Analyzing the Impact of Data Resampling on Stroke Prediction using Machine Learning

Majid Rahardi
Department of Informatics, Faculty of Computer Science, Universitas Amikom Yogyakarta, Sleman, Indonesia
majid@amikom.ac.id (corresponding author)

Afrig Aminuddin
Department of Information System, Faculty of Computer Science, Universitas Amikom Yogyakarta, Sleman, Indonesia
afrig@amikom.ac.id

Ferian Fauzi Abdulloh
Department of Informatics, Faculty of Computer Science, Universitas Amikom Yogyakarta, Sleman, Indonesia
ferian@amikom.ac.id

Bima Pramudya Asaddulloh
Department of Informatics, Faculty of Computer Science, Universitas Amikom Yogyakarta, Sleman, Indonesia
bima@students.amikom.ac.id

Hesmeralda Rojas Enriquez
Escuela Academico Profesional de Ingenieria Informatica y Sistemas, Facultad de Ingenieria, Universidad Nacional Micaela Bastidas de Apurimac, Abancay, Peru
hrojas@unamba.edu.pe

Kusnawi Kusnawi
Department of Informatics, Faculty of Computer Science, Universitas Amikom Yogyakarta, Sleman, Indonesia
khusnawi@amikom.ac.id

Received: 27 November 2024 | Revised: 8 January 2025 | Accepted: 12 January 2025

 DOI: https://doi.org/10.48084/etasr.9736

ABSTRACT

This study focuses on stroke prediction using machine learning algorithms and evaluates the impact of different resampling techniques, including original, under-sampling, and over-sampling, on classification performance. The classifiers used in this study include Random Forest (RF), Decision Tree (DT), Gradient Boosting (GB), and K-Nearest Neighbor (KNN). Each model was trained and evaluated using performance metrics such as accuracy, precision, recall, F1-score, and AUC. The results demonstrate that RF trained on the oversampled dataset achieved the best performance with an accuracy of 94.31%, a precision of 93.52%, a recall of 95.27%, an F1-score of 94.39%, and an AUC of 98.46% on the test set. These findings highlight the effectiveness of oversampling in handling imbalanced datasets and the superiority of RF in stroke prediction tasks compared to other classification methods and resampling techniques.

Keywords-machine learning; stroke classification; resampling

I. INTRODUCTION

Stroke is a medical condition that occurs when blood flow to a part of the brain is stopped or significantly reduced, resulting in damage to brain tissue [1]. This disruption can be due to blockage (ischemic stroke) or rupture of a blood vessel (hemorrhagic stroke), which causes the affected area of the brain to lose oxygen and essential nutrients [2]. Stroke symptoms often appear suddenly and include weakness or numbness in the limbs, confusion or difficulty speaking and understanding speech, visual disturbances in one or both eyes, and severe headaches that appear suddenly for no apparent reason [3]. As stroke is a leading cause of disability and death worldwide, awareness of early signs and the importance of receiving medical treatment as soon as possible is crucial [4]. Stroke prevention efforts involve controlling risk factors [5]. Increasing knowledge about strokes, both in terms of prevention and treatment, is an essential step in reducing their impact [6].

Disease identification using Machine Learning (ML) technology has become one of the innovative approaches in the early detection of various medical conditions [7]. In this context, ML algorithms can be trained to recognize complex patterns in medical data, such as patient medical history, food and beverage consumption habits, physical movement habits, etc. [8]. Through extensive data analysis, machine learning algorithms accurately identify early stroke indicators before clinical symptoms appear [9]. Similarly, for chronic diseases, ML models can predict a person's risk based on a combination of risk factors that influence the occurrence of a stroke. ML has the potential to be an invaluable tool in the healthcare system, enabling faster and more targeted interventions, which can ultimately save lives and improve patient quality of life [10].

This study uses four classification algorithms, Gradient Boosting (GB), K-Nearest Neighbors (KNN), Decision Tree (DT), and Random Forest (RF), to evaluate their performance in early stroke detection. GB is known for improving model accuracy by combining weak learners into a strong ensemble, effectively handling complex data patterns [11]. KNN is a simple but powerful algorithm that makes decisions based on the proximity of data points in the feature space. However, its performance can be affected by large datasets and high dimensionality [12]. The DT algorithm provides an intuitive structure by splitting data based on decision rules, although it can sometimes overfit [13]. RF addresses this limitation by building an ensemble of decision trees, reducing the risk of overfitting and improving prediction accuracy through aggregation [14]. Comparing these four algorithms allows research to compare the strengths and weaknesses of different approaches, ensuring that the most accurate and practical model is selected for early disease detection.

This study aimed to develop an effective ML pipeline for predicting stroke likelihood using data preprocessing, feature selection, and model evaluation techniques. The method begins with data cleaning, including handling missing values, encoding categorical features, and balancing the dataset using Random Undersampling (RUS) and Synthetic Minority Over-sampling Technique (SMOTE) techniques. Exploratory Data Analysis (EDA) is performed to understand the data, followed by correlation analysis to identify essential features. The data are then normalized and divided into training, validation, and testing sets. Four machine learning models, GB, KNN, DT, and RF, are trained and evaluated using various performance metrics, including accuracy, precision, recall, F1-score, and ROC-AUC. Results are compared in original, undersampled, and over-sampled datasets to determine the most accurate and effective model for stroke prediction.

