SAGA: Journal of Technology and Information Systems
Vol 2, Issue 2, May 2024, Page 234-244
ISSN: 2985-8933 (Media Online)
DOI: 10.58905/SAGA.vol2i2.253

Implementation of K-Nearest Neighbour (KNN) Algorithm and Random Forest Algorithm in Identifying Diabetes

Virly Diranisha[1], Agung Triayudi[2]*, Ratih Titi Komalasari[3]

Informatics Study Program, Faculty of Communication and Information Technology, Universitas Nasional, Jakarta, Indonesia

Author Email: virlydiranisha1771@gmail.com[1], agungtriayudi@civitas.unas.ac.id[2*], ratih.titi@civitas.unas.ac.id

Abstract. Diabetes, one of the noncommunicable diseases (NCDs), is currently a major health threat worldwide. So far, diabetes symptoms have only been diagnosed by people according to known physical characteristics without the support of factual evidence or other medical considerations. With the advancement of technology, it is possible to use algorithms to solve various kinds of problems. One of artificial intelligence (AI), machine learning, concentrates on creating systems that can learn from data. This research uses the K-Nearest Neighbor (KNN) and Random Forest algorithms that can be utilised as testing algorithms to identify diabetes. Classification is done based on training data that has been provided in the dataset. The purpose of this research is to determine the best classification in identifying diabetes with the K-Nearest Neighbor (KNN) algorithm and the Random Forest algorithm and is expected to provide more understanding of the implementation of machine learning models. comparing the two algorithms between the KNN algorithm and the Random Forest algorithm. By dividing the testing data and training data using a ratio of 20%: 80% randomised data 300 times. The results of the accuracy evaluation obtained from the Confusion Matrix show that the Random Forest Algorithm has the best accuracy value of 77%, Precision 89%, Recall 78% and F1-Score 83% with an estimator of 100 trees. While the KNN algorithm obtained accuracy of 73%, Precision 87%, Recall 73% and F1-Score 79% of the value of K = 7. Based on the comparison results of the two algorithms, it shows that the accuracy value obtained is greater than the Random Forest algorithm even though the value obtained is not much different.

Keywords: Classification, Comparison, Diabetes, K-Nearest Neighbours, Random Forest.

1 Introduction

Diabetes, one type of disease that cannot be transmitted between people (NCD), is currently a threat to health worldwide. So far, diabetes symptoms are only diagnosed by people according to known physical characteristics without the support of factual evidence or other medical considerations [1].

Chronic diabetes can cause permanent damage, dysfunction, or failure in many parts of the body such as kidneys, blood vessels, eyes heart and nerves. In fact, it is not uncommon for people with chronic diabetes to undergo amputation due to organ decay. Although the symptoms of diabetes can be detected, health research shows that only a small percentage of sufferers know that they have diabetes. Therefore, to improve the understanding of the signs of diabetes, an optimal and accurate classification model is needed as a diabetes disease prediction, so that people who suffer from diabetes can be predicted earlier.

Classifying diabetes is one of the methods to identify the condition, and can be utilised as a tool in this process. Classification in machine learning is a method that data mining can be used to do. Data mining is one type of data processing due to the rapid growth in data collection and various storage technologies [2].

In the context of classification, machine learning is used to create models that can recognise patterns and make predictions or decisions based on the data provided is one of the common tasks in machine learning. Classification is one of the common activities in this context and can be described as the process of identifying a particular category or class of an object or data [3]. Pada era saat ini, di mana komunikasi data dapat terjadi dengan cepat, banyak faktor yang mendorong peningkatan jumlah data. Bidang kesehatan adalah salah satunya di mana kemajuan teknologi sangat penting. Dalam ranah kesehatan, diperlukan sistem atau perangkat yang mampu melakukan diagnosis atau perkiraan penyakit berdasarkan faktor-faktor khusus. Teknik data mining dapat digunakan untuk memprediksi penyakit dari sejumlah besar informasi yang terhimpun di rumah sakit atau lembaga kesehatan lainnya [4].

