Vol. 3, No. 2, Agustus 2022, hal. 107-113

Jurnal Computer Science and Information Technology (CoSciTech)

p-ISSN: 2723-567X
e-ISSN: 2723-5661

http://ejurnal.umri.ac.id/index.php/coscitech/index

Teknik SMOTE untuk mengatasi imbalance data pada deteksi penyakit stroke menggunakan algoritma random forest

Desti Mualfah[1], Wahyu Fadila[2], Rahmad Firdaus[3]
Email: [1]destimualfah@umri.ac.id, [2]180401070@student.umri.ac.id, [3]rahmadfirdaus@umri.ac.id

[123]Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Muhammadiyah Riau

Diterima: 01 Agustus 2022 | Direvisi: 08 Agustus 2022 | Disetujui: 11 Agustus 2022

Abstrak

Stroke merupakan penyakit yang berpotensi menyebabkan kelumpuhan bahkan kematian. Pada tahun 2022, stroke terdapat 12,2 juta kasus stroke baru yang menambah jumlah total penderita stroke sebesar 101,4 juta. Dari perolehan data maka diperlukan sebuah teknik yang mampu melakukan deteksi pada penyakit tersebut untuk membantu dalam mendeteksi penyakit stroke, dalam hal ini pendekatan *machine learning* sebagai salah satu solusi yang dapat digunakan untuk melakukan deteksi pada penyakit stroke. Namun sayangnya data yang diperoleh dalam mendeteki penyakit stroke ditemukan adanya *imbalance class* dalam menangani tidak imbangnya class sehingga dapat mempengaruhi hasil nilai akurasi dalam mendekteksi penyakit stroke, untuk itu dibutuhkan sebuah algoritma *random forest* dan metode SMOTE dalam menangani *imbalance class*. Output yang dihasilkan ialah berupa nilai akurasi, presisi, *recall*, dan *f1-score* pada algoritma *random forest* tanpa SMOTE sebesar 0.98, 0.69, 0.51, dan 0.51. Sedangkan algoritma *random forest* dengan SMOTE mendapatkan masing-masing sebesar 0.91, 0.92, 0.91, 0.91. Terjadi kenaikan signifikan pada presisi, recall, dan *f1-score*.

Kata kunci: *stroke, random forest, SMOTE.*

SMOTE technique to overcome imbalance data in stroke detection using random forest algorithm

Abstract

Stroke is a disease that has the potential to cause paralysis and even death. In 2022, there will be 12.2 million new stroke cases, which adds to the total number of stroke survivors by 101.4 million. From the data acquisition, a technique that is able to detect the disease is needed to assist in detecting stroke, in this case a machine learning approach as a solution that can be used to detect stroke. But unfortunately the data obtained in detecting stroke found an imbalance class in dealing with class imbalances so that it can affect the results of the accuracy value in detecting stroke, for that we need a random forest algorithm and SMOTE method in dealing with class imbalance. The resulting output is in the form of accuracy, precision, recall, and f1-score values in the random forest algorithm without SMOTE of 0.98, 0.69, 0.51, and 0.51. While the random forest algorithm with SMOTE got 0.91, 0.92, 0.91, 0.91 respectively. There was a significant increase in precision, recall, and f1-score.

Keywords: *stroke, random forest, SMOTE.*

1. PENDAHULUAN

Stroke atau *Cerebrovascular Accident (CVA)* merupakan penyakit yang disebabkan karena adanya gangguan aliran darah ke otak yang dapat mengakibatkan kelumpuhan bahkan kematian[1]. Pada tahun 2022, terdapat 12,2 juta kasus baru di dunia, 62% diantaranya terjadi pada orang di bawah usia 70 tahun. Secara global, orang yang terkena stroke berjumlah 101,4 juta dan jumlah kematian yang diakibatkan stroke sebesar 6,5 juta. Dinyatakan bahwa satu dari empat orang di atas usia 25 tahun terkena stroke

selama hidup mereka[2]. Secara keseluruhan, tingkat kejadian stroke mengalami penurunan, tetapi jumlah absolut orang yang memiliki stroke, meninggal, atau menjadi disabilitas karena stroke telah meningkat[3][4].