In [15], an explainable ML pipeline was proposed to predict stroke occurrences using imbalanced data. This study aimed to create reliable models, address class imbalance, and interpret the model output. Six classifiers were compared, finding that Multi-Layer Perceptron (MLP) was the most effective, achieving an accuracy of 70.85% and a false-negative rate of 18.60%. Shapley Additive Explanations (SHAP) were used to understand the impact of various risk factors on predictions. The study concluded that age, BMI, and average glucose level were the most significant predictors, and the proposed method could enhance risk stratification and timely diagnosis for stroke patients. In [16], an automated stroke prediction system using ML was presented to enhance early intervention and reduce stroke-related disabilities and mortality. This study compared six classifiers, with more complex models achieving up to 91% accuracy. SHAP and LIME techniques were applied to explain the models' decisions, ensuring transparency and trust in medical applications. The study also addressed class imbalance issues and proposed a web-based application for real-time stroke prediction, emphasizing the importance of Explainable AI (XAI) in healthcare.

In [17], an XAI model was introduced to predict strokes using EEG signals. ML models were used to classify patients with ischemic stroke and healthy individuals, achieving around 80% accuracy with Adaptive Gradient Boosting (AGB). The Eli5 and LIME tools were employed to explain the model's behavior and identify significant features contributing to stroke prediction, such as spectral delta and theta waves. The study involved 48 stroke patients and 75 healthy adults, with EEG data collected during activities such as walking and reading. The findings aimed to improve post-stroke treatment and recovery by making diagnostic decisions more explainable to healthcare professionals. In [18], distributed ML on Apache Spark was investigated for stroke prediction. This study compared four machine learning algorithms: DT, SVM, RF, and Logistic Regression (LR). Data preprocessing, hyperparameter tuning, and cross-validation were applied to enhance the models' performance. The results showed that RF achieved the highest accuracy at 90%, demonstrating its effectiveness in predicting strokes. This study highlighted the importance of early stroke prediction for improving patient outcomes and reducing long-term disabilities.

In [19], various Electronic Health Record (EHR) factors were analyzed to predict stroke using ML and neural networks. Age, heart disease, average glucose level, and hypertension were identified as the most critical factors for stroke prediction. This study used principal component analysis to reduce the feature space. Several ML models were combined, finding that a perceptron neural network with these four critical attributes

provided the highest accuracy. The highly imbalanced dataset was balanced using subsampling techniques to ensure reliable results. This study concluded that focusing on these four attributes could significantly improve stroke prediction, achieving an accuracy of 92%.

Existing research on stroke prediction using ML has focused primarily on addressing class imbalance and improving model accuracy. However, these studies did not comprehensively compare different resampling techniques, such as undersampling, oversampling, and the original dataset. Most studies, such as [15, 18], evaluated specific classifiers without exploring how various resampling methods affect a broader set of models. Additionally, comprehensive evaluation metrics are limited beyond accuracy, and the interaction of data preprocessing steps with resampling techniques remains underexplored. Furthermore, these studies often fail to validate models across diverse datasets, raising concerns about the generalizability of their findings.

This research contributes to the field by systematically comparing the impact of different resampling methods on stroke prediction models. This evaluates a diverse set of classifiers using comprehensive evaluation metrics, providing a more holistic assessment. This study also integrates key preprocessing steps, such as feature selection and data normalization, analyzing their combined effects with resampling techniques to enhance model performance. Additionally, it ensures robust validation through cross-validation, addressing the generalizability concerns often overlooked in existing research.

TABLE I. STROKE CLASSIFICATION DATA SAMPLE

No	Class	BMI	Smoking	…	Sex	AgeCategory	GenHealth	Sleeptime
1	No	16.60	Yes	…	Female	55-59	Very Good	5
2	No	20.34	No	…	Female	80 or older	Very Good	7
3	No	26.58	Yes	…	Male	65-69	Fair	8
4	No	24.21	No	…	Female	75-79	Good	6
5	No	23.71	No	…	Female	40-44	Very Good	8
…	…	…	…	…	…	…	…	…
319790	Yes	27.41	Yes	…	Male	60-64	Fair	6
319791	No	29.84	Yes	…	Male	35-39	Very Good	5
319792	No	24.24	No	…	Female	45-49	Good	6
319793	No	32.81	No	…	Female	25-29	Good	12
319794	No	46.56	No	…	Female	80 or older	Good	8

II. PROPOSED METHOD

A. Dataset

The dataset [20] has several features that support stroke prediction, including demographic information, such as age and gender, medical history, such as hypertension and heart disease, lifestyle factors, including smoking status, and other relevant attributes. The prediction target in this dataset is the stroke column with two unique values: "Yes" or "No. The data distribution in this dataset consists of 289,653 data with the "No" label and 12064 data with the "Yes" label. Table I presents a sample of the dataset.

