**Journal of Applied Informatics and Computing (JAIC)**
Vol.9, No.4, August 2025, pp. 1663~1669
e-ISSN: 2548-6861


# Early Detection of Type 2 Diabetes Using C4.5 Decision Tree Algorithm on Clinical Health Records

**Hani Setiani [1]*, Muhammad Noor Arridho [2]**, Supriyanto [3]***
* Informatika, Universitas Sragen
** Teknologi Rekayasa Logistik, Politeknik Sinar Mas Berau Coal
hani.setiani@unissra.ac.id [1], arridho@polteksimasberau.ac.id [2], supriyanto@unissra.ac.id [3]

**Article Info**

***Article history:***

Received 2025-07-12
Revised 2025-07-21
Accepted 2025-07-30

***Keyword:***

*Classification,*
*Type 2 Diabetes,*
*C4.5 Algorithm.*

**ABSTRACT**

Type 2 Diabetes is a chronic metabolic disorder marked by elevated blood glucose levels. It is the most prevalent form of diabetes in society, commonly triggered by poor lifestyle habits and hereditary factors. If left unmanaged, the disease can lead to serious complications such as hypertension and other chronic conditions. Therefore, early detection plays a critical role in minimizing long-term impacts and promoting healthier behavioral changes. This research focuses on classifying Type 2 Diabetes using clinical data with the C4.5 Decision Tree algorithm. The dataset encompasses attributes including gender, age, height, weight, waist circumference, BMI, systolic and diastolic blood pressure, respiratory rate, and pulse rate. The model was evaluated under two scenarios: without data balancing and after applying the SMOTE technique for balancing. In the first scenario, the best performance was achieved with a training-testing split of 80:20, resulting in an F1 Score of 67.76%. However, the performance varied across different data proportions. In contrast, the second scenario showed more consistent results, with the 60:40 split yielding the highest F1 Score of 66.67%. These findings suggest that SMOTE effectively reduces bias toward the majority class and enhances sensitivity to the minority class. Therefore, data balancing is a crucial step in developing a reliable classification model for Diabetes Mellitus diagnosis.



## I. INTRODUCTION

Diabetes is a disease characterized by increased blood sugar levels [1]. This disease can be caused by various factors, such as an unhealthy lifestyle, genetic factors, exposure to toxins in food, and infections. In addition, diabetes also carries the risk of causing serious complications, including coronary heart disease and ischemic stroke, which are the main causes [2]. Diabetes can affect anyone, regardless of age or gender [3], In clinical practice, the two predominant types of diabetes are Type 1 and Type 2 Diabetes Mellitus (DM) [4]. Type 2 Diabetes Mellitus is the most prevalent form of diabetes, making up around 90% of all diagnosed cases. It is marked by the body's resistance to insulin, disrupting normal blood sugar regulation. Given the serious implications of this condition, researchers are increasingly driven to create machine learning algorithms that can enhance the precision of predictions and classifications related to Type 2 diabetes.

With the capability to identify patterns independently, machine learning enables more precise medical diagnoses [5]. In the medical field, machine learning offers key advantages like faster diagnostic turnaround and reduced operational costs [2].

Multiple studies have explored the use of machine learning techniques with diabetes-related datasets. One such study conducted in 2023 applied the Naive Bayes algorithm across five trials, yielding accuracy rates of 57.38%, 70.27%, 78.26%, 85.71%, and 88.93%. The dataset was compiled from the medical records of Dr. Soekardjo Regional General Hospital, spanning July 2019 to December 2022. It comprised 502 patient samples and 15 key parameters, including age, gender, weight, systolic and diastolic blood pressure, fasting and postprandial blood glucose, HbA1c levels, triglycerides,

LDL, HDL, total cholesterol, urea, and creatinine [6]. That year, another study used the UCI Machine Learning dataset focused on early-stage diabetes, containing 520 records and 17 features (16 predictors and 1 class). Naïve Bayes testing delivered 87.88% accuracy with a 12.12% error rate [7]. In 2024, a study was conducted utilizing a dataset sourced from the Mergangsan Community Health Center in Yogyakarta. This dataset included ten key features: gender, smoking behavior, weight, height, Body Mass Index (BMI), presence of hypertension, age, physical activity level, alcohol intake, and prior history of non-communicable diseases. The researchers developed Deep Neural Network (DNN) models by tuning various parameters, such as the number of layers, neurons, activation functions, learning rates, batch sizes, weight configurations, optimizers, loss functions, training epochs, and bias values. Data was split for training and evaluation using 10-fold cross-validation. The performance metrics from this model yielded an F1-score of 90%, accuracy of 85%, precision of 95%, and recall of 89% [8].