With the advancement of technology, it is possible to use algorithms to solve various kinds of problems. One of artificial intelligence (AI), machine learning, concentrates on creating systems that can learn from data. The goal is to give computers the ability to learn and improve their performance automatically without having to be explicitly programmed to perform specific tasks. This includes the development of statistical models and algorithms that enable specialised computers to identify patterns in data, make predictions, and take steps without being explicitly programmed to perform specific tasks [5].

Furthermore, training machine learning models are given training data to learn the patterns in the data. During training, the model is adjusted repeatedly to optimise its performance. Model evaluation, after training, the model is tested on test data to evaluate how strong the model's ability to correctly classify the data is. Several assessment matrices including accuracy, precision, recall, and F1-score can be applied.

In this research, machine learning technology is used with K-Nearest Neighbor (KNN) and Random Forest algorithms to improve the accuracy of the diabetes dataset classification results. These algorithms are able to perform the classification process optimally.

The following are general steps in applying machine learning for classification tasks, namely model selection and in this study KNN and Random Forest models are used, data processing in this study uses a diabetes dataset which will be used as training and testing material for the model. The data division on this dataset consists of training data which is used to provide instructions to the model and test data which is used to evaluate the performance of the model on data that has not been seen before [6].

Research that discusses Machine Learning Models for Detecting Diabetes is compared and results in the K-Nearest Neighbors (KNN) model showing optimal performance with an accuracy rate of 82%, as well as balanced values for Precision, Recall, and F1-Score. If high priority is given to accuracy and balance in predicting positive and negative classes, this model becomes an excellent choice. In addition, good results can also be obtained from the Light Gradient Boosting (LGB) and Random Forest models, with accuracy and F1-Score of around 79%. Despite KNN being the best choice for prediction tasks with consistent performance, LGB and Random Forest models also show good results in certain contexts. Nonetheless, it is important to remember that this research has some limitations, including the size of the dataset used [7].

Application of Naive Bayes and KNN (K-Nearest Neighbor) Algorithms compared in Diabetes Disease Classification. The results showed that each classification had the best results. When the data is cleaned, there is some data that has missing values, so, data replacement is done by using the median and mean to calculate the missing values. K-NN classification has the best accuracy value of 90% from the value of K = 5, compared to the Naive Bayes classification method which has a maximum accuracy of 80% using the median and mean [8].

Thus, the K-Nearest Neighbor (KNN) and Random Forest algorithms can be utilised as testing algorithms to identify diabetes. Classification is done based on the training data that has been provided in the dataset. The purpose of this research is to determine the best classification in identifying diabetes with the K-Nearest Neighbor (KNN) algorithm and the Random Forest algorithm and is expected to provide more understanding of the implementation of machine learning models.

2 Methodology

This study uses data derived from diabetes information obtained from the Kaggle.com website with the Pima Indian Diabetes Dataset which was updated 1 year ago. The amount of data used is 768. The data is divided into 2 for testing datasets and training datasets. Then the model is designed on both algorithms by comparing accuracy, precision, recall, and f1-score to find out which algorithm or method can perform the best classification. Next, enter the model evaluation using Confusion Matrix to find out which is the best performance in the classification process. The flow of programme design can be seen in Image 1.