B. Exploratory Data Analysis (EDA)

The initial step in the proposed method involved data loading and EDA, which provides insights into the data distribution and relationships between features, guiding the feature selection and data transformation processes. The dataset used was loaded using the pd.read_csv() function from the pandas library. The distribution of the target variable was assessed using a pie chart. Understanding the class distribution is crucial, especially for imbalanced datasets, as it directly influences the choice of resampling methods. Figure 2 presents the distribution of the target variable.

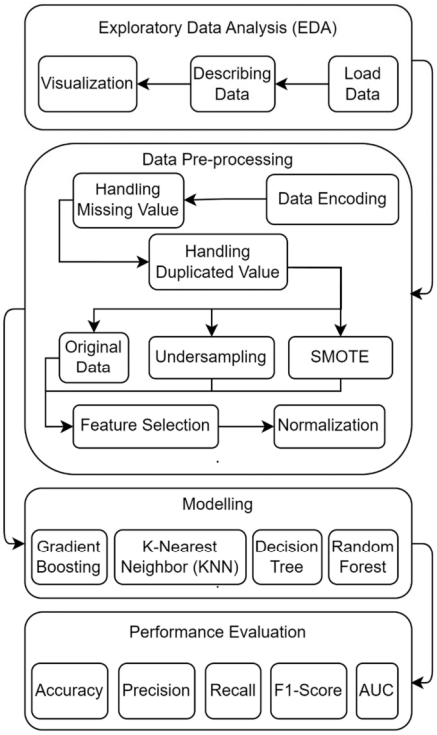

Fig. 1. The research flow diagram.

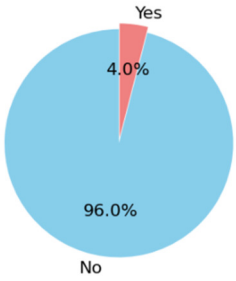

Fig. 2. Target variable distribution.

C. Data Preprocessing

The dataset comprises numerical and categorical features, necessitating various preprocessing steps to ensure the data is in a suitable format for modeling. Categorical features are transformed into numerical representations using LabelEncoder() from scikit-learn. Label encoding converts categories into numerical labels ranging from 0 to $n-1$, where n is the number of unique categories in the feature [21]. This transformation is crucial for algorithms that require numerical input. Missing values in the dataset can introduce biases or inaccuracies in the model [22]. To address this, missing values were replaced with the mean of each column using the df.fillna(df.mean()) method. This approach assumes that the missing values are missing at random and replaces them with the average, preserving the overall distribution. Duplicate rows, which can lead to overfitting, were removed using df.drop_duplicates(). Removing duplicates ensures that the dataset's variability is accurately captured and the model's performance reflects true predictive ability rather than redundancy [23].

$$X_{encoded} = LabelEncoder(X) \quad (1)$$

$$X_{imputed} = X \cup \{mean(X)\} \quad (2)$$

Data balancing techniques address the class imbalance commonly observed in stroke prediction datasets, where non-stroke cases far outnumber stroke cases. Random Undersampling (RUS) balances the dataset by reducing the number of samples in the majority class [24]. This technique involves randomly selecting a subset of the majority class such that its size matches that of the minority class. Although this method can lead to information loss, it helps prevent the model from being biased toward the majority class. The Synthetic Minority Over-sampling Technique (SMOTE) generates synthetic samples for the minority class [25]. SMOTE creates new instances by interpolating between existing minority class instances, enhancing it without replicating samples and improving the model's generalization ability.

$$X_{oversampled} = X_{minority} + \lambda \cdot (X_{neighbor} - X_{minority}) \quad (3)$$

where λ is a random number between 0 and 1, and $X_{neighbor}$ is a randomly selected minority class instance.

Feature selection was performed to identify the most relevant features for stroke prediction, enhancing the model's efficiency and interpretability [26]. The Pearson correlation coefficient evaluates the linear relationship between features and the target variable. The correlation coefficient is given by:

$$\rho(X,Y) = \frac{\sum(X-\bar{X})(Y-\bar{Y})}{\sqrt{\sum(X-\bar{X})^2 \sum(Y-\bar{Y})^2}} \quad (4)$$

Features with the highest absolute correlation with the target variable were selected, as they were expected to contribute most significantly to the model's predictive power. The top five features with the highest absolute correlation with the target variable were selected for model training. This selection was based on their potential to improve model accuracy and reduce computational complexity.

Data normalization was applied to standardize the feature values, ensuring that each feature contributes equally to the model's performance [27]. StandardScaler was employed to scale features to have zero mean and unit variance. This normalization is crucial for algorithms sensitive to the scale of input features, such as gradient-based models.

$$\rho(X,Y) = \frac{\sum(X-\bar{X})(Y-\bar{Y})}{\sqrt{\sum(X-\bar{X})^2 \sum(Y-\bar{Y})^2}} \quad (5)$$

where μ is the mean and σ is the standard deviation of the feature.

