a good job of identifying the majority of accepted students, but there are still some it misses. This evaluation helps in understanding where the model may need further improvement. From the results of the evaluation matrix scores, accuracy of 90.62%, precision of 100%, recall of 90.62%, and f1-score of 95.08%, it can be concluded that the naïve Bayes algorithm with the Gaussian type has good performance in predicting new student admissions at SMP Negeri 2 Bilah Hilir. Hopefully, the results of this research can contribute to helping schools prepare to accept new students. Overall, the Gaussian Naïve Bayes model used to predict new student admissions shows excellent performance with perfect precision and recall and a very high F1-score. This shows that the model is very effective in predicting new student admissions with little error.

REFERENCES

[1] J. Cirelli, A. M. Konkol, F. Aqlan, and J. C. Nwokeji, "Predictive Analytics Models for Student Admission and Enrollment," in *Proceedings of the International Conference on Industrial Engineering and Operations Management*, 2018, pp. 1395–1403.

[2] G. W. N. Wibowo, Z. Arifin, M. A. Romli, and N. I. Amal, "Prediksi Kelanjutan Studi Siswa Ke Perguruan Tinggi Dengan Naive Bayes," *J. DISPROTEK*, vol. 11, no. 1, pp. 41–46, 2020, doi: 10.34001/jdpt.v11i1.1159.

[3] S. Rizal and M. Lutfi, "Penerapan Algoritma Naïve Bayes Untuk Prediksi Penerimaan Siswa Baru Di SMK Al-Amien Wonorejo," *Explor. IT J. Keilmuan dan Apl. Tek. Inform.*, vol. 10, no. 1, pp. 14–21, 2018, doi: 10.35891/explorit.v10i1.1671.

[4] P. D. Silitonga, H. Himawan, and R. Damanik, "FORECASTING ACCEPTANCE OF NEW STUDENTS USING DOUBLE EXPONENTIAL SMOOTHING METHOD," *J. Crit. Rev.*, vol. 7, no. 1, pp. 300–305, 2020, doi: 10.31838/jcr.07.01.57.

[5] S. Suwayudhi, E. Irawan, and B. E. Damanik, "Teknik Klasifikasi dalam Memprediksi Penerimaan Siswa Baru Menggunakan Metode Naive Bayes," *JOMLAI J. Mach. Learn. Artif. Intell.*, vol. 1, no. 3, pp. 251–256, 2022, doi: 10.55123/jomlai.v1i3.963.

[6] N. A. Jalil, H. J. Hwang, and N. M. Dawi, "Machines Learning Trends, Perspectives and Prospects in Education Sector," in *Proceedings of the 3rd International Conference on Education and Multimedia Technology*, in ICEMT '19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 201–205. doi: 10.1145/3345120.3345147.

[7] K. T. Chui, D. C. L. Fung, M. D. Lytras, and T. M. Lam, "Predicting at-risk university students in a virtual learning environment via a machine learning algorithm," *Comput. Human Behav.*, vol. 107, p. 105584, 2020, doi: https://doi.org/10.1016/j.chb.2018.06.032.

[8] A. Qazdar, B. Er-Raha, C. Cherkaoui, and D. Mammass, "A machine learning algorithm framework for predicting students

Building of Informatics, Technology and Science (BITS)
Volume 6, No 1, June 2024 Page: 421-429
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i1.5363

performance: A case study of baccalaureate students in Morocco," *Educ. Inf. Technol.*, vol. 24, no. 6, pp. 3577–3589, 2019, doi: 10.1007/s10639-019-09946-8.

[9] M. Segura, J. Mello, and A. Hernández, "Machine Learning Prediction of University Student Dropout: Does Preference Play a Key Role?," *Mathematics*, vol. 10, no. 18. 2022. doi: 10.3390/math10183359.

[10] B. Albreiki, N. Zaki, and H. Alashwal, "A Systematic Literature Review of Student' Performance Prediction Using Machine Learning Techniques," *Educ. Sci.*, 2021.