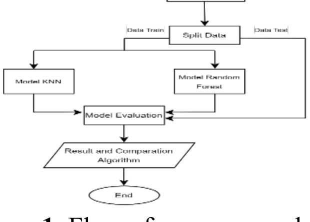
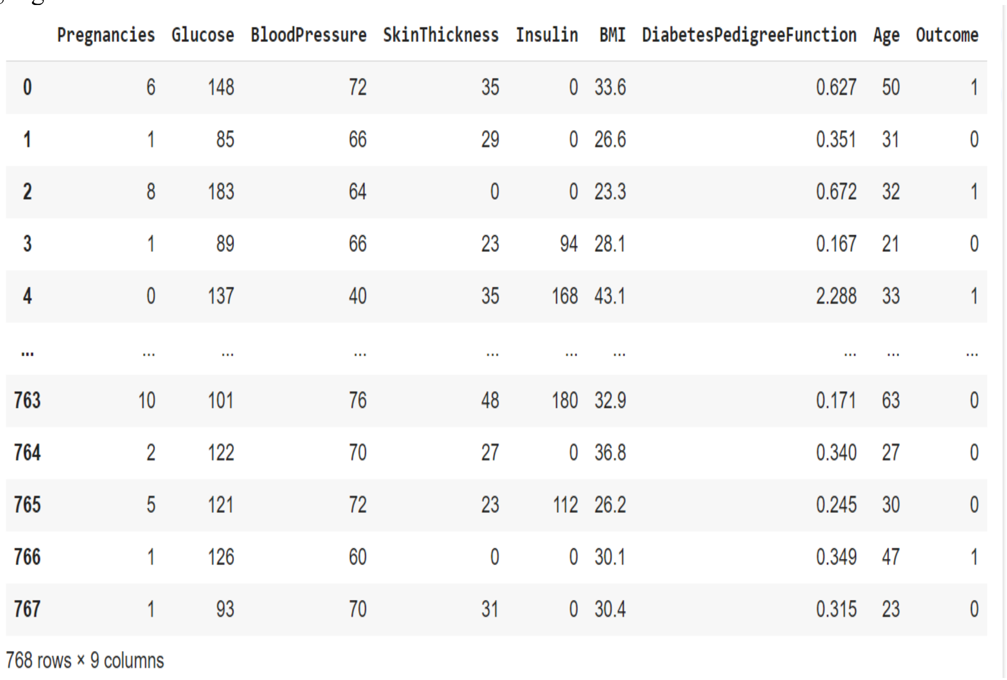


Figure 1. Flow of programme design

2.1 Dataset Retrieval

This research used data from the Kaggle.com website with the Pima Indian Diabetes Dataset which was updated 1 year ago. The amount of data used is 768. Variables that have 8 symptoms and 1 result in identifying diabetes include Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, Age.



	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
...	...	...	...	...	...	...	...	...	...
763	10	101	76	48	180	32.9	0.171	63	0
764	2	122	70	27	0	36.8	0.340	27	0
765	5	121	72	23	112	26.2	0.245	30	0
766	1	126	60	0	0	30.1	0.349	47	1
767	1	93	70	31	0	30.4	0.315	23	0

768 rows × 9 columns

Figure 2. Dataset

Description of the dataset variables used in tables 1 and 2.

Table 1. Independent Variable

No.	Dataset Columns	Description
1.	Pregnancies	How often have you been pregnant
2.	Glucose	Blood sugar concentration at 2 hours after glucose administration
3.	Blood Pressure	Blood pressure that you have
4.	SkinThickness	Thickness of skin folds
5.	Insulin	Insulin levels at 2 hours after glucose administration
6.	BMI (Body Mass Index)	Indicators of ideal weight measurement or not
7.	DiabetesPedigreeFunction	Indicators of family history of diabetes
8.	Age	Current age

Table 2. Dependen Variable

No.	Dataset Columns	Description
1.	Outcome	Shows the result of whether a person has diabetes written with the number 1 or does not have diabetes written with the number 0.

2.2 Data Pre-Processing

In the preprocessing stage, namely by normalising the data first. Data normalisation is a technique that aims to map data to a certain scale in the data mining process. This is important because often the data used in the analysis has a different scale, which can cause problems in comparing or combining data. One of the methods applied in data normalisation is min-max normalisation. This research uses min-max normalisation, which has the following formula:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

Where:
X_norm is the normalised value
X is the original value of the variable
X_min is the minimum value of the variable
X_max is the maximum value of the variable

This normalisation process helps to change the values of the variables so that all values are within the range [0,1]. This is useful to avoid scale issues between variables that may affect the results of the analysis or machine learning model.