D. Classification Modelling

The dataset was split into training, testing, and validation sets using the train_test_split() function to evaluate the models' performance. Data were split into training (60%), testing (20%) and validation (20%) sets using random seed (random_state = 42) to ensure reproducibility. The training set was used for model training, while the testing set was reserved for evaluating the model's generalization performance. The validation set ensures that the models are evaluated on unseen data, providing an unbiased performance estimate [28]. Four classification models were selected to compare their performance on the original, undersampled, and oversampled datasets. These models include RF, KNN, GB, and DT, each offering unique strengths in handling classification tasks.

$$(X_{train}, X_{test}, y_{train}, y_{test}) = train\_test\_split(X, y, test\_size = 0.2, random\_state = 42) \quad (6)$$

The models were initialized using their respective default parameters. This initialization serves as a baseline for evaluating the impact of data resampling on predictive performance. Each model was trained on the training data using the fit() method. Training involves learning patterns from the data to predict unseen data. After training, the models were tested on the testing set using the predict() method. The predictions were compared against the actual values to assess the performance of the models.

$$Model_{train} = Model.fit(X_{train}, y_{train}) \quad (7)$$

$$\hat{y} = Model.predict(X_{test}) \quad (8)$$

E. Performance Evaluation

The evaluation stage involved calculating various metrics to compare the performance of the models trained on the original, undersampled, and oversampled datasets [29]. The

performance of each model was evaluated using accuracy, precision, recall, F1-score, and confusion matrices. These metrics provide a comprehensive view of the model's ability to classify stroke and non-stroke cases correctly. Accuracy measures the proportion of correctly classified instances among the total instances. Precision measures the proportion of true positive predictions among all positive predictions. Recall measures the proportion of true positive predictions among all actual positives. F1-score is the harmonic mean of precision and recall, balancing the two metrics.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (9)$$

$$Precision = \frac{TP}{TP+FP} \quad (10)$$

$$Recall = \frac{TP}{TP+FN} \quad (11)$$

$$F1-score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (12)$$

The confusion matrix summarizes the model's performance by displaying true positives, true negatives, false positives, and false negatives. The Receiver Operating Characteristic (ROC) curve plots the true positive rate against the false positive rate at various threshold settings. The Area Under the Curve (AUC) quantifies the model's ability to distinguish between classes, with higher values indicating better performance.

$$\text{AUC} = \int_0^1 \text{ROC}(t)\ dt \quad (13)$$

III. RESULTS AND ANALYSIS

A. Performance Evaluation of the Models on the Original Dataset

The initial evaluation was conducted on the original imbalanced dataset, where the models were trained and tested without resampling. The results, summarized in Table II, indicate the baseline performance of each classifier. Performance metrics provide a reference point to compare the effects of resampling.

TABLE II. MODELS' PERFORMANCE ON THE ORIGINAL DATASET

Classifier	Accuracy	Precision	Recall	F1-score	AUC
GB (Test)	96.02%	**68.42%**	00.54%	01.07%	81.48%
GB (Val)	**96.04%**	45.83%	00.46%	00.91%	**82.06%**
KNN (Test)	95.84%	21.18%	01.49%	02.79%	59.18%
KNN (Val)	95.85%	16.57%	01.17%	02.19%	59.84%
DT (Test)	92.32%	11.21%	13.31%	12.17%	54.43%
DT (Val)	92.38%	11.66%	**14.06%**	**12.75%**	54.86%
RF (Test)	95.80%	11.58%	00.79%	01.48%	75.74%
RF (Val)	95.84%	10.97%	00.71%	01.34%	76.61%

GB outperformed the other models in terms of accuracy (96.02% on test and 96.04% on validation) and AUC (81.48% on test and 82.06% on validation), suggesting that it had the best balance between correctly predicting both cases. However, it exhibited extremely low precision, recall, and F1 scores across both sets, indicating that although overall prediction accuracy is high, the model struggled significantly in correctly identifying stroke cases, as reflected in low-performance metrics for positive class predictions.

KNN shows comparable accuracy to GB (95.84% on test) but with drastically lower precision, recall, and F1-scores, indicating poor performance in detecting stroke cases. DT had the lowest accuracy (92.32% on test), and its performance metrics were similarly low, demonstrating inadequate model performance. RF performed slightly better than DT in accuracy (95.80% on test) and AUC but still suffered from very low precision, recall, and F1 scores, highlighting a consistent pattern across all models where high accuracy does not translate to reliable detection of stroke cases. Despite their high accuracy, these models may not be effective for clinical use due to their poor ability to correctly identify positive cases, as reflected in the other metrics.

The results in Table II underscore a notable disparity between the overall accuracy of the classifiers and their ability to effectively identify stroke cases, as evidenced by their low precision, recall, and F1 scores. This suggests that the models, particularly GB and KNN, achieve high accuracy largely due to correct predictions of the dominant class, rather than a robust ability to differentiate between positive and negative cases. This imbalance may stem from the class imbalance in the dataset or suboptimal feature representation for positive cases.

B. Performance Evaluation on the Undersampled Dataset

This section evaluates the models trained on the undersampled dataset, where the amount of data rows changed to 24,128 (the original amount was 30,1717). RUS was used to balance the class distribution by reducing the number of samples in the majority class. As shown in Table III, undersampling generally improved the recall rates across all classifiers, indicating a better balance in correctly identifying stroke cases, although often at the cost of reduced accuracy.

