Proceeding of 2nd International Conference on Education, Society and Humanity
e-ISSN : 2986-5832
p-ISSN : 2986-6979

Vol. 02 No. 01 (2024)
Available online at https://ejournal.unuja.ac.id/index.php/icesh

MACHINE LEARNING FOR CLASSIFICATION OF IKM PROGRAMS AT THE DEPARTMENT OF INDUSTRY AND TRADE OF LANGKAT REGENCY

Mimi Chintya Adelina[1], Wanayumini[2], Zakarias Situmorang[3]
[1,2,3]Universitas Potensi Utama, North Sumatra, Indonesia
Email: mimichintya8@gmail.com[1], wanayumini@gmail.com[2],
Zakarias65@yahoo.com[3]

Abstract:

This research attempts to address these challenges by constructing a classification model using the Naive Bayes algorithm. Naive Bayes is a statistical algorithm that can be employed to predict the probability of membership in a class based on the available data. This method can assist the Department of Industry and Trade of Langkat Regency in selecting targeted programs and identifying SMEs (Small and Medium Enterprises) with potential success. The research will involve the collection and analysis of data regarding SMEs in Langkat Regency, including information about the industry type, geographic location, and business formality status. This data will be utilized to train the Naive Bayes classification model to predict the potential success of programs offered by the Department of Trade and Industry. Consequently, it is anticipated that this model can aid in more effective and efficient decision-making in the management of SME programs.

Keywords: *Machine Learning, Naive Bayes, IKM.*

INTRODUCTION

Machine Learning is one of the fields of computer science, with a broad scope of applications that involves automatic detection of meaningful patterns in data. Machine learning tools are associated with providing programs the ability to learn and adapt (Osisanwo et al., 2017). Machine learning encompasses a set of computer algorithms used to optimize the performance of computers or systems based on existing sample data. The primary capability of machine learning is the modification and adaptation of decisions in response to changes. A reliable method used for predictive data analysis in machine learning combines elements from statistics, artificial intelligence, and computer science. In the logistics and supply chain field, machine learning for predictive analysis has been applied to various logistics activities, such as demand forecasting (Sumpena & Kurnia H., 2019)(Pambudi et al., 2020).

Supervised machine learning algorithms more associated with classification include Linear Classifier, Logistic Regression, Naïve Bayes Classifier, Perceptron, Support Vector Machine; Quadratic Classifier, K-Means Clustering, Boosting, Decision Tree, Random Forest (RF); Artificial Neural Networks, Bayesian Networks, and others (Osisanwo et al., 2017). The Naive Bayes algorithm is one of the classification techniques (Indrayuni, 2019:30). It is a statistical classification algorithm used to predict the probability of membership in a class. Bayesian classification is based on Bayes' theorem, which has classification capabilities similar to decision trees and neural networks. Bayesian classification has proven to have high accuracy and speed when applied to databases with large amounts of data. The Bayes method is a

statistical approach for inductive inference in classification issues. It first discusses the basic concepts and definitions of Bayes' Theorem, then uses this theorem for classification in Data Mining (Fitrianah et al., 2021). This model calculates a set of probabilities by summing the frequencies and mixed values from the given dataset. The Naive Bayes procedure assumes that all attributes in each type are independent of each other. Naive Bayes has been proven to have high accuracy and speed when applied to databases with large information. The advantage of using Naive Bayes is that it only requires a small amount of training data to determine the mean and variance parameters needed for classification (Hozairi et al., 2021).

The Department of Industry and Trade of Langkat Regency is responsible for developing and advancing the MSME sector by providing programs that support the growth and sustainability of MSMEs. Langkat Regency is a relatively large area with 23 sub-districts. With such a vast area, the MSME data held by the Department of Industry and Trade of Langkat Regency is extensive, requiring careful consideration of programs to achieve targeted results due to budget constraints.

According to data from the Department of Industry and Trade of Langkat Regency, there has been an increase in MSMEs from 2017 to 2022. In this context, the department faces challenges in classifying potential successful MSMEs in these programs. The programs launched by the Department of Industry and Trade of Langkat Regency aim to provide support, training, and facilities to MSMEs for sustainable growth.