2.3 Split Data

Separating the dataset into two: training data and test data. The purpose is to have unknown data points to test the data, rather than testing with the same points as the trained model. Because it helps the model perform much better. The division of training data and test data will be chosen randomly. The training data is used to train the classifier to recognise the characteristics of patients with and without diabetes. The test data is used to test the created classification model and evaluate its performance by comparing the classification results of the model with each data in the original labelled test data.

2.4 K-Nearest Neighbour

KNN is a machine learning technique where data is classified based on the categorical majority of its K nearest neighbours. The KNN algorithm has advantages in the classification of data that is not well-structured or has noise, but also has disadvantages such as the need for distance calculations that can be time-consuming, especially on large datasets. The K value is the number of neighbours considered in the classification process. To classify new data using KNN, the general steps are as follows, enter the training data or training data, the training data label is k and the testing data, then calculate the distance between each each testing data to each training data, Determine k of the training data that is closest to the test data or testing data, Then check the k data labels, then determine the label with the highest frequency, then enter the testing data into the class with the highest frequency.

The KNN algorithm has advantages in the classification of data that is not well structured or has noise, but also has disadvantages such as the need for distance calculations that can take time, especially on large datasets. The accuracy of KNN can also be affected by choosing the right K parameter [9]. The distance between the new data and every other data is measured using the following formula.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (2)$$

2.5 Random Forest

Random forest consists of several decision trees that select class labels through majority voting, or the most classes from the results of each decision tree will be the final result of the random forest, as seen in the following illustration.

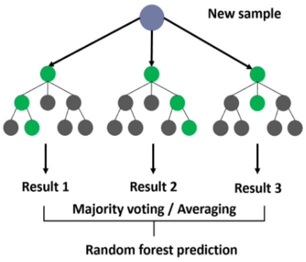


Figure 3. Random Forest

Basically, Random Forest is built from many decision trees. So, the formula is more complex when we consider multiple trees. However, we can understand the basic formation of decision trees in Random Forest:

a. Bootstrapped Sampling each tree in the forest, create a random sample with replacement from the training data.
b. Feature Randomness each node of the decision tree, randomly select a subset of features to consider splitting.
c. Decision Tree Growth, the decision tree is built by splitting the nodes based on the features that provide the best split.
d. Voting or Averaging, in classification tasks, each tree casts a "vote" for a particular class. The class with the most votes is chosen as the final prediction.

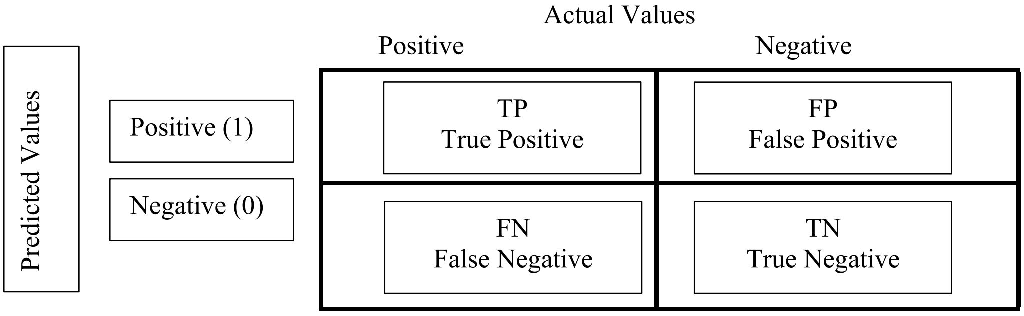
Random Forest works as a set of decision trees that work together to provide better and more stable predictions than a single decision tree. This decision tree consists of several decision tree-like algorithms including ID3 which is based on entropy values and CART which is based on gain values. The conditions are made in the form of branches of the algorithm. The formula of calculating Entropy is shown in equation 3 and the formula of Gain information retrieval is shown in equation 4.