TABLE III. MODELS' PERFORMANCE ON THE UNDERSAMPLED DATASET

Classifier	Accuracy	Precision	Recall	F1-score	AUC
GB (Test)	**75.49%**	**73.83%**	**80.13%**	76.85%	**82.84%**
GB (Val)	74.57%	71.67%	78.96%	**75.14%**	82.72%
KNN (Test)	69.62%	70.72%	68.58%	69.63%	74.03%
KNN (Val)	68.52%	67.62%	67.76%	67.69%	74.07%
DT (Test)	65.46%	66.00%	65.97%	65.99%	65.46%
DT (Val)	65.64%	64.42%	65.63%	65.02%	65.64%
RF (Test)	73.77%	72.47%	77.97%	75.12%	80.55%
RF (Val)	73.60%	70.94%	77.47%	74.06%	79.91%

GB achieved the highest accuracy (75.49% on test and 74.57% on validation) and F1 score (76.85% on test and 75.14% on validation), indicating a solid overall performance with a good balance between precision and recall. Its AUC values (82.84% on test and 82.72% on validation) further confirm its effective discrimination between classes. RF also performed well, with accuracy close to GB (73.77% on test and 73.60% on validation) and substantial precision, recall, and F1 scores, although slightly lower than GB, making it a robust alternative. KNN had a moderate performance with accuracy around 69-70%, showing a decent balance between precision and recall but lower AUC (~74%), indicating less reliable classification than the top models. DT had the lowest accuracy (~65-66%) and F1 scores, reflecting the weakest performance among the classifiers. Overall, GB and RF emerged as the top

performers on the undersampled dataset, offering a good trade-off between precision and recall, making them more suitable for balanced predictions than the other models.

The results in Table III reveal how undersampling the dataset significantly alters the performance dynamics of the classifiers. Unlike the models trained on the original dataset, where accuracy dominated as the primary indicator of success, the undersampled dataset allowed the evaluation metrics to shift focus toward a balance between precision and recall. This adjustment is particularly critical in contexts where the accurate identification of positive cases (stroke) is paramount. Although the models sacrifice some accuracy compared to their performance on the original data, they gain significant improvements in metrics directly tied to identifying stroke cases, such as precision, recall, and F1-score.

C. Performance Evaluation on the Oversampled Dataset

This experiment applied the oversampling method to the original dataset to obtain better results. The number of data rows changed to 579,306 (the original amount was 301,717). This section evaluates models trained on the oversampled dataset, where SMOTE was applied. SMOTE generates synthetic samples to balance the class distribution. As detailed in Table IV, oversampling improved both recall and F1-score significantly, especially for classifiers such as GB and RF, demonstrating that SMOTE effectively mitigated the effects of class imbalance.

TABLE IV. PERFORMANCE METRICS OF MODELS TRAINED ON OVERSAMPLED DATASET

Classifier	Accuracy	Precision	Recall	F1-score	AUC
GB (Test)	82.65%	83.59%	81.41%	82.48%	91.63%
GB (Val)	82.92%	83.81%	81.61%	82.69%	91.83%
KNN (Test)	87.24%	80.79%	97.84%	88.50%	94.58%
KNN (Val)	87.17%	80.61%	**97.89%**	88.41%	94.71%
DT (Test)	91.35%	89.92%	93.21%	91.54%	91.36%
DT (Val)	91.25%	89.78%	93.10%	91.41%	91.27%
RF (Test)	94.31%	93.52%	95.27%	94.39%	98.46%
RF (Val)	**94.40%**	**93.60%**	95.31%	**94.45%**	**98.48%**

RF outperformed had the highest accuracy (94.31% on test and 94.40% on validation) and an outstanding AUC (98.46% on test and 98.48% on validation), indicating excellent performance and a solid ability to distinguish between classes. On the test dataset, its high precision (93.52%), recall (95.27%), and F1 score (94.39%) reflect a balanced and reliable prediction of both positive and negative cases. DT also performed well, achieving high accuracy (91.35% on test and 91.25% on validation) with balanced precision, recall, and F1 scores (~91-93%), making it practical for accurate and consistent predictions. KNN showed slightly lower accuracy (87.24% on test and 87.17% on validation) but excelled in recall (97.84% on test), suggesting a solid sensitivity in identifying positive cases, though at the expense of slightly lower precision. GB performed moderately with an accuracy of 82-83% and an AUC of 91.6-91.8%, indicating decent but less robust performance compared to RF and DT. Overall, RF was the top performer with the most balanced and high metrics, making it the most suitable for datasets that have been oversampled to address class imbalance. The results in Table IV highlight how oversampling transforms the performance of classifiers, improving their ability to identify and balance between classes. This preprocessing technique amplifies the recall across most models, ensuring they are more sensitive to detecting positive cases without significantly compromising precision. The overall effect is a noticeable improvement in F1 score and AUC values compared to the original and undersampled datasets, making the models more viable for practical applications. The enhancements across all models underscore the importance of addressing class imbalance to achieve reliable and fair predictions.

D. Comparative Analysis of Resampling Techniques

This section compares the different resampling techniques (original, undersampling, and oversampling) across all classifiers. Table V consolidates the results to highlight each approach's performance trends and trade-offs. The comparative analysis shows that oversampling generally led to better overall performance, especially regarding recall and F1-score, which are crucial in medical diagnostics.