[11] E. M. Onyema *et al.*, "Prospects and Challenges of Using Machine Learning for Academic Forecasting.," *Comput. Intell. Neurosci.*, vol. 2022, p. 5624475, 2022, doi: 10.1155/2022/5624475.

[12] I. El Guabassi, Z. Bousalem, R. Marah, and A. Qazdar, "A Recommender System for Predicting Students' Admission to a Graduate Program using Machine Learning Algorithms," *Int. J. Online Biomed. Eng.*, vol. 17, no. 02, pp. 135–147, 2021, doi: 10.3991/ijoe.v17i02.20049.

[13] Rasiban and S. P. R. Maruli, "Penerapan Data Mining Untuk Memprediksi Penerimaan Peserta Didik Baru Jalur Prestasi Akademik Di SMA Negeri 13 Jakarta Dengan Menggunakan Algoritma Random Forest," *Innov. J. Soc. Sci. Res.*, vol. 3, no. 4, pp. 10065–10079, 2023.

[14] A. H. Sani, A. Setiawan, and A. Primadewi, "Penerapan Metode Naive Bayes Dalam Rekomendasi Strategi Penerimaan Peserta Didik Baru," *J. Comput.*, vol. 4, no. 1, pp. 245–251, 2022, doi: 10.47065/josyc.v4i1.2438.

[15] I. Loelianto, M. S. S. Thayf, and H. Angriani, "IMPLEMENTASI TEORI NAÏVE BAYES DALAM KLASIFIKASI CALON MAHASISWA BARU STMIK KHARISMA MAKASSAR," *SINTECH*, vol. 3, no. 2, pp. 110–117, 2020, doi: 10.31598/sintechjournal.v3i2.651.

[16] R. Ramadani, B. H. Hayadi, and H. Hartono, "Comparative Analysis of Algorithms Naïve Bayes and C45 for Student Satisfaction with Administrative Services," in *2023 International Conference of Computer Science and Information Technology (ICOSNIKOM)*, 2023, pp. 1–6. doi: 10.1109/ICoSNIKOM60230.2023.10364373.

[17] W. Gata *et al.*, "Algorithm Implementations Naive Bayes, Random Forest. C4.5 on Online Gaming for Learning Achievement Predictions," in *Proceedings of the 2nd International Conference on Research of Educational Administration and Management (ICREAM 2018)*, Atlantis Press, Mar. 2019, pp. 1–9. doi: 10.2991/icream-18.2019.1.

[18] Nurhachita and E. S. Negara, "A comparison between deep learning, naïve bayes and random forest for the application of data mining on the admission of new students," *IAES Int. J. Artif. Intell.*, vol. 10, no. 2, pp. 324–331, 2021, doi: 10.11591/ijai.v10.i2.pp324-331.

[19] M. Garonga and Rita Tanduk, "COMPARISON OF NAIVE BAYES, DECISION TREE, AND RANDOM FOREST ALGORITHMS IN CLASSIFYING LEARNING STYLES OF UNIVERSITAS KRISTEN INDONESIA TORAJA STUDENTS," *J. Tek. Inform.*, vol. 4, no. 6 SE-Articles, pp. 1507–1514, Dec. 2023, doi: 10.52436/1.jutif.2023.4.6.1020.

[20] İ. Koyuncu and S. Gelbal, "Comparison of Data Mining Classification Algorithms on Educational Data under Different Conditions," *J. Meas. Eval. Educ. Psychol.*, vol. 11, no. 4, pp. 325–345, 2020, doi: 10.21031/epod.696664.

[21] S. Rizal and M. Lutfi, "Penerapan Algoritma Naïve Bayes Untuk Prediksi Penerimaan Siswa Baru Di Smk Al-Amien Wonorejo," *Explor. IT J. Keilmuan dan Apl. Tek. Inform.*, vol. 10, no. 1, pp. 14–21, 2018, doi: 10.35891/explorit.v10i1.1671.