Based on observations, the developing MSME sector in Langkat Regency is diverse, ranging from ceramic crafts, Purun crafts, furniture crafts, kitchen utensil crafts, various souvenir crafts, processed food crafts, to building materials such as red bricks and tiles. The observation reveals that communities develop MSMEs like a community, where in one area, people form MSMEs that produce the same or similar types of goods. Thus, each area becomes a center for MSMEs for a specific type of product. Almost every sub-district in Langkat Regency has an MSME center, and some sub-districts have more than one MSME center, such as Tanjung Pura sub-district, which serves as the center for Dodol MSMEs and other centers. Besides these industrial centers, there are still many small home industries scattered throughout Langkat Regency. The most numerous home industries are crafts and processed food industries, such as woven mat crafts, bamboo weaving crafts, raw tempe industries, tempe chips industries, and krupuk industries. These industries are small-scale industries scattered across all sub-districts in Langkat Regency. According to observations, more than 50% of these home industries do not have a business permit and are classified as the informal sector (Ratnasari & Kirwani, 2015).

In connection with the above description, managing this MSME program can be a complex task and requires in-depth analysis. By using Machine Learning techniques, effective methods can be developed to classify successful MSME programs and programs that need improvement, with the aim of ensuring that the programs target the right audience. Based on the explanation above, the goal of this research is to build a Classification model for determining appropriate MSMEs to determine programs using Naive Bayes so that the Department can predict the likelihood of making mistakes in choosing the wrong target MSMEs at an early stage. By anticipating these errors. Based on the background explained, it can be proposed "Machine Learning for the Classification of MSME programs at the Department of Industry and Trade of Langkat Regency."

RESEARCH METHODS

The researcher will now explain the method to be used in addressing the research problems. The method employed for classifying SME programs using the Naive Bayes algorithm can be observed in Figure 1.

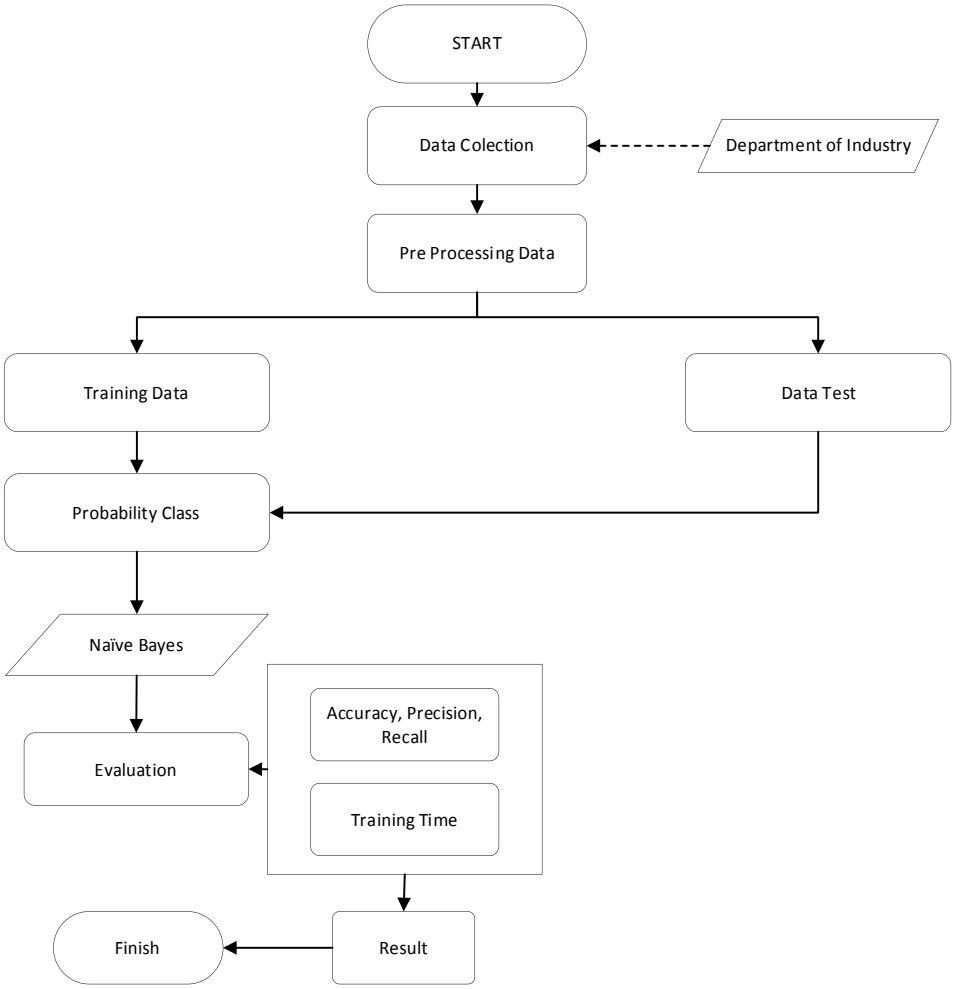


Figure 1. Stages of the Naive Bayes Classifier Algorithm

The following image explains that the research process begins with several steps:

