Jurnal Teknik Informatika (JUTIF)
Vol. 5, No. 4, August 2024, pp. 991-1000

DOI: https://doi.org/10.52436/1.jutif.2024.5.4.1951
p-ISSN: 2723-3863
e-ISSN: 2723-3871

COMPARISON OF K-NEAREST NEIGHBOR AND SUPPORT VECTOR MACHINE ALGORITHM OPTIMIZATION WITH GRID SEARCH CV ON STROKE PREDICTION

Wahyu Aprilliandhika[1], Ferian Fauzi Abdulloh[*2]

[1,2]Informatics, Computer Science Faculty, Universitas Amikom Yogyakarta, Indonesia
Email: [1]wahyuaprilliandhika@students.amikom.ac.id, [2]ferian@amikom.ac.id

(Article received: March 21, 2024; Revision: April 06, 2024; published: July 29, 2024)

Abstract

Stroke ranks second as the leading cause of death globally, with disability being the primary accompanying factor. The cause of death in stroke patients is due to the lack of an optimal stroke prediction system; therefore, identifying whether a patient is experiencing a stroke or not becomes the focus of this research. Thus, the objective of this study is to compare the performance of stroke prediction using two classification models, namely K-Nearest Neighbors (KNN) and Support Vector Machine (SVM), with and without using the GridSearchCV optimization technique. In this experiment, the dataset is processed and divided into training and testing data using the SMOTE oversampling technique. Initial testing is conducted without GridSearchCV. The results of the initial testing show that the KNN model performs better than SVM, with accuracies of 91% and 83%, respectively. After optimizing parameters using GridSearchCV, both models experience a significant performance improvement. The KNN model increases accuracy to 95% with precision of 91% and recall of 98%, while the SVM model increases accuracy to 94% with precision of 90% and recall of 99%. These results indicate that using GridSearchCV to optimize parameters of KNN and SVM models can significantly enhance stroke prediction performance. There are differences in precision and recall between KNN and SVM. The KNN model tends to have higher recall, while the SVM model has higher precision, and for accuracy, the KNN algorithm outperforms SVM in stroke prediction.

Keywords: *Comparison, GridSearchCV, K-Nearest Neighbor, Optimization, Support Vector Machine*.

KOMPARASI OPTIMASI ALGORITMA K-NEAREST NEIGHBOR DAN SUPPORT VECTOR MACHINE DENGAN GRID SEARCH CV PADA PREDIKSI STROKE

Abstrak

Stroke menempati peringkat kedua sebagai penyebab kematian global terbanyak, dengan kecacatan sebagai penyebab utama yang menyertai kondisi tersebut. Penyebab kematian pada penderita stroke disebabkan tidak ada sistem yang memprediksi stroke secara optimal, identifikasi apakah seorang pasien mengalami stroke atau tidak menjadi fokus penelitian ini. Maka, tujuan dari penelitian ini membandingkan kinerja prediksi stroke menggunakan dua model klasifikasi yang digunakan, yaitu *K-Nearest Neighbors* (KNN) dan *Support Vector Machine* (SVM), dengan dan tanpa menggunakan teknik optimasi *GridSearchCV*. Dalam eksperimen ini, dataset diproses dan dibagi menjadi data pelatihan dan pengujian dengan menggunakan teknik *oversampling SMOTE*. Pengujian awal dilakukan tanpa *GridSearchCV*. Hasil pengujian awal menunjukkan bahwa model KNN memiliki performa lebih baik daripada SVM, dengan akurasi 91% dan 83%. Setelah mengoptimalkan parameter menggunakan *GridSearchCV*, kedua model mengalami peningkatan signifikan dalam performa. Model KNN meningkatkan akurasi menjadi 95% dengan presisi 91% dan *recall* 98%, sementara model SVM meningkatkan akurasi menjadi 94% dengan presisi 90% dan *recall* 99%. Hasil penelitian ini menunjukkan bahwa penggunaan *GridSearchCV* dalam mengoptimalkan parameter model KNN dan SVM dapat meningkatkan kinerja prediksi stroke secara signifikan. Terdapat perbedaan dalam aspek presisi dan recall antara KNN dan SVM. Model KNN cenderung memiliki recall yang lebih tinggi, sementara model SVM memiliki presisi yang lebih tinggi, dan untuk akurasi algoritma KNN lebih unggul dibandingkan dengan SVM dalam prediksi Stroke.