This research employs the Decision Tree algorithm to classify patients diagnosed with Type 2 Diabetes Mellitus. The choice of this method stems from its strength in offering clear, interpretable decision flows and flexibility across diverse datasets. Acting as both the initial and concluding model, the Decision Tree proves effective for delivering a well-organized and easily digestible decision framework. The model construction utilizes the C4.5 algorithm, with experimental testing involving parameters such as data split ratios and minimum gain thresholds. To evaluate the model's performance, a confusion matrix is used—analyzing metrics like accuracy, precision, and recall under various configuration scenarios [9].

## II. METODE

A quantitative method was used in this study to evaluate and apply the Decision Tree C4.5 algorithm for identifying Type 2 Diabetes Mellitus patients, using clinical data and performance metrics such as accuracy.

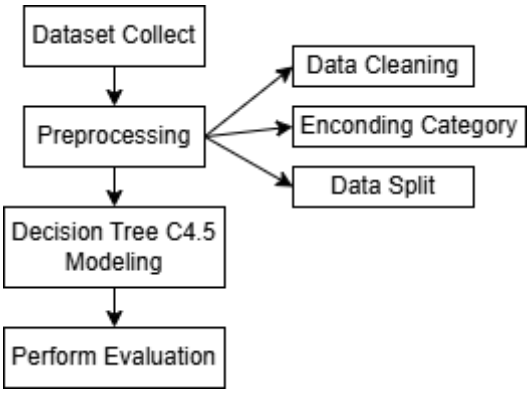


Figure 1. Research Flow (Identification of Type 2 Diabetes Mellitus Patients)

### A. Dataset

This study employed a documentation-based methodology to gather data from medical records maintained by the Community Health Center (Puskesmas) in Bima City, West Nusa Tenggara. The dataset encompasses a six-month period from January to June 2025, comprising 81 patient records. Each entry contains 10 independent variables and one classification target. The dataset was organized using Microsoft Excel (.xlsx), with attributes including gender, age, height (cm), weight (kg), waist circumference (cm), Body Mass Index (BMI), systolic and diastolic blood pressure (mmHg), respiratory rate (breaths per minute), pulse rate (beats per minute), and clinical diagnosis.

TABLE 1
DETERMINING FACTORS OF TYPE 2 DIABETES MELLITUS PATIENTS

| No | Field | Value |
|---|---|---|
| 1 | Sex | Male, Female |
| 2 | Age | < 50, 50 - 59, 60 - 69, ≥ 70 |
| 3 | Height (cm) | < 155, 155 - 160, > 160 |
| 4 | Weight (kg) | <50, 50–59, 60–69, ≥70 |
| 5 | Waist Circumference (cm) | < 80, 80–89, ≥ 90 |
| 6 | BMI (kg/m$^2$) | <18.5, 18.5–22.9, 23–24.9, 25–29.9, ≥30 |
| 7 | Sistolic (mmHg) | < 120, 120–139, 140–159, ≥ 160 |
| 8 | Diastolic (mmHg) | < 80, 80–89, 90–99, ≥ 100 |
| 9 | Respiratory Frequency (/minute) | < 12, 12–20, > 20 |
| 10 | Pulse Rate (/minute) | < 60, 60–100, > 100 |
| 11 | Diagnose | Diabetes, Non Diabetes |

### B. Preprocessing

The preprocessing phase is designed to transform unrefined datasets into structured and analyzable formats, thereby enhancing the effectiveness of classification models used to detect cases of Type 2 Diabetes Mellitus [10]. The preprocessing framework implemented in this study comprises three distinct stages. The initial phase, data cleaning, involves filtering out incomplete or irrelevant records, particularly those containing missing values. The second stage, data encoding, converts categorical variables such as gender and diagnosis into numerical representations to ensure compatibility with machine learning algorithms. Lastly, data sampling is applied to partition the dataset into training and testing sets, allowing for performance evaluation of the constructed classification model [11].