a. Data Preparation

At this stage, data preparation involves processing a set of data from Small and Medium Enterprises (SMEs) obtained from the Office of Industry and Trade of Langkat Regency. The data is split into two subsets: the training set and the testing set, comprising 70% training data and 30% testing data.

b. Data Preprocessing

Commence with proper data preprocessing, including cleaning data from missing or invalid values. Normalize or standardize data to ensure variables undergo proper preprocessing. This involves identifying and handling missing values, dealing with invalid, duplicate, or noisy data, as well as scaling and normalization.

c. Read Training Data

In this stage, the researcher reads the training data, which plays a crucial role in forming the classifier model. The training data is converted based on the dataset obtained from the Department of Industry and Trade of Langkat Regency. Probability Class Calculation At this stage, the researcher calculates the probabilities of each class using the Naive Bayes Classifier (NBC) algorithm.

d. Read Testing Data

In this stage, the researcher reads the testing data, and 30% of business entities are used from the dataset for this testing process.

e. Model Testing:

Naive Bayes:

During Naive Bayes training, you can set parameters such as GaussianNB, which is used for classification. GaussianNB specifically uses a Gaussian (normal) distribution to calculate the probabilities of features for each class.

f. Model Evaluation

At this stage, the researcher performs classification calculations on the testing data using the formula: P is the probability, C is the class, X is the attribute value to be calculated, P(C|X) is the probability value in the class with the same attribute, P(X|C) is the probability value in the attribute with the same class, P(C) is the probability value of the class, and P(X) is the probability value in the attribute.

g. Results

The testing results yield a Confusion Matrix consisting of precision, recall, and accuracy. The calculation of the confusion matrix can be seen in Equations (1), (2), (3).

$$\text{Recall} = \frac{TP}{TP + N} * 100\% \quad \text{..... (1)}$$

$$\text{Accuracy} = \frac{TP+TN}{TP + TN + FP + FN} * 100\% \quad \text{..... (2)}$$

$$\text{Precision} = \frac{TP}{TP + P} * 100\% \quad \text{..... (3)}$$

RESULTS AND DISCUSSION

1. Results

In this chapter, the researcher conducts a study to optimize the allocation of limited resources and budget to support the most potential SME programs for development in Langkat regency using the Naive Bayes Algorithm. The Naive Bayes model is initialized using Gaussian Naive Bayes, suitable for cases where data features are considered as normally distributed random variables. The model is fitted (trained) using training data (X_train and y_train). This means the model learns the probability distribution of features in the given target class. After the model is trained, it is used to make predictions on test data (X_test). The prediction results are stored in the variable y_pred. They are then classified to provide more detailed information, including Precision, Recall, and F1-score for each class, as well as other metrics that can support the research.

2. Discussion

The training process involves the use of pandas, which is a powerful data manipulation library commonly used for working with structured data, such as CSV files or data frames, and the scikit-learn package. Scikit-learn is utilized to split the dataset into training and testing sets, as well as GaussianNB, a variant of the Naive Bayes algorithm that assumes features follow a Gaussian distribution. The libraries used in this research can be observed in Figure 2, as

```
# Menghubungkan Google Colab dengan Google Drive
from google.colab import drive
drive.mount("/content/drive", force_remount=True)

# Impor library yang diperlukan
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, classification_report
```

follows:

Figure 2. Data Collection Packages

The sample used in this study consists of two types, categorized by the type of business and business permits. The total data collected for the sample includes 374 business entities, with each category comprising 298 businesses having operational permits and 76 businesses without permits.

a. Preprocessing Data

Before conducting training and testing, it is essential to understand the data structure by examining its characteristics, including column names, data types, and formats. Identify and handle missing values in the dataset, clean the data by detecting and analyzing duplicate entries, perform data normalization, etc. This process helps expedite the training process and reduce model complexity.