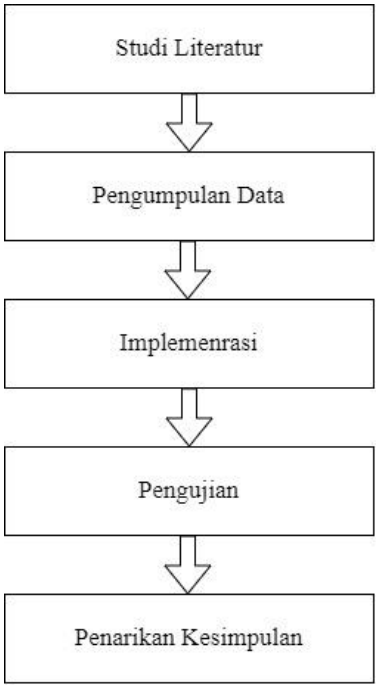
Berdasarkan pemaparan fakta dan permasalahan di atas, diperlukan adanya penelitian dalam mendeteksi penyakit stroke. Pendekatan *machine learning* dapat digunakan untuk mendeteksi penyakit. Dengan pendekatan *machine learning*, ditemukan adanya *imbalance class* pada dataset yang digunakan. *Imbalance class* terjadi karena jumlah *class majority* jauh lebih tinggi pada *class minority*[5]. Hal ini akan mempengaruhi hasil [6] akurasi.

Penelitian sebelumnya terkait dengan memprediksi penyakit stroke menggunakan metode *Deep Neural Network* mendapatkan hasil akurasi 71,6% dan sensifitas sebesar 67,4%[7]. Sedangkan penelitian terkait dengan mengatasi imbalance class menggunakan metode SMOTE dan ADASYN dalam meningkatkan performa klasifikasi herregistrasi mahasiswa baru mendapatkan hasil SMOTE lebih tinggi dari ADASYN dalam menyeimbangkan data[8].

Berdasarkan penelitian sebelumnya yang telah dilakukan, penelitian ini akan membahas deteksi penyakit stroke menggunakan algoritma *random forest* dan mengatasi *imbalance class* menggunakan SMOTE.

2. METODE PENELITIAN

Metode penelitian yang digunakan pada penelitian ini dibagi menjadi beberapa tahap agar proses yang dilakukan bisa berjalan dengan baik. Tahapan-tahapan penelitian digambarkan sebagai berikut:



Gambar 1. Alur Penelitian

2.1. Studi Literatur

Studi Literatur pada penelitian ini dilakukan dengan cara mengumpulkan serta mempelajari literatur-literatur yang berhubungan dengan penyakit stroke, algoritma random forest, dan teknik SMOTE. Sumber literatur yang digunakan berupa paper dan jurnal penelitian sebelumnya yang sesuai dengan topik penelitian ini.

2.2. Pengumpulan Data

Data pada penelitian ini diperoleh dari penelitian sebelumnya[7]. Data yang diperoleh memiliki 43.400 record pasien, 783 diantaranya mengalami stroke. Data ini terdiri dari 12 atribut yang bisa dilihat pada tabel berikut:

Tabel 1. Atribut pada dataset

No	Atribut	Keterangan
1.	*Id*	Merupakan *unique identifier*
2.	*Gender*	Merupakan jenis kelamin dari pasien
3.	*Age*	Umur pasien
4.	*Hypertension*	0 = bukan hipertensi, 1 = hipertensi

5.	*heart_disease*	0 = bukan penyakit jantung, 1 = penyakit jantung
6.	*ever_married*	Pasien pernah menikah atau tidak
7.	*work_type*	Tipe pekerjaan pasien
8.	*residence_type*	Tipe tempat tinggal pasien
9.	*avg_glucose_level*	Rata-rata glukosa dalam darah
10.	*Bmi*	Indeks massa tubuh
11.	*smoking_status*	Status merokok pasien
12.	*Stroke*	0 = bukan stroke, 1 = penyakit stroke