### C. Decision Tree

This algorithm structures decision-making processes into a series of branches, enabling systematic classification based on input features [12]. This tree structure consists of several types of nodes, as follows [13]:

- Root Node: the initial node that represents all data and functions to divide data based on the most informative conditions.

- Internal Nodes: nodes that are at branching points based on testing certain attributes in the dataset. Each branch shows the result of the test.
- Leaf Nodes: the final node that shows the classification results or decisions taken based on the path taken.

Each path from the root node to a leaf node represents a classification or decision rule. The attribute chosen as the root node is generally the one with the highest information value, determined by calculating entropy and information gain. Decision trees operate using a top-down solution-finding approach, That is, when new data is classified, the process is carried out by following a path from the root to the leaf. During this process, the system will calculate the entropy and gain values of each attribute passed through to determine the most appropriate classification path [14].

### *D. C 4.5 Algorithm*

The C4.5 algorithm is used to form a decision tree model. This stage involves sample data as the main input in building the model, as well as test data which is used to measure the level of accuracy of the resulting model [15]. Determining the attributes used as nodes, both at the root node and internal nodes, is done by selecting the attribute that has the highest gain value among all available attributes. The gain value calculation is carried out in stages, starting with calculating the entropy value of each attribute [16]. The formula for calculating entropy is presented in Equation (1).

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

Where $S$ is the set of cases, $pi$ is the proportion of $Si$ to $S$, and $n$ is the number of partitions of $S$. The calculated entropy is used to find the information gain, in the second equation it is used to find the information gain.

After calculating the entropy value, the next step is to calculate the information gain value. The formula used to find the information gain value can be seen in Equation (2).

$$\text{Gain(S, A)} = Entropy\ (S) - \sum_{i=1}^{n} * Entropy\ (Si) \quad (2)$$

Where S is the set of cases, A is the attribute, n is the number of attributes, $|Si|$ is the number of samples for value i and $Entrop(Si)$ is the entropy for samples having value i. After the information gain is calculated using the second equation, the next step is to calculate the split info.

Next, calculate Split Info, which is a measure of how much data is divided due to certain attributes. The formula for calculating split info can be seen in Equation (3).

$$\text{Splitinfo(S, A)} = \sum_{i=1}^{n} * \frac{s_i}{s} log_2 \frac{s_i}{s} \quad (3)$$

Where S is the sample (data) space used for training, A is the attribute, $|Si|$ is the number of samples for value i.

The final calculation in the C4.5 algorithm is the Gain Ratio to select the best attribute. The equation for calculating the gain value can be seen in Equation (4).

$$\text{Gain Ratio(A)} = \frac{Gain\ Information(A)}{Split\ Info(A)} \quad (4)$$

This study uses the Decision Tree C4.5 Algorithm for classification using a dataset of Type 2 Diabetes patients. The stages are as follows:

- The initial stage is preparing the dataset that will be used in C4.5 modeling. Next, the dataset is split into training data and testing data.
- The next stage is build the C4.5 model. At this stage, entropy information gain, split info and gain ratio are calculated to determine the best attributes to be used as nodes in the decision tree structure.
- The final stage is to evaluate the performance of the C4.5 model using evaluation metrics such as accuracy, precision, and recall.

### *E. Evaluation*

Model evaluation was performed using test data that was not involved during the training process, by calculating accuracy, precision, and recall, assessing the algorithm's ability to correctly classify type 2 diabetes and differentiate between the classes contained in the dataset [17].

## III. RESULT AND DISCUSSION

### *A. Decision tree C4.5*

At this stage, the research applied label encoding to convert categorical data into numerical format prior to testing. The data distribution analysis revealed that the Non-Diabetic class accounted for 80.25%, while the Diabetic class represented only 19.75%. The class distribution before applying SMOTE is illustrated in Figure 2.