b. Naive Bayes Classification Calculation

The classification calculation involves using a confusion matrix and visualizing prediction results using a pairplot. The confusion matrix is a table used to evaluate the performance of a classification model. Additionally, this code utilizes seaborn to create a pairplot, which illustrates relationships between pairs of features in the data. Different colors are used to distinguish between correct and incorrect predictions, providing a visual insight into how well the model can separate target classes. Color labels are set according to the

```
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix

# Menghitung confusion matrix
cm = confusion_matrix(y_test, y_pred)

# Menampilkan confusion matrix dalam bentuk grafik
plt.figure(figsize=(8, 6))
sns.heatmap(cm, annot=True, fmt="d", cmap="Blues", xticklabels=unique_values_y, yticklabels=unique_values_y)
plt.title('Confusion Matrix')
plt.xlabel('Predicted')
plt.ylabel('True')
plt.show()


# Gabungkan fitur, hasil prediksi, dan target untuk pairplot
df_pred_pairplot = pd.concat([X_test, pd.Series(y_test, name='actual_target'), pd.Series(y_pred, name='predicted_target')], axis=1)

# Buat pairplot
sns.pairplot(df_pred_pairplot, hue='predicted_target')
plt.show()
```

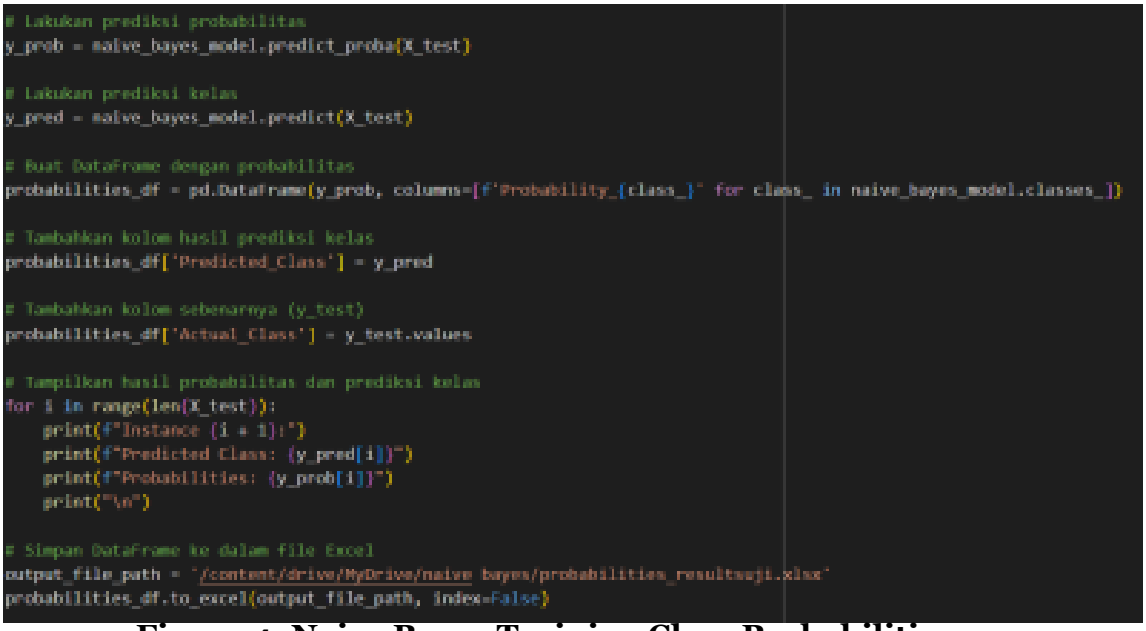
predicted values.

Figure 3. Naive Bayes Classification Division of Training Data and Testing Data for SVM Model

The training data used consists of 250 entries, each of which has been converted, resulting in values for Probability_0, Probability_1, Predicted_Class, and Actual_Class. The testing data comprises 124 entries, also converted, producing values for Probability_0, Probability_1, Predicted_Class, and Actual_Class, respectively.