Kata kunci: *GridSearchCV, K-Nearest Neighbor, Komparasi, Optimasi, Support Vector Machine.*

1. PENDAHULUAN

Stroke adalah suatu keadaan yang terjadi saat aliran darah menuju otak terhambat, mengakibatkan gangguan dalam fungsi saraf [1]. Stroke menempati peringkat kedua sebagai penyebab kematian global terbanyak, dengan kecacatan sebagai penyebab utama

yang menyertai kondisi tersebut. Menurut *World Health Organization* (WHO), sekitar 70% kasus stroke secara global terjadi karena kecacatan. Lebih dari 87% dari jumlah kematian akibat stroke terjadi di negara-negara yang memiliki tingkat pendapatan rendah dan menengah [2]. Di Indonesia, 43,1% stroke umumnya terdeteksi pada individu yang berusia 75 tahun ke atas, dengan insiden mencapai 0,2%. Di sisi lain, terdapat angka yang serupa pada kelompok usia 15-24 tahun. Jika stroke tidak diberi penanganan yang memadai, dapat memiliki konsekuensi fatal [3]. Keadaan berbahaya ini memerlukan penanganan cepat dalam memprediksi stroke secara effisien dan akurat [4].

Mendalaminya risiko stroke di Indonesia mengungkap urgensi masalah kesehatan masyarakat. Data epidemiologi menunjukkan bahwa stroke bukan hanya beban kesehatan signifikan, tetapi juga penyebab utama kematian dan cacat jangka panjang. Urgensi penelitian ini terkait dengan dampak luas risiko stroke terhadap individu dan masyarakat. Dengan prevalensi yang terus meningkat, pemahaman mendalam tentang faktor-faktor yang mempengaruhi risiko stroke menjadi krusial untuk mengembangkan strategi pencegahan yang lebih efektif.

Prediksi dini stroke menjadi fokus utama pencegahan kesehatan masyarakat, dengan metode umum klasifikasi menggunakan metode *K-Nearest Neighbor* (KNN) adalah suatu pendekatan yang simpel yang memanfaatkan data dari tetangga terdekat untuk tujuan klasifikasi, KNN memiliki beberapa kelebihan, seperti mudah diimplementasikan dan dapat digunakan untuk data yang kompleks [5]. Dalam tahap ini, algoritma KNN menggunakan sistem klasifikasi berdasarkan kedekatan tetangga sebagai nilai prediksi, di mana *Euclidean Distance* digunakan untuk menilai seberapa dekat antara objek-objek tersebut. Metode ini juga umumnya memanfaatkan pengukuran jarak yang sering digunakan pada data numerik [6] Namun, KNN juga memiliki beberapa kelemahan, seperti sensitif terhadap data yang tidak seimbang dan memiliki beberapa parameter yang harus dioptimalkan agar dapat memberikan hasil prediksi yang akurat. Meskipun tingkat efektivitasnya sangat tergantung pada factor-faktor seperti nilai K dan jenis fungsi jarak. Khususnya dalam klasifikasi stroke, performa akurasi KNN dipengaruhi oleh jumlah fitur dan faktor-faktor lainnya. Sebagai contoh, dalam studi perbandingan antara KNN dan regresi logistik untuk klasifikasi diabetes, KNN mencapai akurasi 85,06%, sedangkan regresi logistik mencapai 77,92% [7]. Studi lain yang mengevaluasi penggunaan KNN dalam konteks data karyawan yang tidak seimbang untuk keperluan klasifikasi promosi menunjukkan bahwa KNN mencapai akurasi 85,57%, yang merupakan tingkat tertinggi [8]. Selain KNN, SVM merupakan metode klasifikasi untuk data linear dan non-linear. SVM memanfaatkan fungsi kernel yang secara sistematis membuat *support vector classifier* dalam bentuk dimensi yang lebih tinggi [9]. Secara umum, SVM bekerja dengan mencari *hyperplane* yang memiliki jarak maksimum dari kedua kelas data. Peneliti dalam bidang bioinformatika sering memanfaatkan SVM sebagai alat untuk memprediksi penyakit, seperti diabetes mellitus, Tingkat akurasi yang tinggi telah berhasil dicapai, dengan akurasi mencapai 91.2%, presisi 93.0%, *recall* 94.3%, dan skor F1 93.7%. [10]. Studi lain mengaplikasikan SVM dalam upaya memprediksi kemungkinan terjadinya bencana banjir. Hasil akurasi yang diperoleh menggunakan algoritma SVM yang dioptimalkan dengan *Particle Swarm Optimization* (PSO) adalah sebesar 97,62% [11]. Dengan merujuk pada temuan-temuan tersebut, dapat disimpulkan bahwa KNN dan SVM memiliki tingkat ketepatan yang superior jika dibandingkan dengan berbagai algoritma lain.