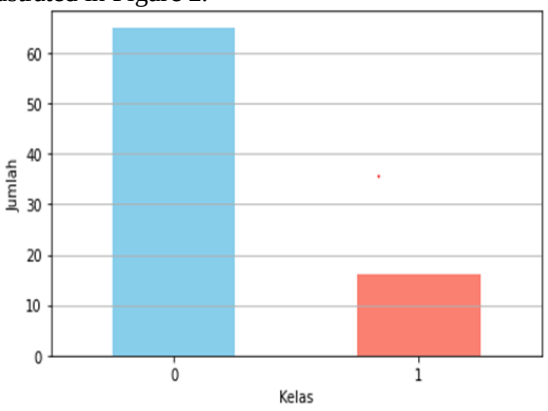


Figure 2. Class distribution before SMOTE

Following the label encoding process, the dataset was immediately subjected to testing without addressing the class imbalance issue. The evaluation was conducted three times, beginning with a 60:40 train-test split, comprising 49 training samples and 32 testing samples. A decision tree model was built using the C4.5 algorithm based on the training data, and

its performance was assessed using the test set. This model achieved an F1 Score of 67.76%. Additionally, calculations were performed for total entropy, Information Gain, Split Information, and Gain Ratio for each attribute, using a minimum gain threshold of 0.01. The study also experimented with alternative data split ratios, such as 70:30 and 80:20. The influence of these variations on model accuracy is presented in Table 2.

TABLE 2
DATA SPLIT BEFORE SMOTE

| No | Split Data | Accuracy | F1 Score |
|---|---|---|---|
| 1 | 60:40 | 54.55% | 57.16% |
| 2 | 70:30 | 60.00% | 62.28% |
| 3 | 80:20 | 64.71% | 67.76% |

Referring to the data in Table 2, the 80:20 data split yielded the highest F1 Score at 67.76%. This result outperformed the 70:30 split, which achieved an F1 Score of 62.28%, showing a difference of 5.48%. Meanwhile, the 60:40 split produced the lowest F1 Score at 57.16%. The lower performance in the 60:40 scenario is likely due to the smaller training set, which may have limited the model's ability to effectively learn data patterns. Although the 80:20 split benefits from a larger training set, the improvement over the 70:30 split is relatively modest, suggesting that the 70:30 ratio may already provide a sufficiently representative dataset for training and testing the model.

```
Rule pohon keputusan:
 |--- Sistole <= 117.00
|   |--- class: 1
|--- Sistole >  117.00
|   |--- Sistole <= 182.00
|   |   |--- Lingkar Perut <= 84.50
|   |   |   |--- Umur  <= 60.50
|   |   |   |   |--- class: 0
|   |   |   |--- Umur  >  60.50
|   |   |   |   |--- Diastole <= 69.50
|   |   |   |   |   |--- class: 1
|   |   |   |   |--- Diastole >  69.50
|   |   |   |   |   |--- Umur  <= 61.50
|   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |--- Umur  >  61.50
|   |   |   |   |   |   |--- class: 0
|   |   |--- Lingkar Perut >  84.50
|   |   |   |--- Sistole <= 131.00
|   |   |   |   |--- Tinggi <= 165.00
|   |   |   |   |   |--- Lingkar Perut <= 93.50
|   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |--- Lingkar Perut >  93.50
|   |   |   |   |   |   |--- class: 0
|   |   |   |   |--- Tinggi >  165.00
|   |   |   |   |   |--- class: 0
|   |   |   |--- Sistole >  131.00
|   |   |   |   |--- class: 0
|   |--- Sistole >  182.00
|   |   |--- class: 1
```

Figure 3. Important Rule Representation of Decision Trees

Figure 3 illustrates the structure of a decision tree built using the C4.5 algorithm to classify data into two categories: class 0 and class 1. This tree leverages medical attributes such as systolic and diastolic blood pressure, height, waist circumference, and age to guide its decisions. Each node represents a rule based on a threshold value of a specific attribute, with branches indicating the direction of decision-making, and leaves showing the final classification result. For instance, if systolic blood pressure is less than or equal to 117, the data is directly classified as class 1. Conversely, if systolic pressure exceeds 117 and waist circumference is less than or equal to 84.50, the data is classified as class 0. In more complex scenarios—such as when systolic pressure is greater than 117, waist circumference exceeds 84.50, and age is less than or equal to 60.50—further decisions are made based on additional attributes like diastolic pressure and height. This tree structure offers transparency in data-driven decision-making and facilitates easier interpretation in the context of health classification.