c. Class Probabilities

Class probabilities for each row in the testing data are calculated using predict_proba. This aids in understanding how the model classifies each row in the data, both individually in the training data and in the testing data. The probability results for each class are stored in a DataFrame, including columns for each class, and the predicted class outcome column is saved to an Excel file. The display of Probability and Prediction results can be seen in Figure 4 as follows:



```
# Lakukan prediksi probabilitas
y_prob = naive_bayes_model.predict_proba(X_test)

# Lakukan prediksi kelas
y_pred = naive_bayes_model.predict(X_test)

# Buat DataFrame dengan probabilitas
probabilities_df = pd.DataFrame(y_prob, columns=[f'Probability_{class_}' for class_ in naive_bayes_model.classes_])

# Tambahkan kolom hasil prediksi kelas
probabilities_df['Predicted_Class'] = y_pred

# Tambahkan kolom sebenarnya (y_test)
probabilities_df['Actual_Class'] = y_test.values

# Tampilkan hasil probabilitas dan prediksi kelas
for i in range(len(X_test)):
    print(f"Instance {i + 1}:")
    print(f"Predicted Class: {y_pred[i]}")
    print(f"Probabilities: {y_prob[i]}")
    print("\n")

# Simpan DataFrame ke dalam file Excel
output_file_path = '/content/drive/MyDrive/naive bayes/probabilities_resultsuji.xlsx'
probabilities_df.to_excel(output_file_path, index=False)
```

Figure 4. Naive Bayes Training Class Probabilities

The explanation of the Naive Bayes training process in the research is as follows:

- Data Separation:
  The data is divided into two main parts: the training set and the test set using the `train_test_split` function. This helps train the model on a subset of the data and test its performance on data it has never seen before.
- Initialization of Naive Bayes Model:
  The Naive Bayes model (Gaussian Naive Bayes) is initialized using the scikit-learn library.
- Model Training:
  The Naive Bayes model is trained using the training data (X_train and y_train) using the `fit` method.
- Prediction of Probabilities and Classes:
  After training, the model is used to make predictions on the test data (X_test). The `predict_proba` function is used to obtain prediction probabilities for each class, and the `predict` function is used to obtain the final predicted class.
- Creation of DataFrame for Probabilities:
  The probability results for each class and predicted class are stored in a DataFrame (`probabilities_df`).
- Evaluation and Understanding of Results:
  Accuracy and classification reports are calculated to evaluate the model's performance.
  The probability results and predicted classes for each instance in the test data are displayed.
  The DataFrame containing probabilities and predicted class results is saved to an Excel file.

This process involves basic steps in Naive Bayes model training, performance evaluation, and saving results for further analysis. By using the Naive Bayes method, the model can learn patterns from the training data and then be used to make predictions on new data.

d. Naive Bayes Model

The evaluation results of the Naive Bayes model in the process are as follows: Accuracy of 78.8% indicates that the Naive Bayes model performs very well in classifying data in the test set. Precision of 82% shows that when the

Naive Bayes model predicts a positive label, 80% of the predictions are correct. Recall of 68% indicates that the model is able to detect all true positive instances. In this case, overall recall is 68%, but weighted recall is higher at 79%. If weighted recall is higher than overall recall, it may indicate an imbalance between classes in the dataset. Weighted recall gives greater attention to classes with many instances or higher weights in the dataset. This can provide a more accurate picture of the model's performance, especially in the presence of class imbalances. The F1 score is a combined measure of precision and recall. It provides a better overview of the balance between the model's ability to predict positive classes and its ability to identify all instances that should be positive. In this case, the overall F1 score is 70%, with a weighted F1 score of 76%.

e. Model Evaluation

Let's review the model evaluation based on the generated process: Good accuracy (78.8%) indicates that the model, overall, is capable of making accurate predictions. High precision (82%) signifies that when the model makes positive predictions, the tendency to be correct is very high. Lower recall (68%) may suggest that the model is less able to comprehensively identify all instances that should be positive. A fairly good F1 score (70%) indicates a relatively good balance between precision and recall.

Recommendations:

If emphasizing the identification of as many positive instances as possible (recall) is crucial, improvements to the model may be necessary to enhance recall. If minimizing false positives is the top priority, adjustments can be made to improve precision.

f. Results

Based on the training of the Naive Bayes model, which experienced increased accuracy, precision, recall, and F1 Score (Accuracy = 78.8%, Precision = 82%, Recall = 79%, F1 Score = 76%), it can be concluded that the model is capable of providing predictions with a fairly high level of accuracy. However, it is important to note that the model's performance may depend on the context and specific needs of the application in question.