2.3. Implementasi

Pada tahap implementasi ini akan dilakukan penyusunan kode-kode program. Dalam penyusunan kode-kode program ini menggunakan Bahasa pemograman python dengan memanfaatkan platform yang disediakan oleh Google yakni Google Colaboratory atau lebih dikenal dengan Google Colab. Tahapan implementasi terbagi lagi menjadi beberapa tahapan seperti yang terlihat pada gambar berikut:

Gambar 2. Tahapan-tahapan pada implementasi

2.4. Pengujian

Tahapan pengujian dilakukan untuk mengetahui apakah model yang dibuat dapat mendapatkan hasil akurasi yang baik dan sesuai dengan yang diharapkan. Pada tahapan pengujian ini nantinya meliputi akurasi *(accuracy)*, presisi *(precision)*, dan *recall* dalam mendeteksi penyakit stroke.

2.5. Penarikan Kesimpulan

Setelah semua tahapan di atas telah selesai dilakukan, maka selanjutnya dilakukan penarikan kesimpulan untuk memberikan hasil akhir dari penelitian. Hasil akhir dari penelitian ini akan dibandingkan dengan penelitian sebelumnya, sehingga diketahui apakah hasil penelitian ini lebih baik dari penelitian sebelumnya atau tidak.

3. HASIL DAN PEMBAHASAN

Bab ini menguraikan tentang tahapan implementasi lebih lanjut, dimulai dari tahapan data preparation, data preprocessing, resampling, splitting data, modeling, dan evaluasi.