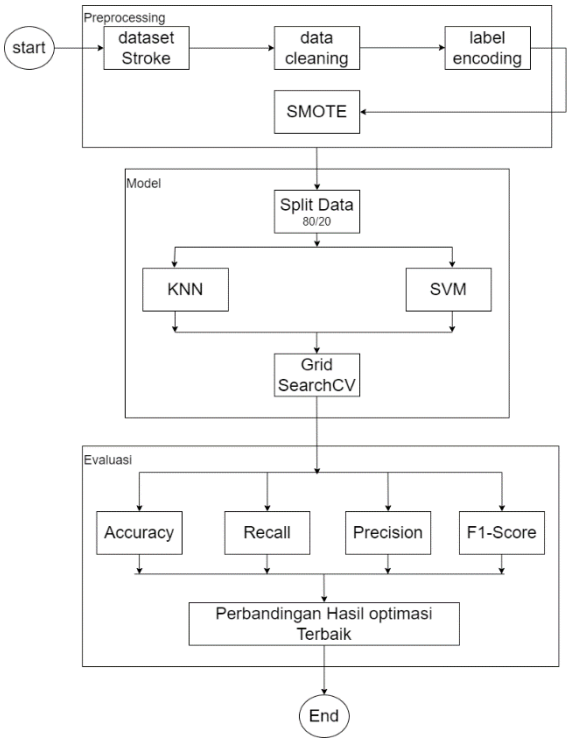
Dalam kerangka penelitian ini, akan melaksanakan penerapan metode *Grid Search CV*, suatu pendekatan yang memfasilitasi pencarian nilai optimal dengan menggabungkan *grid search* dan *cross-validation*. Metode ini bekerja dengan cara membagi data menjadi beberapa bagian dan melakukan validasi silang pada setiap kombinasi parameter. Dengan menggunakan *Grid Search CV*, parameter pada algoritma KNN dan SVM dapat dioptimalkan sehingga dapat memberikan hasil prediksi yang lebih akurat pada prediksi stroke. Pada metode ini pernah digunakan pada Efisien Diagnosis Medis Penyakit Jantung Manusia Menggunakan Teknik *Machine Learning* Dengan dan Tanpa *GridSearchCV*. mendapatkan akurasi sebesar 99% [12]. Penelitian lain juga berhasil meningkatkan tingkat akurasi dengan menggunakan *Grid Search CV* pada studi Pendekatan Pembelajaran Mesin untuk Mengidentifikasi Penyakit Parkinson dengan Menggunakan Fitur Sinyal Suara. sebesar 95% dan pada MLP mencapai 98,31% [13]. Pada penelitian Meningkatkan Akurasi Prediksi Penyakit Jantung melalui Teknik Pembelajaran Mesin dan Optimalisasi. juga mendapatkan hasil akurasi yang baik sebesar 95 % meningkat 5 % sebelum di optimasi [14]. mengoptimalkan *Support Vector Machine* (SVM) dalam analisis ulasan e-commerce dengan Grid Search dan Unigram, meningkatkan akurasi hingga 80.80% [15]. Penggunaan Grid Search juga berhasil meningkatkan akurasi sebesar 3 sampai 4% pada pendeteksi kanker payudara dengan memilih hyperparameter terbaik menerapkan metode Grid Search pada algoritma *K Nearest Neighbor* [16].

Berdasarkan permasalahan yang dihadapi tersebut, maka penelitian ini bertujuan untuk membandingkan tingka kinerja prediksi stroke dengan mengoptimalkan algoritma *K-Nearest Neighbor* dan *Support Vector Machine*, dengan mengusulkan pemanfaatan *GridSearchCV*. Penelitian ini membandingkan tingkat optimasi dari kedua model *K-Nearest Neighbor* (KNN) dan *Support*

Vector Machine (SVM) dalam memprediksi risiko stroke dengan penekanan pada faktor-faktor yang memengaruhi prediksi dini stroke. Metode KNN dan SVM digunakan secara khusus untuk klasifikasi stroke berdasarkan dataset yang tersedia. Evaluasi performa dari kedua algoritma akan difokuskan pada variabel seperti jumlah fitur, ketidakseimbangan data, serta pembagian dataset antara data pelatihan dan pengujian, bersamaan dengan penentuan nilai K untuk KNN dan parameter optimal untuk SVM yang dapat mempengaruhi akurasi prediksi yang dihasilkan. Penelitian ini menggunakan informasi yang tersedia publik mengenai stroke. Dengan mengoptimalkan kedua algoritma melalui teknik *GridSearchCV* dan mempertimbangkan aspek interpretatif, diharapkan penelitian ini dapat memberikan kontribusi yang bermakna dalam pengembangan model prediktif untuk mendeteksi stroke.