After identifying the patterns formed by the decision tree structure, this study proceeded with a deeper analysis to assess the prediction quality of the model that achieved the highest accuracy, which was 64.71%. The evaluation was carried out using a confusion matrix to closely examine how the model classified the data. The confusion matrix for the best-performing model is presented in Figure 4.

```
: # Classification report
print("\nEvaluasi Model:")
print(classification_report(y_test, y_pred, digits=4))
```

```
Evaluasi Model:
              precision    recall  f1-score   support

           0     0.8333    0.7143    0.7692        14
           1     0.2000    0.3333    0.2500         3

    accuracy                         0.6471        17
   macro avg     0.5167    0.5238    0.5096        17
weighted avg     0.7216    0.6471    0.6776        17
```

Figure 4. Confusion Matrix C4.5 Results

Based on the classification report, the model achieved an accuracy of 64.71%, indicating that approximately two-thirds of the data were correctly classified. For class 0 (diabetic), the model demonstrated relatively strong performance, with a precision of 0.8333 and a recall of 0.7143. These values suggest that most predictions for diabetic cases were accurate, and the model was able to identify the majority of actual diabetic instances. In contrast, the performance for class 1 (non-diabetic) was considerably lower, with a precision of 0.2000 and a recall of 0.3333, indicating a high rate of misclassification and limited ability to detect non-diabetic cases. The macro-average F1-score of 0.5096 reflects the imbalance in performance across the two classes, while the weighted average F1-score of 0.6776 is influenced by the predominance of diabetic cases in the dataset.

### *B. Decision tree C4.5 Using SMOTE*

Label encoding was first applied to convert categorical data into numerical values for distribution testing. SMOTE was

then used to balance the dataset by augmenting the minority class with synthetic samples, resulting in equal representation of both classes. Post-oversampling, each class—non-diabetic (class 0) and diabetic (class 1) comprised 35 samples. Figure 5 illustrates the balanced class distribution.

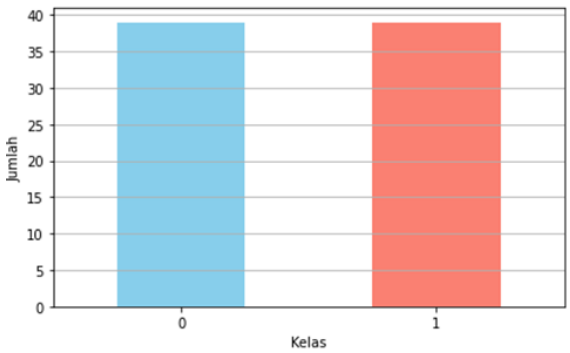


Figure 5. Class distribution after SMOTE

Following the application of SMOTE to balance the dataset, the model was evaluated using two approaches: k-fold cross-validation and train-test split. The k-fold cross-validation method (with 5 folds) was employed to obtain a more general and stable estimate of the model's performance. Meanwhile, the train-test split approach was used to examine the impact of varying data split ratios on model accuracy. In this scenario, three different proportions were tested: 60:40, 70:30, and 80:20. The effect of these variations on model accuracy is presented in Table 3.

TABLE 3
DATA SPLIT AFTER SMOTE

| No | Split Data | Accuracy | F1 Score |
|---|---|---|---|
| 1 | 60:40 | 63.64% | 66.72% |
| 2 | 70:30 | 48.00% | 52.77% |
| 3 | 80:20 | 58.82% | 61.00% |

The evaluation results indicate that the 60:40 data split yielded the most optimal performance, with an accuracy of 63.64% and an F1 Score of 66.67%. In contrast, the 70:30 split produced the lowest performance, with an accuracy of 48.00% and an F1 Score of 52.77%. Meanwhile, the 80:20 split showed moderate results, achieving an accuracy of 58.82% and an F1 Score of 61.00%. These findings suggest that, despite the dataset being balanced using SMOTE, variations in training and testing data proportions still influence model performance. The 60:40 split proved to be the most effective in this context, while increasing the training data in the 80:20 split did not necessarily lead to proportional improvements in accuracy or F1 Score.