$$\text{Entropy (S)} = \sum_{i=1}^{n} pi * \log 2\ (pi) \qquad (3)$$

$$\text{Gain (S, A)} = \text{Entropy (S)} - \sum_{i=1}^{n} \frac{ISiI}{ISI} * \text{Entropy}\ (Si) \qquad (4)$$

2.6 Model Evaluation

Confusion matrix is a method that can be utilised to assess the performance of a model in classification measured by comparing the predicted results with the original data values. Confusion matrix describes how many incorrect and correct predictions are made by the model in the form of a table that shows the number of correct and incorrect data categorised. True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN) are the four values generated in the confusion matrix table. The following is an illustration of the confusion matrix formula :



Predicted Values	Actual Values: Positive	Actual Values: Negative
Positive (1)	TP True Positive	FP False Positive
Negative (0)	FN False Negative	TN True Negative

Figure 4. Ilustrasi Table Confusion Matrix

Some measurements that can be used to measure classification performance based on the Confusion Matrix in the following equation.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (5)$$

$$Precision = \frac{TP}{TP+FP} \qquad (6)$$

$$Recall = \frac{TP}{TP+FN} \qquad (7)$$

$$F1\ Score = \frac{2 \times Presisi \times Recall}{Presisi + Recall} \qquad (8)$$

3 Results and Discussion

This research utilises the Indian pima diabetes dataset from Kaggle. The data is divided into training data and test data. The ratio was set at 80:20 for 80% training data and 20% testing data.

3.1 Implementation K-Nearest Neighbor

The application of manual calculations for the K-Nearest Neighbor and Random Forest algorithms while for random forest sample data this uses 21 sample data, namely 20 training data and 1 test data.

Table 3. KNN Training Data Sample

No	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes Pedigree Function	Age	Outcome
1	6	148	72	35	0	33,6	0,627	50	1
2	1	85	66	29	0	26,6	0,351	31	0
3	8	183	64	0	0	23,3	0,672	32	1
4	1	89	66	23	94	28,1	0,167	21	0
5	0	137	40	35	168	43,1	2	33	1
6	5	116	74	0	0	25,6	0,201	30	0
7	3	78	50	32	88	31	0,248	26	1
8	10	115	10	0	0	35,3	0,134	29	0
9	2	197	70	45	543	30,5	0,158	53	1
10	8	125	96	0	0	0	0,232	54	1
11	4	110	92	0	0	37,6	0,191	30	0
12	10	168	74	0	0	38	0,537	34	1
13	10	139	80	0	0	27,1	1	57	0
14	1	189	60	23	846	30,1	0,398	59	1
15	5	166	72	19	175	25,8	0,587	51	1
16	7	100	0	0	0	30	0,484	32	1
17	0	118	84	47	230	45,8	0,551	31	1
18	7	107	74	0	0	29,6	0,254	31	1
19	1	103	30	38	83	43,3	0,183	33	0
20	1	115	70	30	96	34,6	0,529	32	1

Table 4. Sample Testing Data

No	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes Pedigree Function	Age	Outcome
1	5	132	50	10	110	40,1	0,245	37	?

Table 5. Training and Testing Data

No	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes Pedigree Function	Age	Outcome
1	6	148	72	35	0	33,6	0,627	50	1
2	1	85	66	29	0	26,6	0,351	31	0
3	8	183	64	0	0	23,3	0,672	32	1

