

# Web Design for Stroke Early Detection Using Decision Tree C5.0

Endah Purwanti<sup>1,\*</sup>, Reza Ummam Nor<sup>2</sup>, Soegianto Soelistyono<sup>3</sup>

<sup>1,2,3</sup>Biomedical Engineering, Faculty of Science and Technology, Airlangga University, Surabaya,  
East Java, Indonesia

---

## Abstract

Stroke is a disease that needs serious attention because it can cause disability and even death. According to World Health Organization (WHO) in 2022, stroke is the second leading cause of death and a leading cause of disability in the world. In Indonesia, stroke is the first leading of non-communicable disease proportion according to Riset Kesehatan Dasar in 2018. This study aims to design a web application that can help stroke early detection in a person so that people are more concerned about preventing a stroke. This study used Decision Tree (DT) C5.0 method by utilizing 10 stroke risk factors to analyze the risk of stroke in a person. Decision Tree method can break down complex datasets into several simple rules illustrated by a tree, hence the name Decision Tree. The DT C5.0 is one kind of Decision Tree method that has fast performance in classifying data compared to other methods. Therefore, this study observes how DT C5.0 works in detecting stroke risk. The output of this web application is a statement whether a person has a stroke risk or not. The secondary dataset used for model development totaled 5,109 data consisting of 249 stroke patient data and 4,860 non-stroke patient data. In this study, data balancing and cross validation were carried out so that the performance of the training results model was obtained, namely accuracy 83.54%, precision 78.67%, sensitivity 92.20%, and specificity 74.87%. Furthermore, the performance of the test results model is accuracy 84.42%, precision 79.26%, sensitivity 93.10%, and specificity 75.80%.

**Keywords:** *Stroke; Early Detection; Decision Tree C5.0; Web Application*

---

## 1. Introduction

Stroke is a disease that requires serious attention because of its high prevalence. Globally according to the World Stroke Organization (WSO), in 2019 there are more than 101 million people living with stroke [1]. This makes stroke the second leading cause of death and the leading cause of disability in the world [2]. Meanwhile, in Indonesia, based on the 2018 Riset Kesehatan Dasar (Riskesmas) data, the prevalence of stroke has increased from 7 to 10.9 per mile for residents  $\geq 15$  years [3]. This makes stroke the major cause of death in Indonesia [4].

The impact of a stroke is very serious, there are disability and even death [2]. According to the American Heart Association/American Stroke Association (AHA/ASA) there are two risk factors for stroke, namely controllable and uncontrollable so stroke can be predicted by taking into account the risk factors [5]. If stroke can be predicted early, stroke prevention can be attempted

---

\*Corresponding author. E-mail address: [endah-p-1@fst.unair.ac.id](mailto:endah-p-1@fst.unair.ac.id)

Received: 29 June 2023, Accepted: 11 July 2023 and available online 30 July 2023

<http://dx.doi.org/10.33751/komputasi.v20i2.8265>

by minimizing risk factors that can be controlled. However, public knowledge about stroke risk factors is not good enough where in Muller et al research, only 68% were able to name  $\geq 1$  correct stroke risk factor, and 13% named 4 correct risk factors [6]. So that people have quite difficulty in predicting the risk of stroke in themselves.

As one of solution in helping people to detect the risk of stroke is the development an early detection system for stroke with artificial intelligence [7]. Machine Learning (ML) is part of artificial intelligence which has predictive capabilities from the results of training datasets that have been labeled, thus triggering several studies implementing ML to detect stroke [5]. Making the user interface in the development of a stroke detection system will be more attractive and easier for users.

Several studies have been conducted on stroke risk prediction using machine learning. Research [8] to predict stroke with imbalanced data using a combination of the Random Forest and Deep Neural Network methods resulted in an accuracy of 71.6% [8]. Meanwhile, the research conducted by Sailasya and Kumari in 2021 with a balanced dataset shows a comparison of the Decision Tree method which produces 66% accuracy, 77.5% precision and 77.5% recall, with the Random Forest method which produces 73% accuracy, 72% precision, and 73.5% recall [9]. Thus, the Decision Tree method has the potential to be used in the development of a stroke risk detection system.