Figure 6 illustrates the structure of a decision tree constructed using the C4.5 algorithm to classify data into two categories: class 0 (diabetic) and class 1 (non-diabetic). The model utilizes several medical attributes—including Body Mass Index (BMI), systolic and diastolic blood pressure, gender, weight, height, and age—as the basis for decision-making. Each node in the tree represents a classification rule based on a threshold value of a specific attribute, while branches indicate the direction of the decision path, and leaves display the final classification outcome. For example, if BMI ≤ 20.05, the data is directly classified as class 0 (diabetic). Conversely, if BMI > 20.05 and systolic pressure ≤ 119.50, the data is categorized as class 1 (non-diabetic). In more complex scenarios, such as BMI > 20.05, systolic pressure > 119.50, and female gender > 0.50, the model evaluates additional attributes—such as weight, diastolic pressure, height, and age—before determining the final classification. This tree structure enhances transparency in data-driven decision-making and facilitates interpretation within the context of health assessment.

```
Rule pohon keputusan:
|--- IMT <= 20.05
|   |--- class: 0
|--- IMT >  20.05
|   |--- Sistole <= 119.50
|   |   |--- class: 1
|   |--- Sistole >  119.50
|   |   |--- Jenis Kelamin_Laki-laki <= 0.50
|   |   |   |--- Jenis Kelamin_Perempuan <= 0.50
|   |   |   |   |--- class: 1
|   |   |   |--- Jenis Kelamin_Perempuan >  0.50
|   |   |   |   |--- Umur  <= 51.50
|   |   |   |   |   |--- Berat Badan <= 75.50
|   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |--- Berat Badan >  75.50
|   |   |   |   |   |   |--- Diastole <= 75.00
|   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- Diastole >  75.00
|   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |--- Umur  >  51.50
|   |   |   |   |   |--- IMT <= 23.94
|   |   |   |   |   |   |--- Umur  <= 72.50
|   |   |   |   |   |   |   |--- Tinggi <= 159.50
|   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |--- Tinggi >  159.50
|   |   |   |   |   |   |   |   |--- IMT <= 22.66
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- IMT >  22.66
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |--- Umur  >  72.50
|   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |--- IMT >  23.94
|   |   |   |   |   |   |--- Tinggi <= 156.50
|   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- Tinggi >  156.50
|   |   |   |   |   |   |   |--- class: 1
|   |   |--- Jenis Kelamin_Laki-laki >  0.50
|   |   |   |--- Umur  <= 48.50
|   |   |   |   |--- Diastole <= 77.00
|   |   |   |   |   |--- class: 0
|   |   |   |   |--- Diastole >  77.00
|   |   |   |   |   |--- class: 1
|   |   |   |--- Umur  >  48.50
|   |   |   |   |--- class: 0
```

Figure 6. Important Rule Representation of Decision Trees Using SMOTE

Following the identification of decision tree patterns, a comprehensive evaluation was conducted to assess the predictive performance of the model exhibiting the highest accuracy, measured at 63.64%. To facilitate a detailed analysis of the model's classification capabilities, a confusion matrix was utilized. The corresponding matrix for the optimal model is presented in Figure 7.

```
# Classification report
print("\nEvaluasi Model:")
print(classification_report(y_test, y_pred, digits=4))
```

```
Evaluasi Model:
              precision    recall  f1-score   support

           0     0.8500    0.6538    0.7391        26
           1     0.3077    0.5714    0.4000         7

    accuracy                         0.6364        33
   macro avg     0.5788    0.6126    0.5696        33
weighted avg     0.7350    0.6364    0.6672        33
```

Figure 7. Confusion Matrix C4.5 Using SMOTE Results

According to the classification report, the evaluated classification model enhanced through SMOTE and K-Fold Cross-Validation techniques—achieved an accuracy of 63.64%. For class 0 (diabetes), the model demonstrated satisfactory performance, with a precision of 0.8500 and a recall of 0.6538, indicating that most predictions for this class were correct and most diabetes cases were successfully identified. In contrast, the model's performance for class 1 (non-diabetes) remained suboptimal, with a precision of 0.3077 and a recall of 0.5714, suggesting that a significant portion of non-diabetes cases were not accurately detected, and both false positive and false negative rates were relatively high. The macro-averaged F1-score was 0.5696, while the weighted average F1-score reached 0.6672, largely influenced by the disproportionate number of instances in class 0.