[22] F. Santoso, Sunardi, and H. Z. Lukman, "Implementasi Data Mining dengan Metode Naive Bayes Untuk Memprediksi Penerimaan Siswa Baru di MTS NU Islamiyah Asembagus," *G-Tech J. Teknol. Terap.*, vol. 7, no. 4, pp. 1355–1366, 2023, doi: 10.33379/gtech.v7i4.3086.

[23] E. K. Ampomah, G. Nyame, Z. Qin, P. C. Addo, E. O. Gyamfi, and M. Gyan, "Stock Market Prediction with Gaussian Naïve Bayes Machine Learning Algorithm," *Informatica*, vol. 45, pp. 243–256, 2021, doi: 10.31449/inf.v45i2.3407.

[24] Afdhaluzzikri, H. Mawengkang, and O. S. Sitompul, "Perfomance of Naive Bayes method with data weighting," *SinkrOn*, vol. 7, no. 3, pp. 817–821, 2022, doi: 10.33395/sinkron.v7i3.11516.

[25] T. Agustina, M. Masrizal, and I. Irmayanti, "Performance Analysis of Random Forest Algorithm for Network Anomaly Detection using Feature Selection," *Sinkron*, vol. 8, no. 2, pp. 1116–1124, 2024, doi: 10.33395/sinkron.v8i2.13625.

[26] S. S. Muliani, V. Sihombing, and I. R. Munthe, "Implementation of Exploratory Data Analysis and Artificial Neural Networks to Predict Student Graduation on-Time," *Sink. J. dan Penelit. Tek. Inform.*, vol. 8, no. 2, pp. 1188–1199, 2024, doi: 10.33395/sinkron.v8i2.13658.

[27] V. R. Joseph, "Optimal ratio for data splitting," *Stat. Anal. Data Min. ASA Data Sci. J.*, vol. 15, no. 4, pp. 531–538, Aug. 2022, doi: https://doi.org/10.1002/sam.11583.

[28] Q. H. Nguyen *et al.*, "Influence of Data Splitting on Performance of Machine Learning Models in Prediction of Shear Strength of Soil," *Math. Probl. Eng.*, vol. 2021, no. 1, p. 4832864, Jan. 2021, doi: https://doi.org/10.1155/2021/4832864.

[29] T. Verdonck, B. Baesens, M. Óskarsdóttir, and S. vanden Broucke, "Special issue on feature engineering editorial," *Mach. Learn.*, vol. 113, no. 7, pp. 3917–3928, 2021, doi: 10.1007/s10994-021-06042-2.

[30] A. Khoirunnisa and N. G. Ramadhan, "Improving malaria prediction with random forest and robust scaler: An integrated approach for enhanced accuracy," *J. INFOTEL*, vol. 15, no. 4, pp. 326–334, 2023, doi: 10.20895/infotel.v15i4.1056.

[31] M. V. Anand, B. KiranBala, S. R. Srividhya, K. C., M. Younus, and M. H. Rahman, "Gaussian Naïve Bayes Algorithm: A Reliable Technique Involved in the Assortment of the Segregation in Cancer," *Mob. Inf. Syst.*, no. 1, p. 2436946, Jan. 2022, doi: https://doi.org/10.1155/2022/2436946.

[32] R. F. Nasution, M. H. Dar, and F. A. Nasution, "Implementation of the Naïve Bayes Method to Determine Student Interest in Gaming Laptops," *Sinkron*, vol. 8, no. 3, pp. 1709–1723, 2023, doi: 10.33395/sinkron.v8i3.12562.

[33] Samsir, Kusmanto, A. H. Dalimunthe, R. Aditiya, and R. Wathrianthos, "Implementation Naïve Bayes Classification for Sentiment Analysis on Internet Movie Database," *Build. Informatics, Technol. Sci.*, vol. 4, no. 1, pp. 1–6, 2022, doi: 10.47065/bits.v4i1.1468.

[34] F. F. Hasibuan, M. H. Dar, and G. J. Yanris, "Implementation of the Naïve Bayes Method to determine the Level of Consumer Satisfaction," *SinkrOn*, vol. 8, no. 2, pp. 1000–1011, 2023, doi: 10.33395/sinkron.v8i2.12349.