The Decision Tree method has several advantages such as being simple to understand, easy to apply, requires little knowledge, able to handle numerical and categorical data, tough, and can handle large datasets [10]. This method has several algorithms such as IDE3, C4.5, CART, and C5.0. Based on the research that has been done for the detection of Chronic Obstructive Pulmonary, the Decision Tree C5.0 algorithm produces a better output score than other Decision Tree algorithms, namely IDE3, C4.5, and CART [11]. Thus, the Decision Tree with the C5.0 algorithm was chosen as the method in this study.

The development of an early detection system for stroke along with its interface has also been carried out. In 2016, Olivia Aulia Nastiti developed a stroke classification system using the Naïve Bayes Classifier and Certainty Factor methods based on Delphi software on a Personal Computer (PC) [12]. The product is less flexible because it can only be used by installing software on a PC. Furthermore, in 2020, Hasanah et al developed an android-based application using the Decision Tree C4.5 method and Adaptive Boosting [13]. Android applications are more flexible than PC software because they can be embedded in smartphones. However, the application cannot be used if the user does not have a smartphone or the smartphone's internal storage is lacking. Therefore, there is another option for the early stroke detection system interface, namely with a web application because it is a central platform that can be accessed anywhere and anytime [14].

The purpose of this study is to build a system that can help someone in early detection of stroke. The web application created will later accommodate stroke risk factor information entered by the user. Furthermore, the data is processed with a classification model to produce an output in the form of the possibility that the user has a stroke risk or not. The research that has been carried out has the advantage of handling imbalanced data in a fairly large amount of data and methods that have the opportunity to produce good performance based on the literature. To find out the performance of the system that was built, an evaluation was carried out for both the classification model and the web application.

## 2. Methods

The research material is secondary datasets of stroke medical records which are open access from Kaggle with the link: <https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset> [15]. The dataset contains 12 columns (attribute); 5,109 number of rows (data); 249 number of stroke data; and 4,860 number of non-stroke data.

At the first stage, data preprocessing is carried out to fix the raw dataset so that it is ready for next processing. The process is carried out eliminating the id column, eliminating rows that contain missing values (N/A, unknown, and other), changing the name of the category "yes" and "no" to "1" and "0", and categorizing attributes with numeric values The age attribute is grouped according to the age category released by the Indonesian Health Ministry [16],

**Table 1.** Stroke Risk Prediction Parameters from the Raw Dataset

Attribute	Category
Id	{67, ..., 72940}
Gender	{ <i>male, female, other</i> }
Age	{0.08, ..., 82.00}
Hypertension	{0, 1}
Heart disease	{0, 1}
Ever married	{ <i>no, yes</i> }
Work type children, govt job, never worked, private, self- employed Residence type	{ <i>rural, urban</i> }
Avg glucose level	{55.12, ..., 271.74}
BMI	{10.3, ..., 97.6, <i>N/A</i> }
Smoking status smokes, formerly smoked, never smoked, unknown	{0, 1}
Stroke	

- a. Baby :  $\leq 1$  years
- b. Toddler : 1 – 5 years
- c. Children : 6 – 10 years
- d. Teenager : 11 – 19 years
- e. Adult : 20 – 44 years
- f. Pre Aged : 45 – 59 years
- g. Aged :  $\geq 60$  years

The avg glucose level attribute is the result of checking blood sugar for a diabetes diagnosis. Therefore, the value is categorized according to Indonesian Health Ministry guidelines [17] to be,

- a. 1 : Diabetes (avg glucose level  $\geq 200$  mg/dl)
- b. 0 : No diabetes (avg glucose level  $< 200$  mg/dl)

The BMI attribute indicates a person's obesity status. Thus, the BMI value is categorized according to the threshold released by Indonesian Health Ministry [18], namely:

1. Thin :  $\leq 18.4$
2. Normal : 18.5 – 25.0
3. Obesity :  $\geq 25.1$

The second stage, balancing data is used because the total stroke class data is 4.87% compared to the non-stroke class data which is 95.13% so that the dataset is imbalance. An imbalanced dataset will result a poor classification model. The model will focus on majority data so it is good for predicting the majority class but not good for predicting the minority class. The Google Colaboratory service with the Imblearn package is used for dataset balancing. There are 2 techniques used in balancing the dataset,

1. Random Over Sampling (ROS) technique: Selecting data randomly from the minority class and then adding it to the dataset so that the amount of minority class data is the same as the majority class.

2. Random Under Sampling (RUS) technique: Several majority class data are discarded randomly until the majority class data is the same as the minority class.