### *C. Model Performance Comparison*

A comparative analysis of both scenarios reveals that the model developed under the second scenario where data balancing was performed using the SMOTE technique exhibited superior and more stable performance, particularly at the 60:40 train-test split, achieving an F1 Score of 66.67%. Although the first scenario with an 80:20 split yielded a slightly higher F1 Score of 67.76%, the noticeable fluctuations across different proportions suggest a strong dependency of the model on the train-test data distribution, likely due to data imbalance. In contrast, the second scenario demonstrated more consistent F1 Scores, indicating that the resampling process via SMOTE effectively mitigated bias toward the majority class and enhanced the model's ability to detect patterns within the minority class. This is further supported by the highest performance observed at the 60:40 split, despite the smaller training datasets compared to the 80:20 configuration. A detailed comparison of both scenarios is presented in Figure 8.

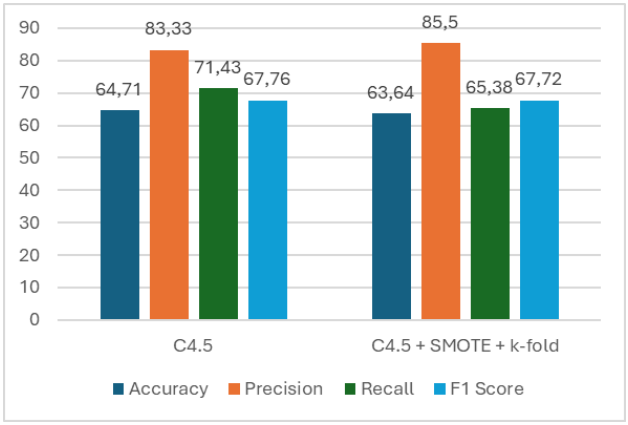


Figure 8. Performance Comparison of Two Classification Methods

## IV. CONCLUSION

This study aimed to develop a classification model for identifying Type 2 Diabetes Mellitus based on clinical attributes, using the C4.5 Decision Tree algorithm. The model was evaluated under two scenarios: one without data balancing and another incorporating the SMOTE technique for balancing. In the first scenario, the highest performance was observed with an 80:20 train-test split, yielding an F1 Score of 67.76%. Conversely, the second scenario demonstrated more stable performance following data resampling via SMOTE, with the 60:40 split achieving the highest F1 Score of 66.67%. These findings suggest that SMOTE effectively mitigates bias toward the majority class and enhances the model's sensitivity to the minority class. Despite the smaller training set in the 60:40 configuration, the model maintained consistent performance. Therefore, data balancing strategies are essential in disease classification tasks such as Diabetes Mellitus, where class distribution is typically skewed. For future work, the implementation of ensemble methods such as Random Forest or Boosting is recommended to further improve model accuracy and its ability to detect underrepresented diabetes cases.

## REFERENCES

[1] Dewi Nasien *et al.*, "Perbandingan Implementasi Machine Learning Menggunakan Metode KNN, Naive Bayes, dan Logistik Regression Untuk Mengklasifikasi Penyakit Diabetes," *JEKIN - J. Tek. Inform.*, vol. 4, no. 1, pp. 10–17, 2024, doi: 10.58794/jekin.v4i1.640.

[2] L. M. Cendani and A. Wibowo, "Perbandingan Metode Ensemble Learning pada Klasifikasi Penyakit Diabetes," *J. Masy. Inform.*, vol. 13, no. 1, pp. 33–44, 2022, doi: 10.14710/jmasif.13.1.42912.

[3] A. P. Silalahi, H. G. Simanullang, and M. I. Hutapea, "Supervised Learning Metode K-Nearest Neighbor Untuk Prediksi Diabetes Pada Wanita," *METHOMIKA J. Manaj. Inform. dan Komputerisasi Akunt.*, vol. 7, no. 1, pp. 144–149, 2023, doi: 10.46880/jmika.vol7no1.pp144-149.

[4] J. Ginting, R. Ginting, and H. Hartono, "Deteksi Dan Prediksi Penyakit Diabetes Melitus Tipe 2 Menggunakan Machine Learning (Scooping Review)," *J. Keperawatan Prior.*, vol. 5, no. 2, pp. 93–105, 2022, doi: 10.34012/jukep.v5i2.2671.

