

Aplikasi Deteksi Dini Penyakit Stroke Menggunakan Stramlit

^{1*}Theo Krisna Amarya, ²Rina Firliana, ³Aidina Ristyawan

¹⁻³ Sistem Informasi, Universitas Nusantara PGRI Kediri

E-mail: ¹theokrisnaamarya@gmail.com, ²rina@unpkediri.ac.id, ³aidinaristi@unpkediri.ac.id

Penulis Korespondens : Rina Firliana

Abstrak— Di seluruh dunia, stroke adalah penyebab kematian dan kecacatan jangka panjang nomor dua. Untuk mengurangi risiko komplikasi dan meningkatkan peluang kesembuhan pasien, deteksi dini stroke sangat penting. Tujuan penelitian ini adalah untuk meningkatkan akurasi deteksi dini stroke dan menyediakan platform interaktif yang dapat digunakan oleh praktisi kesehatan maupun masyarakat umum dalam upaya pencegahan stroke secara lebih efektif. Dataset *Kaggle* dengan 5.110 sampel dan dua belas fitur digunakan. Implementasi model, pembagian dataset (80 persen pelatihan, 20 persen pengujian), dan pra-pemrosesan data (menangani nilai yang tidak ada, transformasi data, dan penyeimbangan kelas menggunakan *SMOTETomek*) adalah tahapan penelitian. Hasilnya menunjukkan akurasi sebesar 96% untuk kedua kelas—stroke dan non-stroke—dengan presisi dan *recall* yang seimbang. Untuk memfasilitasi visualisasi dan prediksi secara real-time, framework Streamlit digunakan dalam pengembangan aplikasi. Hasil penelitian menunjukkan bahwa kombinasi *AdaBoost-Random Forest* berhasil dalam klasifikasi stroke. Aplikasi berbasis web ini mungkin menjadi alat bantu yang bermanfaat untuk diagnosis stroke cepat.

Kata Kunci— *AdaBoost*, Deteksi Dini, Random Forest, *Streamlit*.

Abstract— Worldwide, stroke is the second leading cause of death and long-term disability. To reduce the risk of complications and increase patients' chances of recovery, early detection of stroke is essential. The aim of this study is to improve the accuracy of early stroke detection and provide an interactive platform that can be used by both healthcare practitioners and the general public to prevent stroke more effectively. The *Kaggle* dataset with 5,110 samples and twelve features was used. Model implementation, dataset division (80 percent training, 20 percent testing), and data pre-processing (handling missing values, data transformation, and class balancing using *SMOTETomek*) were the research stages. The results showed an accuracy of 96% for both classes—stroke and non-stroke—with balanced precision and recall. To facilitate real-time visualization and prediction, the *Streamlit* framework was used in the application development. The results showed that the *AdaBoost-Random Forest* combination was successful in stroke classification. This web-based application may be a useful tool for rapid stroke diagnosis.

Keywords— *AdaBoost*, Early Detection, Random Forest, *Streamlit*.

This is an open access article under the CC BY-SA License.



I. PENDAHULUAN

Stroke adalah Salah satu penyebab utama kecacatan jangka panjang dan kematian nomor dua di dunia [1]–[3]. Untuk mengurangi risiko komplikasi dan meningkatkan peluang kesembuhan pasien, sangat penting untuk mendeteksi potensi stroke sejak dini. Dalam bidang kesehatan,

penggunaan pembelajaran mesin telah berkembang menjadi metode yang menguntungkan untuk membantu pengambilan keputusan medis yang cepat dan tepat [4].