The third stage is dividing the dataset into training, validation, and testing group. In this step K-Fold cross validation method is used. The fourth stage is developing Decision Tree model with the C5.0 algorithm. There are three important variables,

- a. Entropy

$$Entropy(S) = - \sum_{j=1}^k p_j \log_2 p_j \quad (1)$$

S is set of data which the entropy value will be looking for, k is the number of classes in the data, and  $p_j$  is the proportion of  $S_j$  to S. The entropy value is used to determine the impurity of the data. The entropy value has a range of 0 to 1. The closer entropy value is to 1, the greater the impurity of the data.

- b. Gain

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^m \frac{|S_i|}{S} \times Entropy(S_i) \quad (2)$$

A is the attribute being processed, m is the number of categories in attribute A,  $S_i$  is the set of cases of the attribute A category,  $|S_i|$  is the number of cases of category  $i$ , and S is the number of cases in S. The greater the gain value, the greater the role of the attributes in determining the class of a data.

- c. Gain Ratio

$$GainRatio = \frac{Gain(S, A)}{SplitInfo(S, A)} \quad (3)$$

Gain (S, A) is the gain value of attribute A and SplitInfo(S,A) is obtained from the following equation,

$$SplitInfo(S, A) = - \sum_{i=1}^c \frac{S_i}{S} \log_2 \frac{S_i}{S} \quad (4)$$

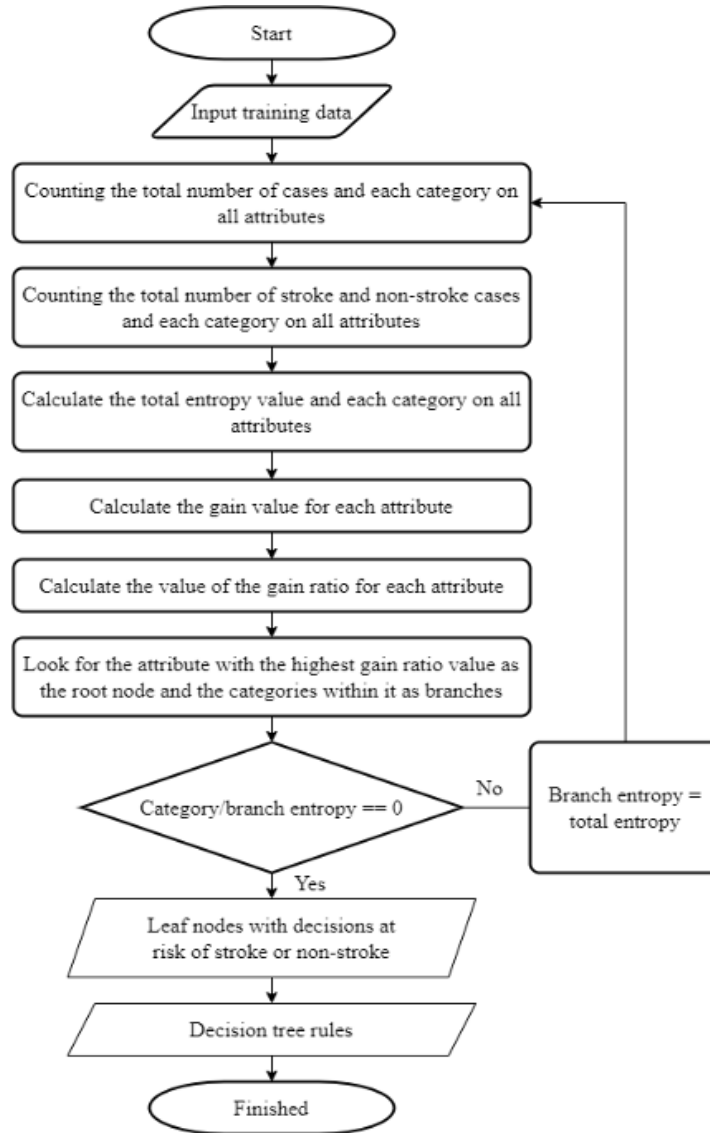
Gain ratio gives more meaning when determining the most influential attribute than just gain. Attributes with fewer categories will be prioritized than attributes with higher gain but have more categories.

After building the decision tree, the fifth step is to test it to find the best model for web application. The decision tree resulting from the design of the model represents the rules for classifying the risk of stroke. The model will be tested on validation data. The prediction results of the classification model will be compared with the actual results using the confusion matrix.