2. METODE PENELITIAN

Dalam penelitian ini, penggunaan *GridSearchCV* dimanfaatkan untuk mengoptimalkan metode *k-nearest neighbor* dan *support vector machine* dalam prediksi stroke. Langkah-langkah penelitian ini dapat disimak melalui Gambar 1 berikut ini.



Gambar 1. Diagram Penelitian

2.1. Preprocessing

Preprocessing data merupakan langkah awal dalam proses data mining yang bertujuan untuk mengubah data mentah menjadi format dan informasi yang lebih efisien dan bernilai [16]. Dalam fase data mining, penting untuk melakukan preprocessing data karena tidak semua data atau atribut dalam kumpulan data diperlukan dalam proses tersebut. Tujuan dari proses ini adalah untuk menyelaraskan data yang akan digunakan dengan kebutuhan analisis yang diinginkan [17].

2.1.1. Dataset Stroke

Dataset stroke berasal dari sumber data publik di *Kaggle* yang terdiri dari 5110 entri data dengan 12 fitur [18]. Tipe data pada setiap attribut pada dataset stroke disajikan dalam Tabel 1.

Tabel 1.Dataset Stroke

No	Attribut	Tipedata
1	id	Numeric
2	gender	string
3	age	numeric
4	hypertension	numeric
5	heart_disease	numeric
6	ever_married	string
7	work_type	string
8	residence_type	string
9	avg_glucose_level	numeric
10	bmi	numeric
11	smoking_status	numeric
12	stroke	numeric

2.1.2. Data Cleaning

Pada tahap data cleaning, tindakan-tindakan seperti menghilangkan data duplikat, mengecek konsistensi data, dan melakukan koreksi terhadap kesalahan pada dataset merupakan bagian dari proses tersebut [19].

2.1.3. Label Encoding

Encoding merupakan proses mengonversi data dari satu bentuk ke bentuk lain agar dapat diolah atau disimpan dengan lebih efisien dan format yang dapat dipahami oleh *machine* [20]. Pada dataset penelitian ini terdapat attribute yang memerlukan encoding agar data dapat di proses dalam data mining.

2.1.4. SMOTE

Penanganan ketidakseimbangan kelas pada dataset terjadi ketika jumlah instansi atau sampel pada satu kelas jauh lebih banyak atau lebih sedikit daripada kelas lainnya. Untuk mengatasi masalah ini, diterapkan metode atau teknik khusus agar model pembelajaran mesin tidak cenderung memihak pada kelas mayoritas, yang dapat menyebabkan kinerja yang bias. Teknik *SMOTE* digunakan untuk mengatasi ketidakseimbangan data dengan mengadopsi *oversampling* dan mempertimbangkan dampaknya terhadap wilayah keputusan pada konteks yang bersangkutan. Berbeda dengan pendekatan undersampling, penerapan *SMOTE* dianggap lebih efektif karena tidak melibatkan pengurangan jumlah data [21].

2.2. Model

Setelah Melakukan preprocessing data, pada tahap ini, diterapkan model *K-Nearest Neighbors* (KNN) dan *Support Vector Machine* (SVM) sebagai algoritma pembelajaran mesin untuk memprediksi stroke. Dan setelah itu melakukan optimasi menggunakan *Grid Search*.

2.2.1. Algoritma K-Nearest Neighbor

Algoritma *K-Nearest Neighbors* (KNN) metode klasifikasi yang bekerja berdasarkan prinsip bahwa data dengan atribut yang serupa cenderung berada dalam kelompok yang sama atau memiliki label yang mirip. Algoritma ini melakukan pengambilan nilai K data terdekat dengan membandingkan data baru dengan kategori data yang sudah ada dalam dataset, kemudian memberikan label berdasarkan mayoritas tetangga terdekatnya [22]. Untuk mengukur jarak antara dua objek, menggunakan rumus Euclidean seperti yang terlihat pada persamaan 1.

$$dist = \sqrt{\sum(\mathcal{X}_1 + \mathcal{X}_2)} \quad (1)$$

Dalam konteks ini, dist merujuk pada jarak Euclidean untuk X1 yang merupakan data pelatiha n, sementara pada X2, merujuk pada data pengujian, Selanjutnya, dalam algoritma KNN, nilai K ditetapkan sebagai parameter, yang mencakup jumlah tetangga terdekat. Untuk data pengujian, dilakukan perhitungan jarak Euclidean terhadap setiap data pelatihan.