3.1. Data *Preparation*

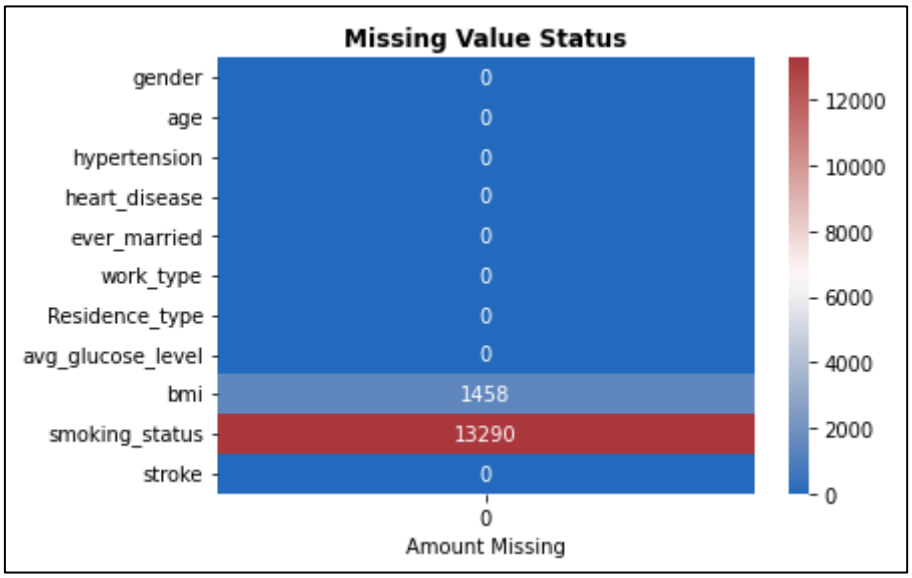
Tahapan *Data Preparation*, dimana pada tahap ini data-data dan *library* yang dibutuhkan dalam pemodelan ini dipersiapkan. Pada tahapan inilah dataset akan diimport ke Google Colab. Library-library yang digunakan dapat dilihat pada gambar berikut:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
import seaborn as sns
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from imblearn.over_sampling import SMOTE
from sklearn import preprocessing
```

Gambar 3. *Libray-library* yang digunakan

3.2. Data Preprocessing

Tahapan *Data Preprocessing* dilakukan untuk memperbaiki dan mengatasi data yang *error*, *missing value*, dan tidak konsisten[9][10]. Preprocessing yang dilakukan pada penelitian ini adalah *Handling Missing Values* dan *Label Encoder*. *Handling Missing Values* bertujuan untuk mengatasi *missing value* pada dataset. Sedangkan *Label Encoder* berfungsi dalam transformasi kolom yang memiliki tipe data object ke bentuk numerik. Missing value dapat dilihat pada gambar di bawah ini:

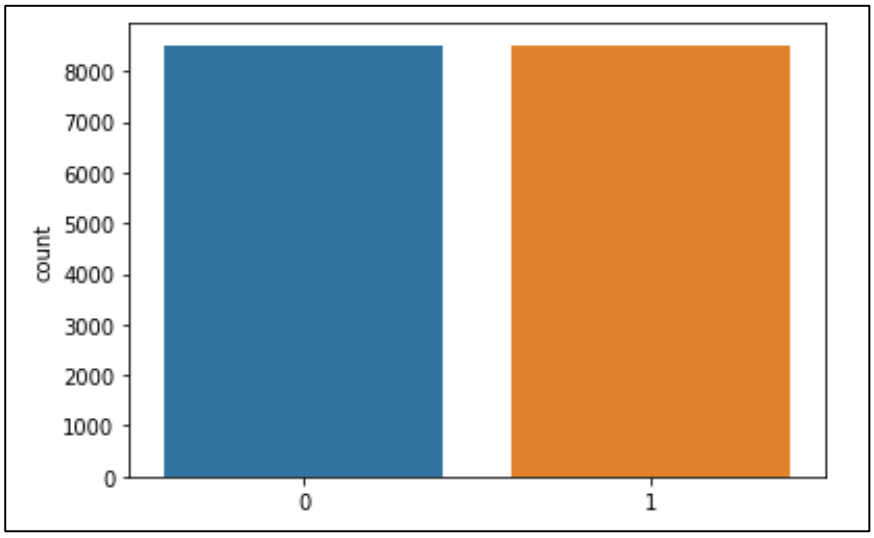


Gambar 4. *Checking missing value*

Dalam mengatasi *Missing Value* pada kolom *bmi* dan *smoking_status* memanfaatkan metode *mean* dan *modus.* Metode *mean* digunakan untuk mengatasi *missing value* pada kolom *bmi*. Sedangkan pada kolom *smoking_status* menggunakan metode *modus*.

3.3. Resampling

Teknik *resampling* pada penelitian ini menggunakan SMOTE untuk mengatasi *imbalance class* 0 dan 1 pada kolom stroke. SMOTE bekerja dengan cara menaikkan jumlah data *minority* sehingga *balance* dengan data *majority*. Hasil dari *resampling* menggunakan SMOTE dapat dilihat pada gambar berikut:



Gambar 5. *Resampling* menggunakan SMOTE

3.4 Splitting Data

Splitting Data merupakan proses memisahkan dataset menjadi data *training* dengan data *testing*. Data *training* digunakan untuk melatih algoritma untuk mendeteksi penyakit stroke. Data testing digunakan sebagai data dalam menguji algoritma sesudah melakukan training. Pada penelitian ini 80% dari dataset dialokasikan menjadi data *training*, sedangkan 20% menjadi data *test*.

3.5 Modeling

Random Forest merupakan metode *ensemble* yang terdiri dari banyak pohon keputusan[11][12]. Modeling menggunakan algoritma *random forest* ini akan digunakan untuk mendeteksi penyakit stroke berdasarkan label 0 dan 1, dimana 0 merupakan bukan penyakit stroke sedangkan 1 merupakan penyakit stroke. Dalam meningkatkan akurasi dalam mendeteksi, parameter yang