**Table 2.** Confusion matrix

		Predictions	
		Stroke	Non-Stoke
True Value	Stroke	TP (True Positif)	FN (False Negatif)
	Non-Stoke	FP (False Positif)	TN (True Negatif)

Furthermore, the model performance parameters used are accuracy, precision, sensitivity, and specificity. Accuracy serves to see a comparison between correct predictions compared to all actual data. Then precision to see a comparison of the correct stroke predictions compared to all stroke predictions. Furthermore, the sensitivity is to see a comparison of correct stroke predictions compared to all correct stroke data. Finally, the specificity value is to see a comparison of predictions that are not correct compared to all data that is not correct.



**Figure 1.** Flowchart for making the Decision Tree C5.0 model

To get the model performance parameter values, the TP, TN, FP, and FN scores are processed with the following equation,

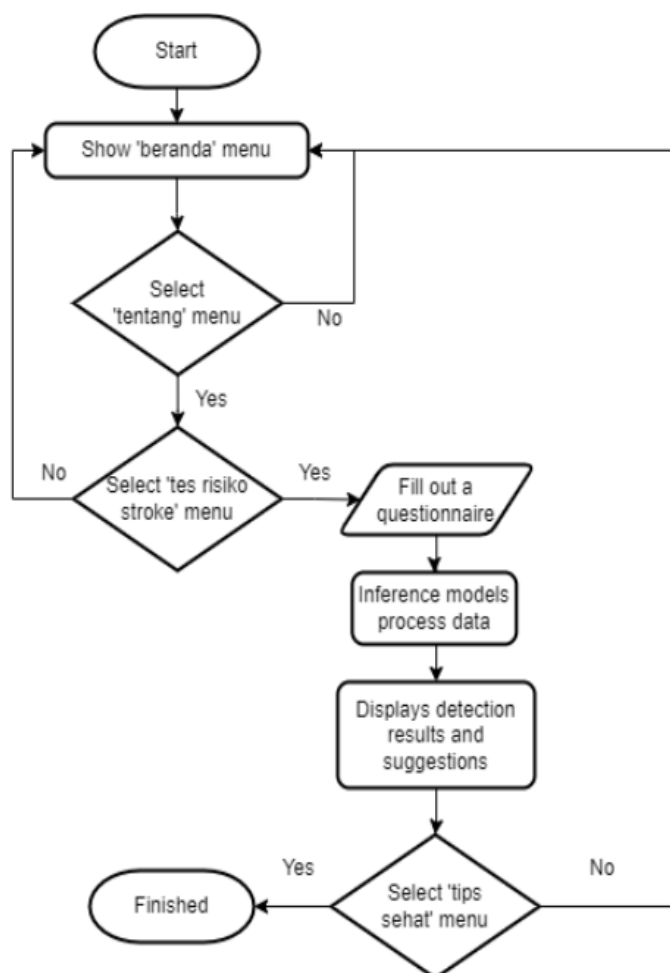
$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \times 10 \quad (5)$$

$$Precision = \frac{TP}{(TP + FP)} \times 100\% \quad (6)$$

$$Sensitivity = \frac{TP}{(TP + FN)} \times 100\% \quad (7)$$

$$Specificity = \frac{TN}{(TN + FP)} \times 100\% \quad (8)$$

After best model was found, then sixth step is building web application with utilizes Visual Code Studio software as a code editor. The programming language used is HTML and CSS to build the appearance of web applications. Besides that, it also uses JavaScript to implement a classification model in web applications



**Figure 2.** Flowchart of web application development

In testing this web application for the seventh step is using white box testing, black box testing, and ISO/IEC 25010 methods as a benchmark for website performance.

- a. White box testing is a method that tests the internal structure, design, and associated software program code. Testing was carried out by researchers as developers. In white box testing, each decision tree rule will be tested through the web application interface whether it produces the appropriate decision or not.
- b. Black box testing is a method that tests software functionality without knowledge of implementation details and program code. Black box testing was tested on 30 respondents as web application users.
- c. ISO/IEC 25010 is the latest standard from the International Organization for Standardization and the International Electrotechnical Commission as a benchmark for software quality analysis that is relevant for evaluating information systems [19]. The following are the aspects selected to assess the quality of web applications,
  - (a) Performance efficiency  
Performance testing is conducted to ensure the system can handle extreme loads without unacceptable degradation of operability. For the testing process using the Gtmetrix web tool software to find out the page load speed, performance, structure, and grade values.
  - (b) Portability  
Portability testing aims to find errors in unique host configurations. This test was

carried out by running web applications on 3 types of browsers that are often used in Indonesia, namely Chrome, Firefox, and Microsoft Edge. Next, evaluate whether each feature of the web application can appear and run well on all types of browsers.