[5] A. M. Ridwan and G. D. Setiawan, "Perbandingan Berbagai Model

Machine Learning Untuk Mendeteksi Diabetes," *Teknokom*, vol. 6, no. 2, pp. 127–132, 2023, doi: 10.31943/teknokom.v6i2.152.

[6] C. A. Rahayu, "Prediksi Penderita Diabetes Menggunakan Metode Naive Bayes," *J. Inform. dan Tek. Elektro Terap.*, vol. 11, no. 3, 2023, doi: 10.23960/jitet.v11i3.3055.

[7] M. Danny and A. Muhidin, "Analisis Prediksi Resiko Diabetes Tahap Awal Menggunakan Algoritma Naive Bayes," *J. Teknol. Inform. dan Komput.*, vol. 9, no. 2, pp. 1443–1459, 2023, doi: 10.37012/jtik.v9i2.2017.

[8] M. Rizky, A. Pramuntadi, W. D. Prastowo, and D. H. Gutama, "Implementasi Metode Deep Neural Network pada Klasifikasi Penyakit Diabetes Melitus Tipe 2," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 3, pp. 1043–1050, 2024, doi: 10.57152/malcom.v4i3.1279.

[9] D. Avianto and A. P. Wibowo, "Pembentukan Pohon Keputusan Untuk Penerima Bantuan Beras Miskin Menggunakan Algoritma Decision Tree C4.5," *Netw. Eng. Res. Oper.*, vol. 9, no. 1, pp. 59–68, 2024, doi: 10.21107/nero.v9i1.28020.

[10] H. Setiani, A. Sunyoto, and A. Nasiri, "Metode Naïve Bayes dan Particle Swarm Optimization untuk Klasifikasi Penyakit Jantung," *Explore*, vol. 12, no. 2, p. 6, 2022, doi: 10.35200/explore.v12i2.566.

[11] H. Setiani and N. Tristanti, "Penerapan Metode Correlated Naive Bayes Untuk Klasifikasi Penyakit Kanker Payudara," *J. Inf. Syst. Informatics Eng.*, vol. 9, no. 1, pp. 18–26, 2025.

[12] Ihsan Zulfahmi, "Analisis Sentimen Aplikasi PLN Mobile Menggunakan Metode Decission Tree," *J. Penelit. Rumpun Ilmu Tek.*, vol. 3, no. 1, pp. 11–21, 2023, doi: 10.55606/juprit.v3i1.3096.

[13] Imam Nawawi and Zaehol Fatah, "Penerapan Decision Trees dalam Mendeteksi Pola Tidur Sehat Berdasarkan Kebiasaan Gaya Hidup," *J. Ilm. Sains Teknol. Dan Inf.*, vol. 2, no. 4, pp. 34–41, 2024, doi: 10.59024/jiti.v2i4.969.

[14] R. N. Sari and I. Purwanto, "Sistem Informasi Geografis Fasilitas Kesehatan di Tuntungan Berbasis Android," *Bull. Comput. Sci. Res.*, vol. 3, no. 3, pp. 257–262, 2023, doi: 10.47065/bulletincsr.v3i3.244.

[15] A. K. Wahyudi, N. Azizah, and H. Saputro, "Data Mining Klasifikasi Kepribadian Siswa Smp Negeri 5 Jepara Menggunakan Metode Decision Tree Algoritma C4.5," *J. Inf. Syst. Comput.*, vol. 2, no. 2, pp. 8–13, 2022, doi: 10.34001/jister.v2i2.392.

[16] J. M. H. Y. Al-Afghoni, Wahyudi Setiawan, and Y. Dwi Putra Negara, "Klasifikasi Jenis Benih Kacang Menggunakan Smote Dan Decision Tree C4.5," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 9, no. 1, pp. 462–469, 2024, doi: 10.36040/jati.v9i1.12366.

[17] N. Qisthi, D. Kasoni, L. Liesnaningsih, and N. Heriyani, "Penerapan Data Mining Untuk Prediksi Stunting Pada Balita Menggunakan Algoritma C4.5," *Insa. Pembang. Sist. Inf. dan Komput.*, vol. 12, no. 2, pp. 18–25, 2024, doi: 10.58217/ipsikom.v12i2.314.