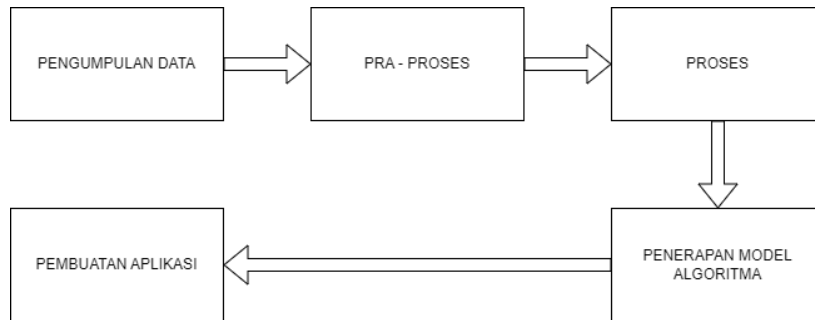
Salah satu teknik pembelajaran mesin yang umum digunakan dalam klasifikasi penyakit adalah *Random Forest*, sebuah algoritma berbasis pohon keputusan yang mampu menangani data dengan kompleksitas tinggi dan menghasilkan akurasi prediksi yang cukup baik [5], [6], *Random Forest* merupakan sebuah algoritma multifungsi berbasis pohon keputusan [7]. Namun demikian, kinerja algoritma ini masih dapat dioptimalkan lebih lanjut dengan menerapkan pendekatan *boosting*. *AdaBoost (Adaptive Boosting)* merupakan salah satu algoritma *boosting* yang terbukti efektif dalam meningkatkan performa model klasifikasi dengan menggabungkan beberapa *weak learners* menjadi model yang lebih kuat [8].

Dalam upaya mendekatkan hasil penelitian kepada pengguna akhir, *Streamlit*, sebuah *framework Python* untuk membangun aplikasi web interaktif, telah menjadi pilihan populer bagi para ilmuwan data dan insinyur pembelajaran mesin, seperti pada penelitian [9] *framework streamlit* digunakan untuk membuat *dashboard* kinerja program studi. Penelitian [10] menggunakan *frame streamlit* untuk memprediksi penurunan tekanan ban pada kendaraan pertanian, penelitian [11] menggunakan *frame work streamlit* untuk sistem rekomendasi tanaman, kemudian penelitian [12] menggunakan *framework* untuk memprediksi penyakit tanaman, penelitian [13] menggunakan *framework streamlit* untuk menganalisa sentimen pada platform *twitter* secara *realtime*. Dari hasil penelitian sebelumnya *framework* ini memungkinkan pengembangan antarmuka pengguna yang mudah dipahami dengan cepat tanpa memerlukan keahlian khusus dalam pengembangan web, sehingga sangat potensial untuk dimanfaatkan dalam penelitian ini sebagai sarana untuk *visualisasi* data dan penyajian hasil analisis secara *real-time*.

Penelitian ini adalah lanjutan dari penelitian yang diteliti oleh [5], [14], dengan menggabungkan algoritma *Random Forest* dan *Adaptive Boosting (AdaBoost)* dalam membangun sebuah aplikasi deteksi dini penyakit stroke berbasis machine learning menggunakan *framework Streamlit*. Tujuan utama penelitian ini adalah untuk meningkatkan akurasi deteksi dini stroke dan menyediakan platform interaktif yang dapat digunakan oleh praktisi kesehatan maupun masyarakat umum dalam upaya pencegahan stroke secara lebih efektif.

II. METODE

Metode yang digunakan penelitian ini adalah metode kuantitatif dengan pendekatan *experiment*, pendekatan *experiment* digunakan pada pembuatan model algoritma. Pendekatan *experiment* digunakan untuk menguji hubungan sebab akibat antara variabel bebas (*Independent*) dan variabel terikat (*Dependent*) [15]. Langkah – langkah metode penelitian ini dapat dilihat pada Gambar 2.1.



Gambar 1. Diagram Alur Penelitian

Pada gambar 1 dapat dideskripsikan sebagai berikut:

A. Pengumpulan Data

Data yang digunakan pada penelitian ini adalah dataset publik dengan nama “*stroke dataset*” yang diambil dari website *kaggle* [16]. Data ini terdiri dari 5110 baris dan 12 kolom serta sudah pernah dibuat dataset untuk penelitian yang dilakukan oleh [5], [17] yang dapat diunduh pada situs *kaggle* (<https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset>). Data pasien yang terkena stroke ditandai dengan angka 1 dan data pasien yang tidak terkena stroke ditandai dengan angka 0.

B. Pra Proses

Pra Proses data berguna untuk membersihkan dan mempersiapkan data agar dapat digunakan dalam analisis algoritma *Random Forest*. Proses ini juga disebut dengan pembersihan data (*cleanning Data*), pembersihan data yang dilakukan pada penelitian ini adalah mengisi data dari kolom dataset yang kosong (*null*) dengan rata rata kolom, menghapus data yang sama (*duplicated*) jika ada, menghapus nilai “*Other*” pada kolom gender, menghapus kolom “*Id*”, kemudian mengubah tipe data (*transform*) pada kategorikal (*non-numerik*) menjadi *number* (*numerik*). Setelah merubah tipe data, langkah berikutnya menyeimbangkan data menggunakan teknik *hybrid SMOTETomek*.