(c) Usability

Testing is used to evaluate the degree to which a user can interact effectively with the website application and the extent to which the website application acts. For the testing process using a questionnaire containing statements and respondents giving an assessment with a score on a scale of 1-4. The number of respondents is 30 people to produce a valid score. The score results identify the usability value according to the guidelines in Figure 2. While the equation for calculating the usability score is presented in Equation 9.

$$UsabilitySkor = \frac{\sum_{a=1}^b (\sum_{p=1}^q skor_{pertanyaan_p}) \times \frac{100}{4a}}{(b)} \quad (9)$$

**Table 3.** Usability Aspect Testing Form

No.	Statement	Strongly Disagree	Don't agree	Agree	Strongly agree
1.	You will be using web applications a lot	Score 1	Score 2	Score 3	Score 4
2.	You judge the web application as useful (according to the objective claim)	Score 1	Score 2	Score 3	Score 4
3.	You rated the web application as easy to navigate	Score 1	Score 2	Score 3	Score 4
4.	You don't need technical assistance to use the web application	Score 1	Score 2	Score 3	Score 4
5.	You assess the functionality of the features provided by a well designed and prepared web application.	Score 1	Score 2	Score 3	Score 4
6.	You rate the consistency of the web application as good	Score 1	Score 2	Score 3	Score 4
7.	You judge how many people can use web applications quickly	Score 1	Score 2	Score 3	Score 4
8.	You rate the web application as uncomplicated to use.	Score 1	Score 2	Score 3	Score 4
9.	You feel less pressure when using web applications.	Score 1	Score 2	Score 3	Score 4
10.	You don't think you need to learn a lot before using a web application.	Score 1	Score 2	Score 3	Score 4

The last stage is evaluating Decision Tree model that used in web application with testing data so that this study will get the testing score for accuracy, precision, specification, and sensitivity.

### 3. Result and Discussion

The result of preprocessing data gets new dataset that contain 11 columns (attribute); 3,736 number of rows (data); 180 number of stroke data; and 3,556 number of non-stroke data. The dataset parameter is on Table 4

**Table 4.** Stroke Prediction Parameters from Data Preprocessing Results

Attribute	Category
Gender	female, male
Age	baby, toddler, children, teenager, adult, pre aged, aged
Hypertension	{0,1}
Heart Disease	{0,1}
Marital Status	{0,1}
Occupation	private, self-employed, not yet working, govt job, never worked
Residence Type	urban, rural
Diabetes	{0,1}
Obesity Category	thin, normal, obesity
Smoking status	smokes, formerly smoked, never smoked
Stroke	{0,1}

Then, the result of balancing data is on Table 5 that shows the number of data before and after balancing with the two methods. The balancing method used is Random Over Sampling (ROS) and Random Under Sampling (RUS). Based on the table, the number of stroke data using the ROS method is increasing. While non-stroke data with the RUS method has decreased.

**Table 5.** Comparison of Balancing Data Results

Description	No balancing	ROS technique	RUS technique
Number of rows (data)	3,736	7,112	360
Number of stroke data	180	3,556	180
Number of non-stroke data	3,556	3,556	180

Result of splitting dataset is on Table 6 Based on Table 6, each dataset (without balancing, ROS, and RUS) will be divided into two groups with a percentage of 80% for the model group and 20% for the test group. The model group was further divided into two groups, namely 90% training and 10% validation. This separation process is necessary for machine learning methods to observe method performance.

**Table 6.** Comparison of Split Dataset Results

Description	No balancing			ROS technique			RUS technique		
	model		test	model		test	model		test
	train	validation		train	validation		train	validation	
Number of rows (data)	2,691	299	746	5,118	569	1,425	264	29	67
Number of stroke data	134	15	31	2,561	285	710	130	14	36
Number of non-stroke data	2,557	284	715	2,557	284	715	134	15	31

In accordance with the K-fold cross validation method, the distribution of training data and validation data will be iterated 10 times so that 10 different training and validation data packages



are produced. Because there are 3 datasets used (without balancing, ROS, and RUS) there are a total of 30 training data packages which will then be used for building a classification model.