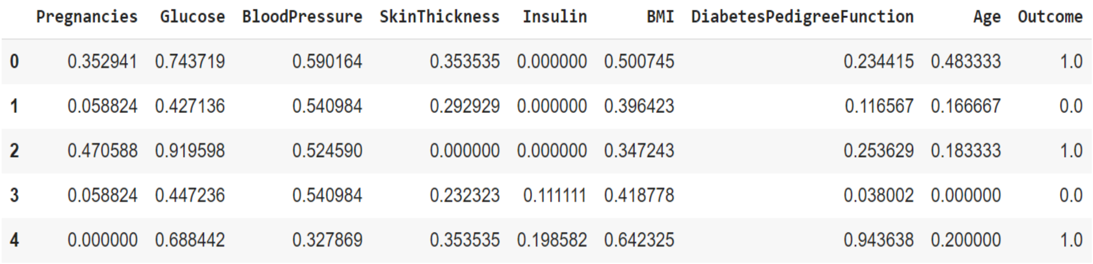
4	1	89	66	23	94	28,1	0,167	21	0
5	0	137	40	35	168	43,1	2	33	1
6	5	116	74	0	0	25,6	0,201	30	0
7	3	78	50	32	88	31	0,248	26	1
8	10	115	10	0	0	35,3	0,134	29	0
9	2	197	70	45	543	30,5	0,158	53	1
10	8	125	96	0	0	0	0,232	54	1
11	4	110	92	0	0	37,6	0,191	30	0
12	10	168	74	0	0	38	0,537	34	1
13	10	139	80	0	0	27,1	1	57	0
14	1	189	60	23	846	30,1	0,398	59	1
15	5	166	72	19	175	25,8	0,587	51	1
16	7	100	0	0	0	30	0,484	32	1
17	0	118	84	47	230	45,8	0,551	31	1
18	7	107	74	0	0	29,6	0,254	31	1
19	1	103	30	38	83	43,3	0,183	33	0
20	1	115	70	30	96	34,6	0,529	32	1
21	5	132	50	10	110	40,1	0,245	37	?

a. Find the value of k = 7
b. Calculate the distance between test data and other training data.
c. Calculating distance using Euclidean Distance calculation is a formula for determining the distance between two points in two-dimensional space. The calculation uses the formula in equation 2.

Table 6. Calculation Result

No	**Distance Calculation Result (euclidean distance)**	**Nearest order**	**Outcome Diabetes**
1	116,9504	9	1
2	123,127	14	0
3	123,7474	15	1
4	54,27712	3	0
5	64,5614	5	1
6	115,2877	7	0
7	63,96726	4	1
8	119,1682	12	0
9	440,1093	19	1
10	127,5579	17	1
11	120,4336	13	0
12	118,7876	11	1
13	117,2366	10	0
14	738,7923	20	1
15	79,66559	6	1
16	125,9169	16	1
17	131,2044	18	1
18	116,4098	8	1
19	52,87952	2	0
20	36,82839	1	1

From the calculation of the testing data determined from the results of the sequence above, it can be seen that the most dominant class is indicated to have diabetes. With 7 nearest neighbours or K = 7, it results in 4 nearest neighbours who have diabetes and 3 neighbours who are not diabetic.



	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	0.352941	0.743719	0.590164	0.353535	0.000000	0.500745	0.234415	0.483333	1.0
1	0.058824	0.427136	0.540984	0.292929	0.000000	0.396423	0.116567	0.166667	0.0
2	0.470588	0.919598	0.524590	0.000000	0.000000	0.347243	0.253629	0.183333	1.0
3	0.058824	0.447236	0.540984	0.232323	0.111111	0.418778	0.038002	0.000000	0.0
4	0.000000	0.688442	0.327869	0.353535	0.198582	0.642325	0.943638	0.200000	1.0

Figure 5. Data Normalisation Results with MinMaxScaler

3.2 Implementation Random Forest

The sample dataset used for manual calculations on the implementation of the Random Forest algorithm is shown in the following table.

Table 7. Random Forest Training Data Sample

No	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes Pedigree Function	Age	Outcome
1	6	148	72	35	0	33,6	0,627	50	1
2	1	85	66	29	0	26,6	0,351	31	0
3	8	183	64	0	0	23,3	0,672	32	1
4	1	89	66	23	94	28,1	0,167	21	0
5	0	137	40	35	168	43,1	2	33	1

Table 6. Bootstrapped Sample Dataset

The following sample dataset is derived from the training dataset which is randomised as Boostrap Sampling

No	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes Pedigree Function	Age	Outcome
2	1	87	66	29	0	26,6	0,351	31	0
1	6	148	72	35	0	33,6	0,627	51	1
4	1	89	66	23	94	28,1	0,167	21	0
3	8	183	64	0	0	23,3	0,672	32	1
5	0	137	40	35	168	43,1	2	33	1

3.3 Building a Decision Tree

For Random Forest classification, it is first necessary to calculate a partition taken from the size of the predictor variables and the number of decision trees to be built. Thus, 100 trees will be formed [10]. Create decision trees by assuming randomly selecting 2 existing variables, for example: Glucose and BloodPressure, SkinThickness and BMI as root node candidates, for example we assume the following.