CONCLUSION

Conclusion after testing using the Naive Bayes algorithm resulted in good accuracy. After conducting the testing, several conclusions were drawn as follows:

1. Accuracy: The Naive Bayes model achieved an accuracy of approximately 78.8%. This indicates that the model has a fairly good level of accuracy in predicting small to medium-sized industry classes.
2. Precision and Recall: Precision for class 0 is 0.86, meaning that when the model predicts class 0, around 86% of its predictions actually belong to that class. Precision for class 1 is 0.78, indicating that about 78% of the model's predictions for class 1 are correct.
3. Recall (sensitivity) for class 0 is 0.39, meaning the model captures about 39% of the total instances that actually belong to class 0. Recall for class 1 is very high, at 0.97, indicating that the model is excellent at detecting instances that actually belong to class 1.
4. F1-Score: The F1-score is a combined measure of precision and recall. The F1-score for class 0 is 0.54, and for class 1, it is 0.86. A high F1-score indicates that the model has a good balance between precision and recall.
5. Macro and Weighted Average Analysis: Macro avg for precision, recall, and f1-score shows lower values compared to weighted avg. This may indicate an imbalance in the class distribution. Weighted avg assigns greater weight to the majority class, which in this context is class 1 (medium-sized industry).

REFERENCES

Agarwal, S., Jha, B., Kumar, T., Kumar, M., & Ranjan, P. (2019). Hybrid of

Naive Bayes and Gaussian Naive Bayes for Classification: A Map Reduce Approach. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 8(6), 266–268.

Barus, S. P. (2021). Implementation of Naïve Bayes Classifier-based Machine Learning to Predict and Classify New Students at Matana University. Journal of Physics: Conference Series, 1842(1). https://doi.org/10.1088/1742-6596/1842/1/012008

Fitrianah, D., Dwiasnati, S., H, H. H., Baihaqi, K. A., Komputer, I., Informatika, T., & Buana, U. M. (2021). Penerapan Metode Machine Learning untuk Prediksi Nasabah Potensial menggunakan Algoritma Klasifikasi Naïve Bayes. 14(2), 92–99.

Hayami, R., Soni, & Gunawan, I. (2022). Klasifikasi Jamur Menggunakan Algoritma Naïve Bayes. Jurnal CoSciTech (Computer Science and Information Technology), 3(1), 28–33. https://doi.org/10.37859/coscitech.v3i1.3685

Huriah, D. A., Nuris, N. D., Usaha, B., Mining, D., Naive, A., & Dalam, B. (2023). Klasifikasi penerima bantuan sosial umkm menggunakan algoritma naïve bayes. Jurnal Mahasiswa Teknik Informatika, 7(1), 360–365.

Ismail, M., Hassan, N., & Saleh Bafjaish, S. (2020). Journal of Soft Computing and Data Mining Comparative Analysis of Naive Bayesian Techniques in Health-Related for Classification Task. Journal of Soft Computing and Data Mining, 1(2), 1–10.http://penerbit.uthm.edu.my/ojs/index.php/jscdm

Osisanwo, F. Y., Akinsola, J. E. T., Awodele, O., Hinmikaiye, J. O., Olakanmi, O., & Akinjobi, J. (2017). Supervised Machine Learning Algorithms : Classification and Comparison. 48(3), 128–138.

Pratama, R., & Izman Herdiansyah, M. (2023). Prediksi Customer Retention Perusahaan Asuransi Menggunakan Machine Learning. Sistem Informasi Dan Komputer), 12, 96–104.

Ratnasari, A., & Kirwani, D. H. (2015). Peranan Industri Kecil Menengah (Ikm) Dalam Penyerapan Tenaga Kerja Di Kabupaten Ponorogo. Jurnal Pendidikan Ekonomi, 1(3), 11–17.https://ejournal.unesa.ac.id/index.php/jupe/article/view/3625

Roihan, A., Sunarya, P. A., & Rafika, A. S. (2020). Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper. IJCIT (Indonesian Journal on Computer and Information Technology), 5(1), 75–82. https://doi.org/10.31294/ijcit.v5i1.7951

Roihan, A., Sunarya, P. A., & Rafika, A. S. (2020). Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper. IJCIT (Indonesian Journal on Computer and Information Technology), 5(1), 75–82. https://doi.org/10.31294/ijcit.v5i1.7951

Sumpena, J., & Kurnia H., N. (2019). Analisis Prediksi Kelulusan Siswa PKBM Paket C Dengan Metoda Algoritma Naive Bayes. Tedc, 13(2), 127–133. http://ejournal.poltektedc.ac.id/index.php/tedc/article/view/13