Each dataset resulting from data separation will be made into a decision tree model. After that, each model will be tested using validation data. Comparison of results between models without balancing, ROS, and RUS is shown in Table 7

**Table 7.** Comparison Performance Model

	Model Performance			
	Accuracy	Precision	Sensitivity	Specificity
<b>No Balancing</b>				
Average	94.01%	1.67%	0.67%	98.91%
Standard deviation	0.67%	5.00%	2.00%	0.69%
<b>ROS</b>				
Average	83.54%	78.67%	92.20%	74.87%
Standard deviation	1.76%	2.06%	2.51%	3.21%
<b>RUS</b>				
Average	65.66%	68.24%	61.24%	70.62%
Standard deviation	5.13%	9.80%	11.31%	13.65%

Based on table 3.5 the best result is the decision tree model for the dataset by balancing the ROS data method. After that look for the best fold based on Table 8

The best model for the ROS dataset is at fold-8. The values of all performance parameters are fairly uniform. This is because the dataset has been balanced so that the classification model can predict well for both stroke and non-stroke classes.

The web application resulting from this research is named 'Stroke Risk Prediction' which has 4 menus namely 'beranda', 'tentang', 'tes risiko stroke', and 'tips sehat'. The web application is deployed by utilizing the GitHub service so that it can be accessed at the link: <https://rezaummamnor.github.io/StrokePredictionRezaUmmam/>. While the complete documentation of the web application program "Stroke Risk Prediction" can be accessed at the link: <https://github.com/rezaummamnor/StrokePredictionRezaUmmam>

Then the web application test results are displayed in the Table 9, 10, 11 and 12 within the explanation

The results of the White Box test show that all 334 ROS classification model decision tree rules are in line with expectations. Thus, the web application "Stroke Risk Prediction" from the internal side of the software and code is good.

**Table 8.** Cross Validation of The ROS Dataset Model

k-Fold	Model Performance			
	Accuracy	Precision	Sensitivity	Specificity
1	81.72%	77.01%	90.53%	72.89%
2	83.83%	80.44%	89.47%	78.17%
3	82.95%	80.52%	87.02%	78.87%
4	84.89%	80.06%	92.98%	76.76%
5	80.67%	74.93%	92.28%	69.01%
6	85.06%	80.49%	92.63%	77.46%
7	81.34%	75.87%	91.90%	70.77%
8	86.09%	80.42%	95.42%	76.76%
9	85.39%	79.65%	95.07%	75.70%
10	83.48%	77.30%	94.72%	72.28%
Average	83.54%	78.67%	92.20%	74.87%
Standard deviation	1.76%	2.06%	2.51%	3.21%

**Table 9.** White Box Test Results

Model Rule Testing Results	Number of rules
Suitable	334
Not Suitable	0

**Table 10.** Black Cox Test Results

Web Page	Number of Respondents	
	Suitable	Not Suitable
Beranda	30	0
Tentang	30	0
Tes Risiko Stroke	30	0
Hasil Risiko Stroke	30	0
Tips Sehat	30	0

Based on the results of the Black Box test, all respondents agreed that all the features of the "Stroke Risk Prediction" web application were fit for purpose. So, from the Black Box testing it can be concluded that the functionality of the web application 'Stroke Risk Prediction' is as planned.

**Table 11.** Performance Efficiency Testing Result

No.	Page	Page Load (second)	Performance	Structure	Grade
1.	Beranda	1.50	97.00%	97.00%	A
2.	Tentang	0.80	93.00%	95.00%	A
3.	Tes Risiko Stroke	1.10	97.00%	98.00%	A
4.	Hasil Risiko Stroke	0.70	98.00%	95.00%	A
5.	Tips Sehat	0.80	97.00%	98.00%	A
	Average	0.98	96.40%	96.60%	A

**Table 12.** Portability Aspect Testing

No.	Page	Browser		
		Chrome	Mozilla Firefox	Microsoft Edge
1.	Beranda	v	v	v
2.	Tentang	v	v	v
3.	Tes Risiko Stroke	v	v	v
4.	Hasil Risiko Stroke	v	v	v
5.	Tips Sehat	v	v	v

Note:

- v: Pages can appear and run well
- × : The page cannot be displayed/not working properly