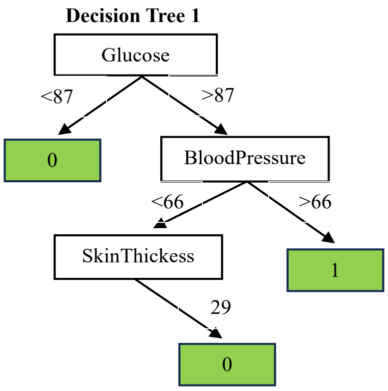


Figure 6. Decision Tree 1

The first tree results in a vote of 0 (no diabetes)

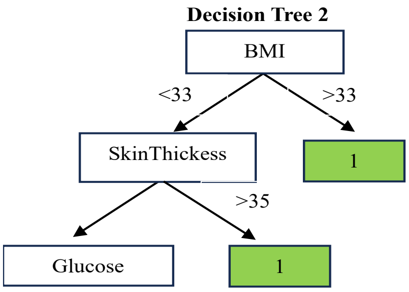


Figure 7. Decision Tree 2

The second tree results in voting 1 (diabetes)

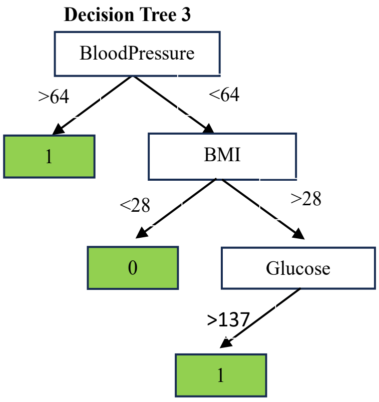


Figure 8. Decision Tree

The third tree results in voting 1 (diabetes)

After obtaining the Random Forest model, the test data is gradually fed into the model.

3.4 Test Evaluation

In this research, evaluation or testing of the KNN method is done by calculating the accuracy value using the confusion matrix method. Performance testing includes calculating accuracy, precision, recall, and f1-score values. The test results are shown in the confusion matrix in table 7.

Table 7. Confusion Matrix of KNN Algorithm Testing Results

	Actual Identified Diabetes	**Actual Unidentified Diabetes**
Predicted Classification Results Identified Diabetes	TP = 88	FP = 13
Predicted Classification Results Not Identified Diabetes	FN = 29	TN = 24

After obtaining the TP, FP, FN, and TN values, the accuracy, precision, recall, f1-score values can be calculated as follows:

$$Accuracy = \frac{88 + 24}{88 + 13 + 29 + 24} \times 100\% = \ 0{,}727 = 73\%$$

$$Precision = \frac{88}{88 + 13} = 0{,}87 = 87\%$$

$$Recall = \frac{88}{88 + 29} = 0{,}73 = 73\%$$

$$F1 - Score = 2 \times \frac{0{,}87 \ \times \ 0{,}73}{0{,}87 + \ 0{,}73} = 0{,}79 = 79\%$$

In this research, evaluation or testing of the Random Forest method is done by calculating the accuracy value of the Random Forest Algorithm using the confusion matrix method. Performance testing includes calculating accuracy, precision, recall, and f1-score values. The test results are shown in the confusion matrix in table 8.