2.2.2. Support Vector Machine

SVM merupakan salah satu teknik dalam *supervised learning* yang digunakan untuk menganalisis data serta menemukan pola dalam proses pengelompokan data. *Support Vector Machine* (SVM) mengubah teks menjadi representasi vektor sebelum diterapkan dalam proses klasifikasi [23]. Konsep dasar dari SVM adalah untuk menemukan garis keputusan (*hyperplane*) yang optimal bagi setiap titik data. Proses klasifikasi melibatkan dua tahap utama, yakni tahap pengujian dan tahap pelatihan. Tahap pelatihan berfungsi untuk menciptakan sebuah model yang akan digunakan dalam pengujian berikutnya [*24*].

2.2.3. Grid Search CV

Proses ini melibatkan penghitungan ukuran akurasi untuk setiap kombinasi parameter pada grid tersebut, Dengan maksud untuk menemukan nilai parameter yang optimal, yaitu nilai yang menghasilkan tingkat akurasi terbaik [25]. *Cross-validation* merupakan metode pengembangan dari validasi pemisahan model, di mana evaluasinya dilakukan dengan mengukur kesalahan pelatihan pada data uji [26]. Dalam riset ini, dilakukan penggunaan cross-validation dengan jumlah *fold* sebanyak 10.

2.3. Evaluasi

Langkah pertama dalam proses ini adalah memisahkan data dalam setiap kasus menjadi dua bagian, data latih dan data uji. Dan data latih berperan sebagai data referensi dalam perhitungan setiap algoritma, sementara data uji digunakan untuk mengevaluasi keakuratan prediksi dan keputusan yang dihasilkan oleh masing-masing algoritma. Dalam penelitian ini, penilaian performa model dilakukan melalui *confusion matrix*. *Confusion matrix* adalah tabel yang mengilustrasikan hasil kinerja suatu model atau algoritma, di mana setiap baris mencerminkan kelas sebenarnya dan setiap kolom mencerminkan kelas yang diprediksi oleh model [27].

Tabel 2. *Confussion Matrix*

TN (True Negatif)	FP (False Positif)
FN (False Negatif)	TP (True Positif)

Dari gambar Tabel 2 Dapat digunakan untuk menghitung *Accuracy, recall, precision, F1-score*.

2.3.1. Accuracy

Accuracy adalah ukuran sejauh mana model mampu melakukan klasifikasi dengan tepat. Rumus untuk menghitung akurasi dapat dalam persamaan 2.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

2.3.2. Precision

Precision adalah mengukur sejauh mana model melakukan prediksi yang akurat sesuai dengan data prediksi. Formula untuk menghitung nilai presisi dapat dinyatakan dalam persamaan 3.

$$Precision = \frac{TP}{FP+TP} \quad (3)$$

2.3.3. Recall

Recall mengukur sejauh mana model melakukan prediksi positif saat kelas aktualnya juga positif. Formula untuk menghitung nilai recall dapat diidentifikasi dalam Persamaan 4.

$$Recall = \frac{TP}{FN+TP} \quad (4)$$

2.3.4. F1-Score

F1-score adalah nilai rata-rata dari presisi dan recall. Formula untuk menghitung nilai *F1-score* dapat ditemukan dalam Persamaan 5.

$$F1 - Score = 2 * \frac{Precision+recall}{Precision+recall} \quad (5)$$

3. HASIL DAN PEMBAHASAN

Dalam penelitian ini, data yang digunakan diambil dari platform *kaggle*. Dataset ini terdiri dari 12 kolom atau atribut dan mencakup 5110 baris atau entri data pasien. Dataset tersebut digunakan untuk melakukan klasifikasi terhadap kasus stroke dengan dua target yang mungkin, yaitu "1" (menderita) dan "0" (tidak menderita). Informasi tentang sebaran data tersaji dalam Tabel 3.

Tabel 3. Data Stroke

Stroke	Jumlah
1	202
0	3364

3.1. *Preprocessing* Data

Pada tahapan penelitian ini dataset harus melakukan beberapa proses dalam preprocessing data yang meliputi tahapan data cleaning, encoding, dan SMOTE. Sebelum dilakukan tahapan data cleaning, kolom smoking_status terdapat bernilai *Uknown* seperti pada Tabel 4. Maka data yang bernilai uknown tersebu kemudian dihapus.