C. Proses

Setelah dataset sudah melalui proses praproses, langkah berikutnya ialah memilih variabel bebas dan variabel terikat, kemudian membagi dataset menjadi dua yaitu *data tranning* dan *data testing* dengan ukuran *data testing* sebesar 20% dari total dataset.

D. Penerapan Model Algoritma

Pada tahap ini, algoritma yang digunakan adalah *AdaBoost Random Forest*. Algoritma *Random Forest* adalah salah satu algoritma pembelajaran mesin yang sangat efektif. Karena kemampuannya untuk menyesuaikan diri dengan non-linearitas data, algoritma ini cocok untuk tugas prediksi. *Random Forest* dikatakan cocok untuk dataset sedang hingga besar. Jika jumlah variabel independen lebih besar dari jumlah observasi, algoritma ini dapat berfungsi karena tidak semua variabel prediktor digunakan sekaligus [7], [18]. Kemudian menyimpan model algoritma yang sudah dibuat dalam bentuk “.*pkl*”.

E. Pembuatan Aplikasi

Pembuatan aplikasi deteksi dini penyakit stroke akan melibatkan *framework* (kerangka kerja), yang berfungsi untuk memudahkan dalam merancang tampilan serta pemasangan algoritma *machine learning* ke aplikasi tersebut. Penelitian ini menggunakan *framework Streamlit*, *streamlit* adalah framework yang dibuat khusus untuk *Python* yang dimaksudkan untuk membangun aplikasi web dalam bidang pembelajaran mesin dan ilmu data. Setelah aplikasi web selesai dibuat, *Streamlit* memungkinkan pengguna menggunakan *platform cloud* untuk menginstal, mengelola, dan hosting aplikasi tersebut [13]. Setelah aplikasi sudah dibuat penulis juga melakukan testing pada aplikasi tersebut.

III. HASIL DAN PEMBAHASAN

Pembuatan aplikasi deteksi dini penyakit stroke dimulai dengan membuat model algoritma, pembuatan model algoritma menggunakan *tools jupyter notebook*, sedangkan untuk pembuatan program aplikasi deteksi dini penyakit stroke menggunakan *tools vscode* dengan bahasa pemrograman *python*. Adapun langkah langkah pembuatan model dapat dilihat pada deskripsi dibawah ini.

A. Persiapan Data

Membaca dataset yang didapatkan melalui website kaggle dengan menggunakan library *pandas* yang tersedia pada aplikasi *anaconda*, setelah dataset terbaca, terdapat 5110 pasien yang terkena stroke dan tidak stroke dan 12 fitur dimana kolom “stroke” merupakan variabel dependen (y) dan sisi kolom merupakan variabel independen (X). Dataset dimuat kedalam bentuk *dataframe* menggunakan library *Pandas* pada *python* agar mudah untuk dimanipulasi, analisis, dan visualisasikan. Atribut dataset tersebut dapat dilihat pada Tabel 1 berikut.

Tabel 1. Atribut Dataset

Kode	Variabel	Deskripsi
A	Id	Identitas pasien
B	Gender	"Laki laki", "Perempuan" atau "Lainnya"
C	Age	Usia Pasien
D	Hypertension	0 Jika Pasien Tidak memiliki, 1 Jika Pasien memiliki
E	Heart_disease	0 Jika Pasien Tidak memiliki, 1 Jika Pasien memiliki
F	Ever_married	"Tidak" atau "Iya"
G	Work_type	"Anak", "Pegawai Pemerintahan", "Tidak Bekerja", "Pribadi" or "Pegawai Swasta "
H	Residence_type	"Perkotaan" or "Perdesaan"
I	Avg_glucose_level	Rata rata gula darah
J	Bmi	body mass index (berat badan ideal)
K	Smoking_status	"Pernah Merokok", "Tidak Pernah merokok", "Perokok" atau "Tidak diketahui"
L	Stroke	0 Jika Pasien Tidak memiliki, 1 Jika Pasien memiliki