TABLE V. PERFORMANCE METRICS UNDER EACH RESAMPLING METHOD

Dataset	Classifier	Accuracy	Precision	Recall	F1-score	AUC
Original	GB (Test)	96.02%	68.42%	00.54%	01.07%	81.48%
Original	GB (Val)	**96.04%**	45.83%	00.46%	00.91%	82.06%
Undersampled	KNN (Test)	75.49%	73.83%	80.13%	76.85%	82.84%
Undersampled	KNN (Val)	74.57%	71.67%	78.96%	75.14%	82.72%
Oversampled	DT (Test)	94.31%	93.52%	95.27%	94.39%	98.46%
Oversampled	DT (Val)	94.40%	**93.60%**	**95.31%**	**94.45%**	**98.48%**

For the original dataset, GB showed high accuracy (96.02% on test and 96.04% on validation) but inferior precision, recall, and F1-scores (all near zero), indicating that despite the high accuracy, the model fails to identify positive cases effectively, making it unreliable for practical use. When trained on the undersampled dataset, GB demonstrated more balanced performance, with reduced accuracy (75.49% on test and 74.57% on validation) but significantly improved precision (around 73-74%), recall (around 80%), and F1 (around 76-75%) scores. This suggests that undersampling helps the model better capture the minority class, though at the cost of overall accuracy. For the oversampled dataset, RF achieved the best overall performance, with high accuracy (94.31% on test and 94.40% on validation) and excellent precision, recall, and F1-scores (all above 93%), alongside an impressive AUC (98.46% on test and 98.48% on validation). This indicates that oversampling effectively addresses class imbalance, allowing the RF model to maintain high predictive power across both classes. Thus, combined with RF, oversampling provides the most reliable and balanced approach for handling imbalanced datasets.

These results show that the choice of resampling method significantly influences the classifiers' performance. The results demonstrate how different approaches to addressing class imbalance can either enhance or hinder a model's ability to identify positive cases effectively, depending on the context. When comparing the three approaches, oversampling emerges as the most effective method to create a balanced and reliable classifier. Although undersampling provides a pathway to improve sensitivity at the cost of accuracy, oversampling preserves high accuracy while addressing the limitations seen with the original dataset. This balance is crucial in practical applications where both overall predictive reliability and sensitivity to minority classes are necessary. Ultimately, the results in Table V reinforce the importance of resampling methods in ML workflows. The stark differences in performance metrics across datasets demonstrate that class imbalance cannot be overlooked, as it significantly impacts a model's utility. Among the resampling strategies, oversampling with models such as DT or RF appears to offer the most comprehensive solution, ensuring both high predictive accuracy and sensitivity to the minority class. This insight is vital for optimizing ML models for real-world, high-stakes applications.

E. Comparison with the Existing Research

This section compares the results of this study with previous research on similar datasets. The proposed models, which utilized DT and RF on oversampled datasets, were evaluated against various approaches, including LR, RF, AGB, and Neural Networks (NN), applied to stroke classification tasks. Table VI presents the comparison between existing schemes and the proposed method.

TABLE VI. COMPARISON WITH EXISTING SCHEMES IN STROKE CLASSIFICATION

Scheme	Best model	Accuracy	Precision	Recall	F1-score	AUC
[15]	LR	73.52%	-	78.12%	-	83.30%
[16]	RF	90.36%	91.5%	91.25%	90.5%	-
[17]	AGB	80%	82%	78%	80%	80%
[18]	RF	90%	91.5%	90.5%	90.5%	-
[19]	NN	78%	78%	71%	74%	-
Proposed method	DT (Test)	91.35%	89.92%	93.21%	91.54%	91.36%
	DT (Val)	91.25%	89.78%	93.10%	91.41%	91.27%
	RF (Test)	94.31%	93.52%	95.27%	94.39%	98.46%
	RF (Val)	**94.40%**	**93.60%**	**95.31%**	**94.45%**	**98.48%**

LR in [15] achieved lower accuracy (73.52%) and recall (78.12%), indicating less reliable performance in classifying positive cases. RF in [16, 18] achieved accuracies of ~90% with high precision, yet it fell short compared to the proposed method's performance, especially in recall and F1-score. The AGB model in [18] achieved balanced but lower metrics (80% accuracy, 82% precision), highlighting that, although it provided consistency, it did not reach the predictive power of the proposed RF method. The NN in [19] showed the lowest performance among these studies, with an accuracy of 78% and a recall of 71%, underscoring the challenges that NNs face in this classification task without substantial data preprocessing.

This comparison highlights the effectiveness of the proposed RF model trained on oversampled data, which consistently outperformed existing methods in accuracy, precision, recall, and AUC. By addressing class imbalance through oversampling, the proposed approach enhances the model's reliability and applicability in medical diagnostics, offering a clear improvement over previously reported techniques. This demonstrates the value of combining robust classifiers with appropriate data resampling techniques to achieve optimal performance in critical classification tasks. The proposed RF approach with oversampling achieved not only the highest accuracy (94.31% on the test set and 94.40% on the validation set) but also excelled in recall (95.27%), F1-score (94.39%), and AUC (98.46%). These metrics indicate a finely tuned model capable of effectively balancing precision and recall while minimizing false positives and false negatives. Such performance is critical in medical diagnostics, where misclassifications can have significant consequences.