digunakan diantaranya seperti n_estimators dan *criterion*. n_*estimators* yang digunakan sebanyak 100, dan *criterion* yang digunakan adalah entropy.

3.6 Evaluasi

Pada tahap evaluasi ini akan membandingkan hasil akurasi algoritma *random forest* tanpa SMOTE dengan algoritma *random forest* yang menggunakan SMOTE.

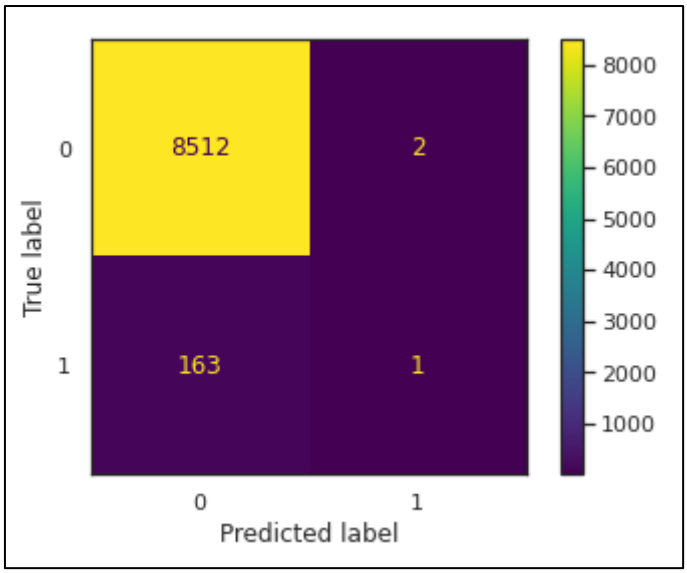
	precision	recall	f1-score	support
0	0.98	1.00	0.99	8514
1	0.33	0.01	0.01	164
accuracy			0.98	8678
macro avg	0.66	0.50	0.50	8678
weighted avg	0.97	0.98	0.97	8678

Gambar 6. Hasil akurasi *random forest* tanpa SMOTE

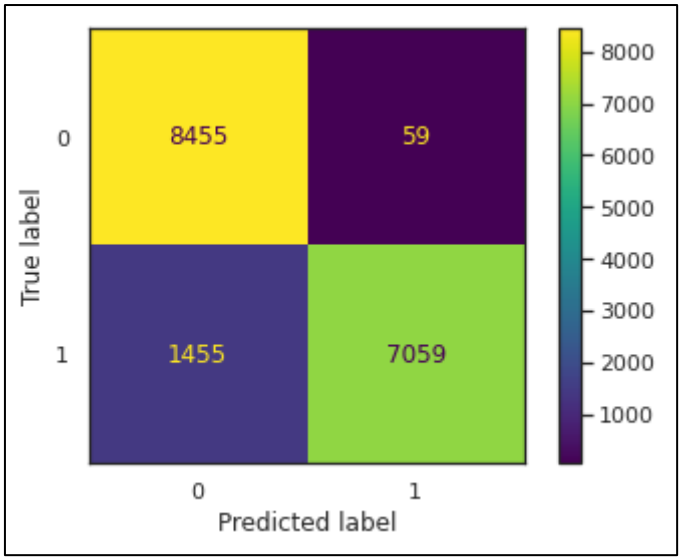
	precision	recall	f1-score	support
0	0.85	0.99	0.92	8514
1	0.99	0.83	0.90	8514
accuracy			0.91	17028
macro avg	0.92	0.91	0.91	17028
weighted avg	0.92	0.91	0.91	17028

Gambar 7. Hasil akurasi *random forest* dengan SMOTE

Selain akurasi, perbandingan juga akan dilakukan dengan *confusion matrix*. Perbandingan dapat dilihat pada gambar di bawah ini:



Gambar 8. *Confusion matrix random forest* tanpa SMOTE



Gambar 9. *Confusion matrix random forest* dengan SMOTE

Tabel berikut merupakan rangkuman hasil perbandingan dari algoritma *random forest* tanpa SMOTE dengan algoritma *random forest* dengan SMOTE.

Tabel 2. Perbandingan hasil algoritma *random forest*

Hasil	Random Forest tanpa SMOTE	Random Forest-SMOTE
Akurasi	0.98	0.91
Precision	0.69	0.92
Recall	0.51	0.91
F1-score	0.51	0.91

Dari tabel perbandingan hasil di atas, terdapat penurunan akurasi saat setelah dilakukan resampling menggunakan SMOTE, tetapi pada presisi, *recall*, *f1-score* terdapat kenaikan signifikan. Berdasarkan hasil yang didapat, disimpulkan penelitian deteksi menggunakan algoritma *random forest* dan SMOTE untuk mengatasi *imbalance* data mendapatkan hasil yang lebih tinggi dari penelitian sebelumnya.

4. KESIMPULAN

Berdasarkan hasil dan pembahasan pada penelitian tentang teknik SMOTE dalam mengatasi Imbalance data pada deteksi penyakit stroke menggunakan algoritma random forest, dapat diambil kesimpulan yaitu:

1. Penelitian menggunakan Algoritma *Random Forest* dan SMOTE untuk deteksi penyakit *stroke* menghasilkan akurasi sebesar 0.91, *precision* 0.92, *recall* 0.91, dan *f1-score* 0.91. Meskipun terjadi penurunan pada akurasi jika dibandingkan dengan *Random Forest* tanpa SMOTE, hal itu dinilai wajar terjadi saat melakukan *balance* pada *class* yang tidak seimbang, pada beberapa penelitian terkait ditemukan hal yang serupa.