Dari tabel 1, isian dataset dapat dilihat pada gambar 2 berikut.

	id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	9046	Male	67.0	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
1	51676	Female	61.0	0	0	Yes	Self-employed	Rural	202.21	NaN	never smoked	1
2	31112	Male	80.0	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
3	60182	Female	49.0	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
4	1665	Female	79.0	1	0	Yes	Self-employed	Rural	174.12	24.0	never smoked	1
...
5105	18234	Female	80.0	1	0	Yes	Private	Urban	83.75	NaN	never smoked	0
5106	44873	Female	81.0	0	0	Yes	Self-employed	Urban	125.20	40.0	never smoked	0
5107	19723	Female	35.0	0	0	Yes	Self-employed	Rural	82.99	30.6	never smoked	0
5108	37544	Male	51.0	0	0	Yes	Private	Rural	166.29	25.6	formerly smoked	0
5109	44679	Female	44.0	0	0	Yes	Govt_job	Urban	85.28	26.2	Unknown	0
5110 rows x 12 columns												

Gambar 2. Isi dataset Stroke

Pada gambar 2, ternyata terdapat fitur “bmi” yang memiliki data kosong. Jumlah data yang hilang pada dataset dapat dilihat pada Tabel 2 berikut.

Tabel 2. Nilai Hilang pada dataset Stroke

Feature	Number of Missing Value
Id	0
Gender	0
Age	0
Hypertension	0
Heart_disease	0
Ever_married	0
Work_type	0
Residence_type	0
Avg_glucose_level	0
Bmi	201
Smoking_status	0
Stroke	0

Dari 5110 total record dataset, kolom “bmi” memiliki total 201 nilai yang hilang (kosong), seperti pada tabel 2. Kemudian penulis mengatasi *missing value* tersebut dengan menggunakan nilai rata-rata dari fitur “bmi”, dan juga menghapus kolom “Id”, hasil dari langkah ini dapat dilihat pada Tabel 3.

Tabel 3. Hasil Impitansi Missing Value

B	C	...	I	J	K	L
Female	61	...	202.21	28.8	never smoked	1
Male	80	...	105.92	32.5	never smoked	1

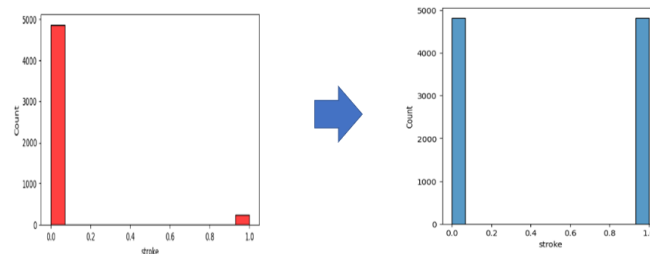
Female	49	...	171.23	34.4	smokes	1
Female	79	...	174.12	24	never smoked	1
...

Kemudian mentransformasikan fitur yang bernilai *non-numerik* (huruf) menjadi *numerik* dengan menggunakan teknik *Labelencoder*, hasilnya ditunjukkan pada Tabel 4. Langkah ini penting dilakukan agar dataset dapat dilatih menggunakan algoritma *Random Forest With AdaBoost*.

Tabel 4. Hasil Transformation

B	C	D	E	F	G	H	...	K	L
1	67	0	1	1	2	1	...	1	1
0	61	0	0	1	3	0	...	2	1
1	80	0	1	1	2	0	...	2	1
0	49	0	0	1	2	0	...	3	1
0	79	1	0	1	3	1	...	2	1

Setelah melakukan transformasi data, peneliti menemukan bahwa dataset stroke mengalami ketidakseimbangan, ketidakseimbangan data dapat menyebabkan penurunan performa algoritma *Random Forest*, sehingga perlu dilakukan metode penyeimbangan data dengan menggunakan teknik hybrid (*SMOTETomek*). Hasil sebelum dan sesudah penyeimbangan data dapat dilihat pada Gambar 3.



Gambar 3. Hasil Penyeimbangan Data

Langkah selanjutnya adalah membagi data menjadi dua bagian, 80% untuk data training dan 20% untuk data testing. Kemudian latih data training dengan skema parameter pada algoritma *Random Forest With AdaBoost* dan prediksi label menggunakan data testing.