Based on the portability test results, the 'Stroke Risk Prediction' web application can be used properly on 3 types of browsers namely Chrome, Mozilla Firefox, and Microsoft edge Then, Based on the usability test results, the web application 'Stroke Risk Prediction' obtained a score of 84.67 so that it is an acceptable web application with a grade scale of 'B' and an adjective scale of 'excellent'

**Table 13.** Performance Classification Models on Test Data

Data Balancing Techniques	Accuracy	Precision	Sensitivity	Specificity
ROS	84.42%	79.26%	93.10%	75.80%

Based on table 3.10, the classification model with the best performance is the Random Over Sampling (ROS) dataset model. Thus, the model is applied to web applications to predict the risk of stroke. The complete decision tree and rule table (334 rules) for the ROS dataset classification model can be found on the GitHub repository with the link:

<https://github.com/rezaummamnor/StrokePredictionRezaUmmam>

The best model analysis can be seen from the model performance scores, namely accuracy of 84.42%, precision of 79.26%, sensitivity of 93.10%, and specificity of 75.80%. This score is good enough for machine learning models. However, for medical cases the performance score still needs to be improved. The accuracy shows that the model can predict quite well the risk class of stroke/no stroke overall. Precision indicates the ability of the model to correctly predict the stroke risk class by considering the incorrect positive prediction of stroke risk. Sensitivity indicates the ability of the model to correctly predict stroke risk class without considering the positive stroke risk prediction incorrectly. Specificity indicates the ability of the model to correctly predict the no-stroke risk class compared to all data labeled as no-stroke.

The precision and specificity scores are still below 80%, which is possible due to the weakness of the Decision Tree method which cannot overcome overlapping cases. This case occurs when two or more attributes have the same highest gain ratio value. In addition, there are conditions where the data for an attribute is not yet homogeneous but all 10 stroke risk factors have been used. These two conditions force the attribute to become a leaf node with a decision based on the proportion of stroke/no stroke data. As a result, an error value arises.

Analysis was also carried out on the order of priority of stroke risk factors in determining stroke risk or not stroke based on the model obtained. Here is the order

1. Age Category
2. Occupation
3. Hypertension
4. Obesity Category
5. Heart Disease
6. Marital Status
7. Smoking Status
8. Diabetes
9. Residence Type
10. Gender

#### 4. Conclusion

The best model is obtained from balancing the dataset with the Random Over Sampling technique and validated with the K fold cross validation method. Then the performance of the Decision Tree C5.0 classification model resulting from training with cross validation was 83.54% accuracy, 78.67% precision, 92.20% sensitivity, and 74.87% specificity. Furthermore, the performance of the test results model is 84.42% accuracy, 79.26% precision, 93.10% sensitivity, and 75.80% specificity. The results of testing the performance of web applications are that the internal code structure (white box testing) and functionality (black box testing) are appropriate. As for the quality of web

applications, performance efficiency scores 0.98 seconds page load, performance 96.4%, structure 96.6%, and grade A. The portability test states that the web application can run well on 3 different browsers. Usability testing got a score of 84.67 making it acceptable with a grade scale of 'B' and an adjective scale of 'excellent'.