Tabel 4. Data bernilai Uknown

No	Smoking_status
1	never smoked
2	Unknown
3	formerly smoked
4	smokes

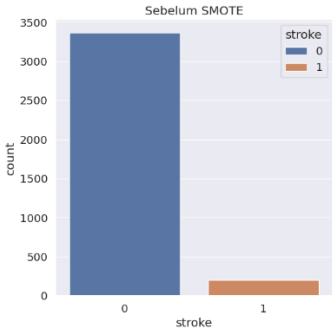
Pada tahap data cleaning kolom id juga di hapus. Setelah melakukan data cleaning tahapan selanjutnya encoding. Pada langkah ini, tipe data dalam dataset diubah menjadi bentuk numerik, khususnya dalam bentuk bilangan bulat.

Tabel 5. Kolom yang bertipe data binary dan kategorikal

Kolom data Binary	Kolom data Kategori
gender	work_type
hypertension	smoking_status
heart_disease	
ever_married	
Residence_type	
stroke	

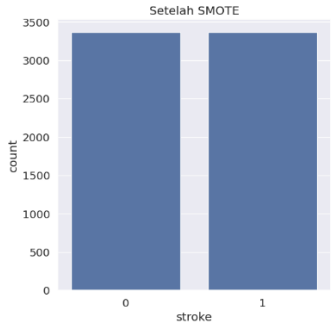
Dari 11 kolom yang ada pada dataset terdapat 6 kolom terdapat data biner dalam Tabel 5 dan ada juga tiga kolom yang berisi data kategori. Kolom yang berjenis binary kemudian diubah dengan *library label encoder*. Langkah encoding ini memiliki manfaat dalam persiapan dataset untuk proses analisis lebih lanjut, terutama pada model-machine learning yang memerlukan input dalam format numerik. Hal ini mempermudah model untuk mengenali dan memahami pola yang mungkin terdapat dalam data. Dengan menerapkan encoding, dataset menjadi lebih siap untuk diintegrasikan dalam berbagai algoritma machine learning yang memerlukan data numerik.

Tahapan selanjutnya setelah melakukan data cleaning, dan encoding yaitu melakukan handling imbalance data, hal ini disebabkan karena data yang kurang seimbang. Maka untuk mengatasi hal ini digunakan SMOTE untuk menerapkan metode oversampling dalam menangani ketidakseimbangan data. Data yang tidak seimbang merupakan data stroke.



Gambar 2. Sebelum SMOTE

Dalam Gambar 2, terlihat ketidakseimbangan antara data yang bernilai 0 dan 1, di mana 0 memiliki lebih dari 100 data, sedangkan 1 kurang dari 40 data. Untuk mengatasi hal ini, digunakan metode SMOTE sehingga distribusi data pada Gambar 3 menjadi lebih seimbang. Data 1 yang awalnya sebelum di SMOTE hanya memiliki kurang dari 40 data dan setelah dilakukan SMOTE data menjadi seimbang. Dengan menerapkan metode SMOTE, data sintetis dibuat untuk kelas minoritas dengan cara mengambil rata-rata dari beberapa tetangga terdekat. Ini membantu meningkatkan jumlah sampel kelas minoritas sehingga mendekati jumlah kelas mayoritas, menciptakan distribusi data yang lebih seimbang. Penting untuk dicatat bahwa implementasi SMOTE ini dilakukan tanpa mengurangi kualitas dari data asli, sehingga memastikan dataset tetap mencerminkan variasi dan kompleksitas yang ada dalam data sebelumnya.



Gambar 3. Setelah SMOTE

Secara keseluruhan, tahap preprocessing data merupakan fondasi penting dalam penelitian ini untuk memastikan keakuratan, reliabilitas, dan keseragaman dataset sebelum diterapkan pada model-machine learning. Proses data cleaning membersihkan dataset dari nilai yang tidak informatif, encoding mengubah tipe data menjadi

format yang lebih seragam dan dapat diolah oleh model, sementara SMOTE membantu mengatasi ketidakseimbangan kelas dalam dataset. Dengan tahapan preprocessing yang baik, dataset siap untuk dijelajahi lebih lanjut dalam pengembangan model.

3.2. Hasil Pelatihan dan Pengujian Model

Setelah melalui tahap preprocessing, dataset kemudian dibagi menjadi data pelatihan dan data pengujian. Dalam penelitian ini, hasil pengujian dilakukan dengan membagi data latih 80% dan data uju 20%. Pembagian data tersebut merupakan hasil dari SMOTE *oversampling*. Data latih terbagi menjadi dua bagian, yaitu *X_train* yang berperan sebagai variabel independen yang memengaruhi target, dan juga *y_train* yang berperan sebagai variabel dependen yang menjadi target. Data latih dimanfaatkan untuk melatih algoritma KNN oleh sistem, kemudian hasil latihan tersebut disimpan untuk keperluan penggunaan selanjutnya dalam proses pelatihan. Berikut data *X_train* dan *y_train* disajikan pada Tabel 6 dan Tabel 7 berikut.