B. Penerapan Model Algoritma

Dataset yang sudah selesai dibagi menjadi dua bagian, berikutnya akan di pakai dan di terapkan pada model, hasil klasifikasi yang berisi *accuracy*, *precision*, *recall*, dan *f1-score* dapat dilihat pada gambar 4 berikut.

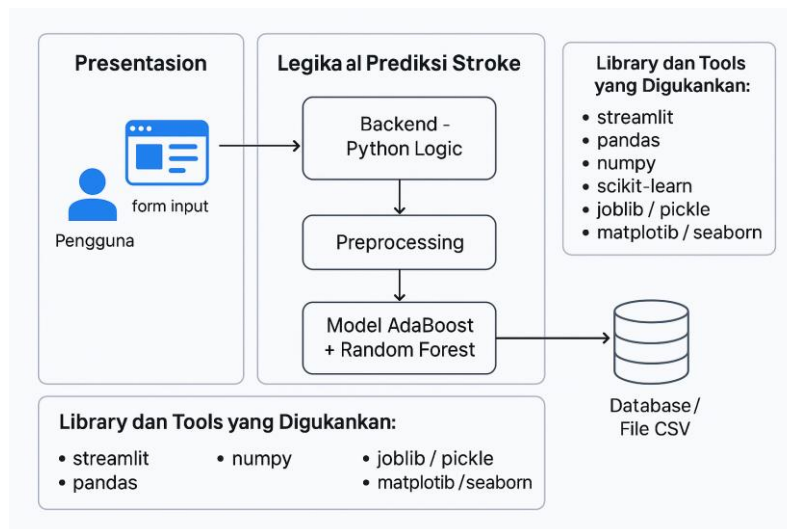
	precision	recall	f1-score	support
0	0.97	0.94	0.96	983
1	0.94	0.97	0.96	943
accuracy			0.96	1926
macro avg	0.96	0.96	0.96	1926
weighted avg	0.96	0.96	0.96	1926

Gambar 4. Hasil Klasifikasi *AdaBoost Random Forest*

Dari gambar 4, akurasi dari model yang dibuat memiliki nilai sebesar 96%, model ini memiliki presisi yang tinggi untuk label “tidak stroke” sebesar 97% dengan recall yang tidak terlalu tinggi dan rendah dengan nilai 94%. Model ini juga memiliki presisi untuk label “stroke” sebesar 94% dengan nilai recall yang tinggi sebesar 97%. Model ini memiliki nilai f1-score yang sama untuk label “tidak stroke” dan “stroke” sebesar 96%.

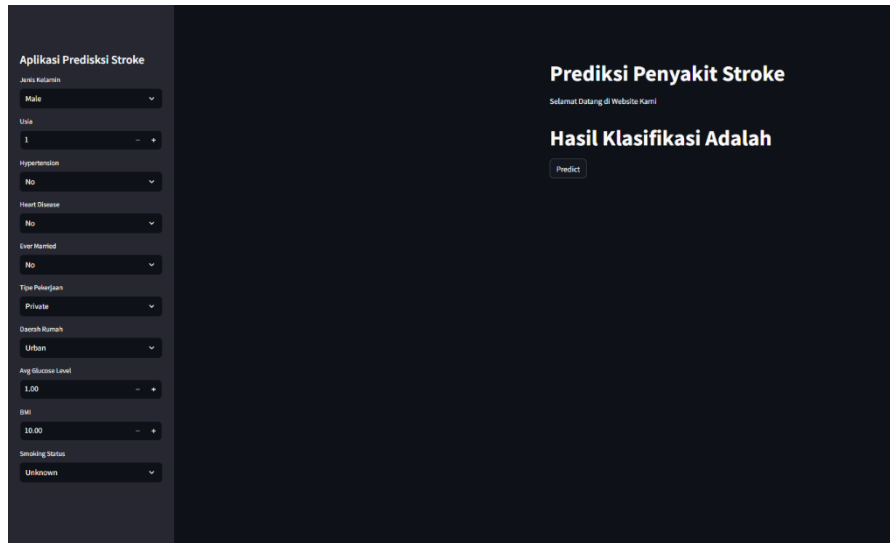
C. Pembuatan Aplikasi

Model algoritma *Random Forest with AdaBoost* diubah menjadi file “.pkl” dengan menggunakan modul *python pickle* untuk meningkatkan efisiensi eksekusi data. Pada penelitian ini, *Framework Streamlit* digunakan karena tidak membutuhkan berkas template khusus, tampilannya disusun di berkas utama *Python* dengan beberapa modul *Streamlit*, tampilan arsitektur aplikasi yang digunakan dapat dilihat pada gambar 5.