## References

- [1] WSO, "Global Stroke Fact Sheet 2022," 2022. Accessed: Aug. 18, 2022. [Online]. Available: <https://www.world-stroke.org/assets/downloads/WSOGlobalStrokeFactSheet.pdf>
- [2] WHO, "World Stroke Day 2022," *World Health Organization*, Oct. 29, 2022. <https://www.who.int/srilanka/news/detail/29-10-2022-world-stroke-day-2022> (accessed Jan. 10, 2023).
- [3] Kemenkes RI, "Hasil Utama RISKESDAS 2018," 2018. Accessed: Aug. 18, 2022. [Online]. Available: [https://kesmas.kemkes.go.id/assets/upload/dir\\_519d41d8cd98f00/files/Hasil-riskesdas-2018\\_1274.pdf](https://kesmas.kemkes.go.id/assets/upload/dir_519d41d8cd98f00/files/Hasil-riskesdas-2018_1274.pdf)
- [4] Venketasubramanian N, Yudiarto FL, Tugasworo D. Stroke Burden and Stroke Services in Indonesia. *Cerebrovasc Dis Extra*. 2022;12(1):53-57. doi: 10.1159/000524161 .
- [5] A. K. Boehme, C. Esenwa, M. S. v Elkind, M. Fisher, C. Iadecola, and R. Sacco, "Stroke Risk Factors, Genetics, and Prevention Stroke Risk Factors, Genetics, and Prevention," *Stroke Compendium*, pp. 472–495, 2017, doi: 10.1161/CIRCRESAHA.116.308398.
- [6] Muller-Nordhorn, J, Nolte, CH, Rossnagel, K, Jungehulsing, G, Reich, A, Roll, S, et al. Knowledge about risk factors for stroke: a population-based survey with 28,090 participants. *Stroke*. (2006) 37:946–50. doi: 10.1161/01.STR.0000209332.96513.82.
- [7] Mohammad, Sikender M. Artificial Intelligence In Information Technology. *International Journal Of Innovations In Engineering Research And Technology [IJERT]*. 2020;7(6):168-175. <http://dx.doi.org/10.2139/ssrn.3625444>
- [8] T. Liu, W. Fan, and C. Wu, "A hybrid machine learning approach to cerebral stroke prediction based on imbalanced medical dataset," *Artif Intell Med*, vol. 101, 2019, doi: 10.1016/j.artmed.2019.101723.
- [9] G. Sailasya and G. L. A. Kumari, "Analyzing the Performance of Stroke Prediction using ML Classification Algorithms," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 6, 2021, doi: 10.14569/IJACSA.2021.0120662.
- [10] Jijo, Bahzad Mohsin Abdulazeez, Adnan. Classification Based on Decision Tree Algorithm for Machine Learning. *Journal of Applied Science and Technology Trends*. 2021;2(1). 20-28.
- [11] J. Peng, C. Chen, M. Zhou, X. Xie, Y. Zhou, and C. H. Luo, "A Machine-learning Approach to Forecast Aggravation Risk in Patients with Acute Exacerbation of Chronic Obstructive Pulmonary Disease with Clinical Indicators," *Sci Rep*, vol. 10, no. 1, 2020, doi: 10.1038/s41598-020-60042-1
- [12] O. A. Nastiti, "Sistem Pakar Klasifikasi Stroke dengan Metode Naive Bayes Classifier dan Certainty Factor Sebagai Alat Bantu Diagnosis," Surabaya, Jun. 2016. Accessed: Aug. 20, 2022. [Online]. Available: <http://repository.unair.ac.id/50732/13/50732.pdf>
- [13] Hasanah, Iswatun Purwanti, Endah Widiyanti, Prihartini. Design and Implementation of an Early Screening Application for Dengue Fever Patients Using Android-Based Decision Tree C4.5 Method. *International Journal on Advanced Science, Engineering and Information Technology*. 2020;10. 2237. 10.18517/ijaseit.10.6.5771.
- [14] H. Rajora, N. Punn, S. Sonbhadra, and S. Agarwal, Web based disease prediction and recommender system. 2021. Accessed: Aug. 19, 2022. [Online]. Available: [https://www.researchgate.net/publication/352209191Web\\_based\\_diseas\\_prediction\\_and\\_recommender\\_system](https://www.researchgate.net/publication/352209191Web_based_diseas_prediction_and_recommender_system).

- [15] Fedesoriano, “Stroke Prediction Dataset,” Kaggle Dataset, 2020. <https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset/discussion/237113> (accessed Aug. 21, 2022).
- [16] MenKes RI, “Peraturan Menteri Kesehatan Republik Indonesia Nomor 25 Tahun 2016 tentang Indonesia Rencana Aksi Nasional Kesehatan Lanjut Usia Tahun 2016-2019,” *Peraturan Menteri Kesehatan Republik Indonesia*. Indonesia, 2016.
- [17] KemenKes, “*Kementerian Kesehatan Republik Indonesia*,” Kementerian Kesehatan Republik Indonesia, 2013. <https://www.kemkes.go.id/index.php?txtKeyword=glukosaact=search-by-mappgnumber=0charindex=strucid=1280fullcontent=1C-ALL=1> (accessed Jan. 19, 2023).
- [18] Kemenkes, “Tabel Batas Ambang indeks Massa tubuh (IMT) - Direktorat P2PTM,” *Kementerian Kesehatan Republik Indonesia*, 2019. <https://p2ptm.kemkes.go.id/infographicp2ptm/obesitas/tabel-batas-ambang-indeks-massa-tubuh-imt> (accessed Jan. 19, 2023).
- [19] ISO, “ISO - ISO/IEC 25010:2011 - Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models,” Mar. 2011. <https://www.iso.org/standard/35733.html> (accessed Aug. 23, 2022).