Tabel 6. Data *X_train*

No	X_train	Jumlah
1	gender	5382
2	age	5382
3	hypertension	5382
4	heart_disease	5382
5	ever_married	5382
6	work_type	5382
7	residence_type	5382
8	avg_glucose_level	5382
9	bmi	5382
10	smoking_status	5382
Total		53.820

Tabel 7. Data *y_train*

No	y_train	Jumlah
1	stroke	5382
Total		5382

Langkah berikutnya adalah melakukan pengujian data, yang kemudian akan dievaluasi untuk menghasilkan nilai akurasi menggunakan metode confusion matrix. Untuk data uji terdapat pada Tabel 8 dan Tabel 9 berikut.

Tabel 8. Data *X_test*

No	X_test	Jumlah
1	gender	1346
2	age	1346
3	hypertension	1346
4	heart_disease	1346
5	ever_married	1346
6	work_type	1346
7	residence_type	1346
8	avg_glucose_level	1346
9	bmi	1346
10	smoking_status	1346
Total		13.460

Tabel 9. Data *y_test*

No	y_test	Jumlah
1	stroke	1346
Total		1346

Pada pengujian awal, dilakukan klasifikasi algoritma KNN dan SVM tanpa memanfaatkan Grid Search. Dalam eksperimen ini, nilai K yang digunakan adalah yang terkecil, yaitu 3. dan untuk SVM Parameter regularisasi sebesar 100, dan kernel nya default rbf, dan gamma nya auto.

Tabel 10. Hasil implementasi knn dan svm sebelum optimasi Grid Search

Model	Precision	Recall	F1-Score	Accuracy
KNN	0,85	0,90	0,91	0,91
SVM	0,78	0,89	0,84	0,83

Pengujian terdokumentasi dalam Tabel 10. Evaluasi performa model menggunakan metrik precision, recall, F1 score, dan akurasi diperoleh dengan nilai yang berbeda-beda. Dalam model KNN menunjukkan tingkat presisi mencapai 0,85, recall mencapai 0,90, F1 Score mencapai 0,91, dan akurasi mencapai 0,91. Sementara itu, pada model SVM menunjukkan tingkat presisi mencapai 0,78, *recall* mencapai 0,89, *F1-Score* mencapai 0,84, dan akurasi mencapai 0,83.

3.3. Hasil optimasi model dengan GridSearchCV

Dalam penelitian ini, model *K-Nearest Neighbors* (KNN) dan *Support Vector Machine* (SVM) dievaluasi dan dioptimalkan menggunakan metode *Grid Search Cross-Validation*. Metode evaluasi performa model dilakukan dengan mengukur *Precision, Recall, F1-Score, dan Accuracy*. model KNN diinisialisasi dengan parameter default, dan kemudian dilakukan penalaan parameter untuk menemukan kombinasi parameter terbaik. *Grid Search* melibatkan eksplorasi nilai K yang digunakan oleh KNN untuk mengukur jarak antar data. Hasilnya menunjukkan bahwa parameter terbaik untuk model KNN adalah menggunakan K 3. Dan untuk model SVM dilakukan penalaan parameter untuk menemukan kombinasi parameter terbaik, hasilnya Parameter regularisasi sebesar 100, dan kernel nya default rbf, dan gamma nya bernilai 1. Selanjutnya, objek GridSearchCV digunakan untuk melakukan *Grid Search* dengan 10-fold *cross-validation*.

Tabel 11. Hasil optimasi menggunakan Grid Search CV

Model	Precision	Recall	F1-Score	Accuracy
KNN	0,91	0,98	0,94	0,95
SVM	0,90	0,99	0,94	0,94

Hasil dari pengujian pada Tabel 11 ini menunjukkan evaluasi kinerja dua model klasifikasi, yaitu *K-Nearest Neighbors* (KNN) dan *Support Vector Machine* (SVM), setelah mengalami proses optimasi menggunakan metode *Grid Search Cross Validation* (CV). Untuk model KNN, ditemukan bahwa presisi (*precision*) sebesar 0.91, yang mengindikasikan bahwa sekitar 91% dari prediksi positif yang dibuat oleh model benar-benar relevan. *Recall* sebesar 0.98 menunjukkan bahwa sekitar 98% dari total sampel positif berhasil diidentifikasi dengan

tepat oleh model. *F1-score*, yang merupakan harmonic mean dari presisi dan *recall*, memiliki nilai sebesar 0.94. Akurasi model KNN mencapai 0.95, yang menunjukkan seberapa baik model dapat mengklasifikasikan keseluruhan data dengan benar. Sementara itu, untuk model SVM, presisi sebesar 0.90 menunjukkan bahwa sekitar 90% dari prediksi positif yang dibuat oleh model benar-benar relevan. *Recall* sebesar 0.99 mengindikasikan bahwa sekitar 99% dari total sampel positif berhasil diidentifikasi dengan tepat oleh model. *F1-score*, yang merupakan harmonic mean dari presisi dan *recall*, juga memiliki nilai sebesar 0.94. Akurasi model SVM mencapai 0.94, yang menunjukkan seberapa baik model dapat mengklasifikasikan keseluruhan data dengan benar.