Tabel 8. Confusion Matrix of Random Forest Algorithm Testing Results

	Actual Identified Diabetes	**Actual Unidentified Diabetes**
Predicted Classification Results Identified Diabetes	TP = 90	FP = 11
Predicted Classification Results Not Identified Diabetes	FN = 24	TN = 29

After obtaining the TP, FP, FN, and TN values, the accuracy, precision, recall, f1-score values are calculated as follows:

$$Accuracy = \frac{90 + 29}{90 + 29 + 11 + 24} \times 100\% = 0{,}77 = 77\%$$

$$Precision = \frac{90}{90 + 11} = 0{,}89 = 89\%$$

$$Recall = \frac{90}{90 + 24} = 0{,}78 = 78\%$$

$$F1 - Score = 2 \times \frac{0{,}89 \times 0{,}78}{0{,}89 + 0{,}78} = 0{,}83 = 83\%$$

4 Conclusion

This study compares the two algorithms between the KNN algorithm and the Random Forest Algorithm using Pima Indian Diabetes data by dividing the testing data and training data using a 20% ratio: 80% randomised data 300 times. The results of the accuracy evaluation obtained from the Confusion Matrix show that the Random Forest Algorithm has an accuracy value of 77%, Precision 89%, Recall 78% and F1-Score 83% with an estimator of 100 trees. While the KNN algorithm obtained accuracy of 73%, Precision 87%, Recall 73% and F1-Score 79% of the value of K = 7. Based on the comparison results of the two algorithms show that the accuracy value obtained is greater Random Forest algorithm although the value obtained is not much different.

References

[1] J. Biologi *et al.*, “Diabetes Melitus: Review Etiologi.” [Online]. Available: http://journal.uin-alauddin.ac.id/index.php/psb

[2] Y. Nora Marlim, L. Suryati, and N. Agustina, “Deteksi Dini Penyakit Diabetes Menggunakan Machine Learning dengan Algoritma Logistic Regression,” 2022.

[3] P. R. Sihombing and I. F. Yuliati, “Penerapan Metode Machine Learning dalam Klasifikasi Risiko Kejadian Berat Badan Lahir Rendah di Indonesia,” *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 20, no. 2, pp. 417–426, May 2021, doi: 10.30812/matrik.v20i2.1174.

[4] L. U. Khasanah, Y. N. Nasution, F. Deny, and T. Amijaya, “Klasifikasi Penyakit Diabetes Melitus Menggunakan Algoritma Naïve Bayes Classifier,” vol. 1, no. 1, pp. 41–50, 2022, [Online]. Available: http://jurnal.fmipa.unmul.ac.id/index.php/basis

[5] A. A. A. S. Z. Gustiana. Muttaqin, *Implementasi Artificial Intelligence Dalam Kehidupan*. Aceh: Yayasan Kita Menulis, 2023.

[6] A. Fauzi, A. Heri, and Y. #2, “JEPIN (Jurnal Edukasi dan Penelitian Informatika) Optimasi Algoritma Klasifikasi Naive Bayes, Decision Tree, K-Nearest Neighbor, dan Random Forest menggunakan Algoritma Particle Swarm Optimization pada Diabetes Dataset”.

[7] A. M. Ridwan and G. D. Setyawan, “PERBANDINGAN BERBAGAI MODEL MACHINE LEARNING UNTUK MENDETEKSI DIABETES,” *TEKNOKOM*, vol. 6, no. 2, pp. 127–132, Aug. 2023, doi: 10.31943/teknokom.v6i2.152.

[8] I. L. Faisal, “Perbandingan Metode Naïve Bayes dan KNN (K-Nearest Neighbor) dalam Klasifikasi Penyakit Diabetes,” 2023.

[9] Audrey Athallah, “Prediksi Diabetes Menggunakan Metode KNN,” *Youtube*. 2020.

[10] D. A. Hadi and D. A. N. Sirodj, “Metode Random Forest untuk Klasifikasi Penyakit Diabetes,” *Bandung Conference Series: Statistics*, vol. 3, no. 2, pp. 428–435, Aug. 2023, doi: 10.29313/bcss.v3i2.8354.