2. Algoritma *Random Forest* efektif dalam mendeteksi penyakit *stroke* dalam jumlah data yang besar, dan metode SMOTE terbukti dapat mengatasi *imbalance class* pada dataset.

DAFTAR PUSTAKA

[1] S. Sutarwi, Y. Bakhtiar, and N. Rochana, "Sensitivitas dan Spesifitas Skor Stroke Literature Review," Gaster, vol. 18, no. 2, p. 186, 2020, doi: 10.30787/gaster.v18i2.521.
[2] World Stroke Organization, "Global Stroke Fact Sheet 2022," pp. 1–14, 2022.
[3] G. J. Hankey, "Stroke," Lancet, vol. 389, no. 10069, pp. 641–654, 2017, doi: 10.1016/S0140-6736(16)30962-X.
[4] H. Mukhtar, R. Muhammad, T. Reny Medikawati, and Yoze Rizki, "Peramalan Kedatangan Wisatawan Mancanegara Ke Indonesia Menurut Kebangsaan Perbulannya Menggunakan Metode Multilayer Perceptron," J. CoSciTech (Computer Sci. Inf. Technol., vol. 2, no. 2, pp. 113–119, 2021, doi: 10.37859/coscitech.v2i2.3324.
[5] A. Indrawati, "Penerapan Teknik Kombinasi Oversampling Dan Undersampling Untuk Mengatasi Permasalahan Imbalanced Dataset," JIKO (Jurnal Inform. dan Komputer), vol. 4, no. 1, pp. 38–43, 2021, doi: 10.33387/jiko.v4i1.2561.
[6] A. H. Hendri and Mochammad Arief Sutisna, "Article Desktop Based National Police Commission Activities Information System," J. CoSciTech (Computer Sci. Inf. Technol., vol. 2, no. 1, pp. 14–23, 2021, doi: 10.37859/coscitech.v2i1.2393.
[7] 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, p. 101723, 2019, doi: 10.1016/j.artmed.2019.101723.

[8] R. A. Nurdian, Mujib Ridwan, and Ahmad Yusuf, "Komparasi Metode SMOTE dan ADASYN dalam Meningkatkan Performa Klasifikasi Herregistrasi Mahasiswa Baru," J. Tek. Inform. dan Sist. Inf., vol. 8, no. 1, pp. 24–32, 2022, doi: 10.28932/jutisi.v8i1.4004.
[9] A. Faisal and A. Subekti, "JEPIN (Jurnal Edukasi dan Penelitian Informatika) Deep Neural Network untuk Prediksi Stroke," vol. 7, no. 3, pp. 443–449, 2021.
[10] A. N. Kasanah, M. Muladi, and U. Pujianto, "Penerapan Teknik SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Objektivitas Berita Online Menggunakan Algoritma KNN," J. RESTI (Rekayasa Sist. dan Teknol. Informasi), vol. 3, no. 2, pp. 196–201, 2019, doi: 10.29207/resti.v3i2.945.
[11] A. Primajaya and B. N. Sari, "Random Forest Algorithm for Prediction of Precipitation," Indones. J. Artif. Intell. Data Min., vol. 1, no. 1, p. 27, 2018, doi: 10.24014/ijaidm.v1i1.4903.
[12] W. Siburian, Vanissa and E. Mulyana, Ika, "Prediksi Harga Ponsel Menggunakan Metode Random Forest," Annu. Res. Semin., vol. 4, no. 1, pp. 144–147, 2018.