Gambar 5. Arsitektur Aplikasi

Dengan menggunakan fitur *Streamlit* yang disebut *Streamlit Share*, aplikasi dapat langsung di-hosting. Seluruh berkas kode aplikasi diunggah terlebih dahulu ke *Github*, dan kemudian berkas kode *Github* di-deploy melalui *Streamlit Share* untuk membuat aplikasi dapat diakses secara online oleh publik. Aplikasi ini dapat diakses melalui URL berikut: "https://strokedetect.streamlit.app/". Tampilan halaman Utama aplikasi ini dapat dilihat pada gambar 6.



Gambar 6. *Dashboard* Aplikasi

D. Testing Aplikasi

Pada tahapan ini penulis melakukan pengujian terhadap aplikasi yang dibuat, dengan data yang ada pada tabel 5 berikut.

Tabel 5. Data Uji Aplikasi

B	C	D	E	F	G	H	I	J	K	L
1	67	0	1	1	2	1	228.69	36.6	1	1
0	61	0	0	1	3	0	202	29	2	1
1	58	1	0	1	2	1	87.96	39.2	2	0
0	75	0	1	1	3	0	243.53	27.0	2	0
0	65	1	0	0	0	0	223.36	45.30	1	0

Dari tabel 5, aplikasi ini dapat menebak bahwa pasien mungkin ada resiko stroke dengan baik, namun pada penelitian ini masih belum menerapkan *pre-processing* yang lebih kompleks, belum menerapkan teknik menangani kebocoran data, dan tampilan website yang belum menarik.

IV. KESIMPULAN

Berdasarkan penelitian yang dilakukan, dapat disimpulkan bahwa *algoritma AdaBoost-Random Forest* memberikan performa yang baik dalam klasifikasi stroke dengan akurasi 96%, serta nilai *precision*, *recall*, dan *F1-score* yang seimbang untuk kedua kelas. Pra-pemrosesan data (penanganan *missing value*, transformasi kategorikal ke numerik, dan penyeimbangan data menggunakan *SMOTETomek*) berperan penting dalam meningkatkan kualitas dataset sebelum pemodelan. *Framework Streamlit* memudahkan pengembangan aplikasi web interaktif tanpa memerlukan keahlian khusus dalam pengembangan *front-end*, sehingga cocok untuk implementasi model pembelajaran mesin dalam dunia kesehatan. Aplikasi deteksi dini stroke

yang dikembangkan dapat memprediksi risiko stroke berdasarkan input pengguna, namun masih memiliki keterbatasan dalam hal tampilan antarmuka dan belum menerapkan teknik pencegahan kebocoran data (*data leakage*).

Saran untuk Pengembangan Selanjutnya menerapkan teknik “*feature engineering*” yang lebih mendalam untuk meningkatkan performa model. Menggunakan *deep learning* atau algoritma lain seperti *XGBoost* untuk perbandingan akurasi. Memperbaiki antarmuka aplikasi agar lebih interaktif dan informatif. Menambahkan validasi eksternal menggunakan dataset lain untuk memastikan generalisasi model. Serta mencari cara untuk menangani kebocoran data.

UCAPAN TERIMAKASIH

Penulis mengucapkan terima kasih kepada Dosen Pembimbing pertama dan kedua, karena bimbingannya, naskah ini dapat terbentuk serta terbit.