IV. CONCLUSION

This study systematically compared the performance of various machine learning classifiers, including RF, DT, GB, and KNN, on stroke prediction, employing different resampling methods. The results indicate that the oversampling technique combined with RF achieved the highest accuracy of 94.31%, with an AUC of 98.46%, demonstrating its superior ability to balance precision and recall. Models trained on the original and undersampled datasets generally performed less effectively, particularly regarding recall and F1-score. These findings underscore the importance of addressing the class imbalance in medical datasets and suggest that RF with oversampling is the most reliable approach for stroke prediction. Future work could explore the use of additional ensemble methods and deep learning techniques to further enhance model performance and generalizability.

ACKNOWLEDGMENT

This research was supported and fully funded by Universitas Amikom Yogyakarta.

REFERENCES

[1] T. Roushdy *et al.*, "Applying the World Stroke Organization roadmap in planning a model for stroke service implementation in Matrouh Governorate-Egypt: a World Stroke Organization young future stroke leaders' analytical study," *The Egyptian Journal of Neurology, Psychiatry and Neurosurgery*, vol. 59, no. 1, Nov. 2023, Art. no. 150, https://doi.org/10.1186/s41983-023-00753-0.

[2] J. Droś, N. Segiet, G. Początek, and A. Klimkowicz-Mrowiec, "Five-year stroke prognosis. Influence of post-stroke delirium and post-stroke dementia on mortality and disability (Research Study – Part of the PROPOLIS Study)," *Neurological Sciences*, vol. 45, no. 3, pp. 1109–1119, Mar. 2024, https://doi.org/10.1007/s10072-023-07129-5.

[3] K. P. Berg, V. F. I. Sørensen, S. N. F. Blomberg, H. C. Christensen, and C. Kruuse, "Recognition of visual symptoms in stroke: a challenge to patients, bystanders, and Emergency Medical Services," *BMC Emergency Medicine*, vol. 23, no. 1, Aug. 2023, Art. no. 96, https://doi.org/10.1186/s12873-023-00870-2.

[4] A. M. Alghamdi, M. A. Al-Khasawneh, A. Alarood, and E. Alsolami, "The Role of Machine Learning in Managing and Organizing Healthcare Records," *Engineering, Technology & Applied Science Research*, vol. 14, no. 2, pp. 13695–13701, Apr. 2024, https://doi.org/10.48084/etasr.7027.

[5] Z. Xie, C. Wang, X. Huang, Z. Wang, H. Shangguan, and S. Wang, "Prevalence and Risk Factors of Stroke in Inpatients with Type 2 Diabetes Mellitus in China," *Current Medical Science*, vol. 44, no. 4, pp. 698–706, Aug. 2024, https://doi.org/10.1007/s11596-024-2911-1.

[6] N. K. Al-Shammari *et al.*, "Cardiac Stroke Prediction Framework using Hybrid Optimization Algorithm under DNN," *Engineering, Technology & Applied Science Research*, vol. 11, no. 4, pp. 7436–7441, Aug. 2021, https://doi.org/10.48084/etasr.4277.

[7] S. M. Alanazi and G. S. M. Khamis, "Optimizing Machine Learning Classifiers for Enhanced Cardiovascular Disease Prediction," *Engineering, Technology & Applied Science Research*, vol. 14, no. 1, pp. 12911–12917, Feb. 2024, https://doi.org/10.48084/etasr.6684.

[8] S. Saturi, "Review on Machine Learning Techniques for Medical Data Classification and Disease Diagnosis," *Regenerative Engineering and Translational Medicine*, vol. 9, no. 2, pp. 141–164, Jun. 2023, https://doi.org/10.1007/s40883-022-00273-y.

[9] A. Hassan, S. Gulzar Ahmad, E. Ullah Munir, I. Ali Khan, and N. Ramzan, "Predictive modelling and identification of key risk factors for stroke using machine learning," *Scientific Reports*, vol. 14, no. 1, May 2024, Art. no. 11498, https://doi.org/10.1038/s41598-024-61665-4.

[10] A. A. Abujaber, Y. Imam, I. Albalkhi, S. Yaseen, A. J. Nashwan, and N. Akhtar, "Utilizing machine learning to facilitate the early diagnosis of posterior circulation stroke," *BMC Neurology*, vol. 24, no. 1, May 2024, Art. no. 156, https://doi.org/10.1186/s12883-024-03638-8.

[11] H. Ha, Q. D. Bui, D. T. Tran, D. Q. Nguyen, H. X. Bui, and C. Luu, "Improving the forecast performance of landslide susceptibility mapping by using ensemble gradient boosting algorithms," *Environment, Development and Sustainability*, Mar. 2024, https://doi.org/10.1007/s10668-024-04694-3.

[12] M. Sabri *et al.*, "A Novel Classification Algorithm Based on the Synergy Between Dynamic Clustering with Adaptive Distances and K-Nearest Neighbors," *Journal of Classification*, vol. 41, no. 2, pp. 264–288, Jul. 2024, https://doi.org/10.1007/s00357-024-09471-5.

[13] Y. Katsura, S. Ohga, K. Shimo, T. Hattori, T. Yamada, and T. Matsubara, "A decision tree algorithm to identify predictors of post-stroke complex regional pain syndrome," *Scientific Reports*, vol. 14, no. 1, Apr. 2024, Art. no. 9893, https://doi.org/10.1038/s41598-024-60597-3.

[14] T. Wang, "Improved random forest classification model combined with C5.0 algorithm for vegetation feature analysis in non-agricultural environments," *Scientific Reports*, vol. 14, no. 1, May 2024, Art. no. 10367, https://doi.org/10.1038/s41598-024-60066-x.

[15] C. Kokkotis *et al.*, "An Explainable Machine Learning Pipeline for Stroke Prediction on Imbalanced Data," *Diagnostics*, vol. 12, no. 10, Oct. 2022, Art. no. 2392, https://doi.org/10.3390/diagnostics12102392.

[16] K. Mridha, S. Ghimire, J. Shin, A. Aran, Md. M. Uddin, and M. F. Mridha, "Automated Stroke Prediction Using Machine Learning: An Explainable and Exploratory Study With a Web Application for Early Intervention," *IEEE Access*, vol. 11, pp. 52288–52308, 2023, https://doi.org/10.1109/ACCESS.2023.3278273.

[17] M. S. Islam, I. Hussain, M. M. Rahman, S. J. Park, and M. A. Hossain, "Explainable Artificial Intelligence Model for Stroke Prediction Using EEG Signal," *Sensors*, vol. 22, no. 24, Dec. 2022, Art. no. 9859, https://doi.org/10.3390/s22249859.

[18] H. Saleh, S. F. A. El-Ghany, E. M. G. Younis, and N. F. Omran, "Stroke Prediction using Distributed Machine Learning Based on Apache Spark." *International Journal of Advanced Science and Technology*, vol. 28, no. 15, pp. 89-97, 2019, https://doi.org/10.13140/RG.2.2.13478.68162.

[19] S. Dev, H. Wang, C. S. Nwosu, N. Jain, B. Veeravalli, and D. John, "A predictive analytics approach for stroke prediction using machine learning and neural networks," *Healthcare Analytics*, vol. 2, Nov. 2022, Art. no. 100032, https://doi.org/10.1016/j.health.2022.100032.

[20] "Indicators of Heart Disease (2022 UPDATE)." Kaggle, [Online]. Available: https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease.

[21] C. Shu, C. Zheng, D. Luo, J. Song, Z. Jiang, and L. Ge, "Acute ischemic stroke prediction and predictive factors analysis using hematological indicators in elderly hypertensives post-transient ischemic attack," *Scientific Reports*, vol. 14, no. 1, Jan. 2024, Art. no. 695, https://doi.org/10.1038/s41598-024-51402-2.

[22] M. Afkanpour, E. Hosseinzadeh, and H. Tabesh, "Identify the most appropriate imputation method for handling missing values in clinical structured datasets: a systematic review," *BMC Medical Research Methodology*, vol. 24, no. 1, Aug. 2024, Art. no. 188, https://doi.org/10.1186/s12874-024-02310-6.

[23] A. Ali, N. A. Emran, and S. A. Asmai, "Missing values compensation in duplicates detection using hot deck method," *Journal of Big Data*, vol. 8, no. 1, Dec. 2021, Art. no. 112, https://doi.org/10.1186/s40537-021-00502-1.

[24] C. Yang, E. A. Fridgeirsson, J. A. Kors, J. M. Reps, and P. R. Rijnbeek, "Impact of random oversampling and random undersampling on the performance of prediction models developed using observational health data," *Journal of Big Data*, vol. 11, no. 1, Jan. 2024, Art. no. 7, https://doi.org/10.1186/s40537-023-00857-7.

[25] S. Mondal, S. Ghosh, and A. Nag, "Brain stroke prediction model based on boosting and stacking ensemble approach," *International Journal of Information Technology*, vol. 16, no. 1, pp. 437–446, Jan. 2024, https://doi.org/10.1007/s41870-023-01418-0.

[26] D. Xu, S. Matinmehr, A. Sawchuk, and X. Luo, "Identifying clinical feature clusters toward predicting stroke in patients with asymptomatic carotid stenosis," *International Journal of Data Science and Analytics*, Aug. 2024, https://doi.org/10.1007/s41060-024-00597-8.

[27] T. Haritha and A. V. S. Babu, "Early-stage stroke prediction based on Parkinson and wrinkles using deep learning," *Neural Computing and Applications*, vol. 36, no. 30, pp. 18781–18805, Oct. 2024, https://doi.org/10.1007/s00521-024-10189-z.

[28] R. Y. Coley, Q. Liao, N. Simon, and S. M. Shortreed, "Empirical evaluation of internal validation methods for prediction in large-scale clinical data with rare-event outcomes: a case study in suicide risk prediction," *BMC Medical Research Methodology*, vol. 23, no. 1, Feb. 2023, Art. no. 33, https://doi.org/10.1186/s12874-023-01844-5.

[29] R. Bhowmick, S. R. Mishra, S. Tiwary, and H. Mohapatra, "Machine learning for brain-stroke prediction: comparative analysis and evaluation," *Multimedia Tools and Applications*, Aug. 2024, https://doi.org/10.1007/s11042-024-20057-6.