DAFTAR PUSTAKA

- [1] T. Y. Utomo, *BUKU AJAR STROKE*, 1st ed. Purbalingga: EUREKA MEDIA AKSARA, JUNI 2024 ANGGOTA IKAPI JAWA TENGAH NO. 225/JTE/2021, 2024.
- [2] W. Riyadina and E. Rahajeng, “Determinan Penyakit Stroke,” *Kesmas Natl. Public Heal. J.*, vol. 7, no. 7, p. 324, Feb. 2013, doi: 10.21109/kesmas.v7i7.31.
- [3] R. Adolph, “Apa itu Stroke?,” <https://p2ptm.kemkes.go.id/infographic-p2ptm/stroke/apa-itu-stroke>, 2016. <https://p2ptm.kemkes.go.id/infographic-p2ptm/stroke/apa-itu-stroke> (accessed Oct. 21, 2024).
- [4] E. Daniati, R. Firliana, and A. S. Wardani, *Dukungan Keputusan Dengan Pendekatan Machine Learning*. Kediri: Universitas Nusantara PGRI Kediri, 2024. doi: <https://doi.org/10.29407/punp.vi.68>.
- [5] T. K. Amarya, A. C. A. Galuh, R. Achmad, E. Daniati, and A. Ristyawan, “Analisa Perbandingan Algoritma Classification Berdasarkan Komposisi Label,” *Pros. SEMNAS INOTEK*, vol. 8, no. 1, pp. 32–40, Aug. 2024, doi: <https://doi.org/10.29407/inotek.v8i1.4906>.
- [6] A. Ristyawan, A. Nugroho, and T. K. Amarya, “Optimasi Preprocessing Model Random Forest Untuk Prediksi Stroke,” *J. Tek. Inform. dan Sist. Inf.*, vol. 12, no. 1, pp. 29–44, 2025, doi: <https://doi.org/10.35957/jatisi.v12i1.9587>.
- [7] L. Breiman, “Random Forests,” 2001.
- [8] A. Byna and M. Basit, “PENERAPAN METODE ADABOOST UNTUK MENGOPTIMASI PREDIKSI PENYAKIT STROKE DENGAN ALGORITMA NAÏVE BAYES,” *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 9, no. 3, pp. 407–411, Nov. 2020, doi: 10.32736/sisfokom.v9i3.1023.
- [9] M. Ilham, T. Siswanto, and D. Sugiarto, “Design of Study Program Performance Dashboard using Streamlit,” *INTELMATICS*, vol. 4, no. 2, pp. 96–100, 2024, doi: <https://doi.org/10.25105/v4i2.20643>.
- [10] R. Yadav and H. Raheman, “Machine learning-based estimation of agricultural tyre sinkage: A streamlit web application,” *J. Terramechanics*, vol. 119, no. December 2024, 2025, doi: 10.1016/j.jterra.2025.101055.
- [11] Y. Akkem, B. S. Kumar, and A. Varanasi, “Streamlit Application for Advanced Ensemble Learning Methods in Crop Recommendation Systems – A Review and Implementation,” *Indian J. Sci. Technol.*, vol. 16, no. 48, pp. 4688–4702, 2023, doi: 10.17485/ijst/v16i48.2850.
- [12] M. K. Warbhe, J. J. Bore, and S. N. Chaudari, “A Deep Learning Based System to Predict the Plant Disease Using Streamlit,” *Int. Conf. Sentim. Anal. Deep Learn.*, 2025, doi: <https://doi.org/10.1109/ICSADL65848.2025.10933389>.
- [13] S. Patil and V. Loksha, “Live Twitter Sentiment Analysis Using Streamlit Framework,” *SSRN Electron. J.*, 2022, doi: 10.2139/ssrn.4119949.
- [14] X. Huang *et al.*, “Novel Insights on Establishing Machine Learning-Based Stroke Prediction Models Among Hypertensive Adults,” *Front. Cardiovasc. Med.*, vol. 9, no. May, pp. 1–11, 2022, doi: 10.3389/fcvm.2022.901240.
- [15] K. Abdullah *et al.*, *Metodologi Penelitian Kuantitatif*. Yayasan Penerbit Muhammad Zaini, 2022.

- [Online]. Available: <https://penerbitzaini.com/>
- [16] Fedesoriano, “Stroke Prediction Dataset,” <https://www.kaggle.com/fedesoriano>, 2020. <https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset> (accessed Oct. 04, 2024).
- [17] M. S. Pathan, Z. Jianbiao, D. John, A. Nag, and S. Dev, “Identifying Stroke Indicators Using Rough Sets,” *IEEE Access*, vol. 8, pp. 210318–210327, Nov. 2020, doi: 10.1109/ACCESS.2020.3039439.
- [18] M. Schonlau and R. Y. Zou, “The random forest algorithm for statistical learning,” *Stata J.*, vol. 20, no. 1, pp. 3–29, 2020, doi: 10.1177/1536867X20909688.