3.4. Hasil perbandingan model KNN dan SVM

Dari Gambar 4 yang telah disajikan, dapat disimpulkan bahwa terdapat 668 data yang telah diprediksi dengan benar untuk label 0, sementara terdapat 678 data yang diprediksi dengan benar untuk label 1. Meskipun belum di optimasi model KNN mampu memberikan akurasi sebesar 0.91.

	precision	recall	f1-score	support
0	0.99	0.83	0.90	668
1	0.86	0.99	0.92	678
accuracy			0.91	1346
macro avg	0.92	0.91	0.91	1346
weighted avg	0.92	0.91	0.91	1346

Gambar 4. Hasil prediksi model KNN Sebelum optimasi

Dari dalam Gambar 5, dapat disimpulkan bahwa ada 668 data yang berhasil diprediksi dengan tepat sebagai label 0, sementara 678 data diprediksi dengan benar sebagai label 1. Model KNN berhasil meningkatkan akurasi setelah optimasi. Akurasi keseluruhan model meningkat dari sebelumnya mencapai 0.95.

	precision	recall	f1-score	support
0	0.99	0.90	0.94	668
1	0.91	0.99	0.95	678
accuracy			0.95	1346
macro avg	0.95	0.95	0.95	1346
weighted avg	0.95	0.95	0.95	1346

Gambar 5. Hasil prediksi model KNN setelah optimasi

Dari Gambar 6 yang ditampilkan, dapat disimpulkan bahwa terdapat 668 data yang berhasil diprediksi dengan akurat sebagai label 0, sementara 678 data diprediksi dengan tepat sebagai label 1. Namun, model SVM hanya mencapai akurasi sebesar 0.83.

	precision	recall	f1-score	support
0	0.88	0.76	0.82	668
1	0.79	0.90	0.84	678
accuracy			0.83	1346
macro avg	0.84	0.83	0.83	1346
weighted avg	0.84	0.83	0.83	1346

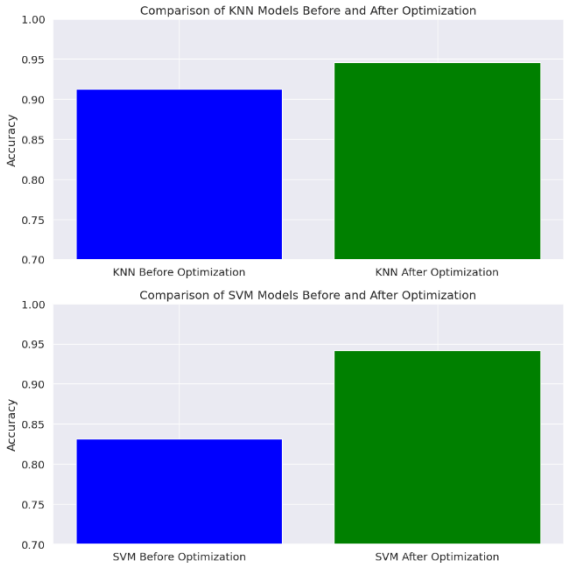
Gambar 6. Hasil prediksi model SVM sebelum optimasi

Dari hasil yang ditunjukkan dalam Gambar 7, dapat disimpulkan bahwa terdapat 668 data yang berhasil diprediksi dengan tepat sebagai label 0, dan 678 data diprediksi dengan benar sebagai label 1. Selain itu, model SVM berhasil meningkatkan akurasinya setelah proses optimasi. Akurasi keseluruhan model meningkat dari sebelumnya menjadi 0.94.

	precision	recall	f1-score	support
0	0.99	0.89	0.94	668
1	0.90	0.99	0.95	678
accuracy			0.94	1346
macro avg	0.95	0.94	0.94	1346
weighted avg	0.95	0.94	0.94	1346

Gambar 7. Hasil prediksi model SVM setelah optimasi

Perbandingan hasil optimasi sebelum dan setelah penerapan GridSearchCV bisa diamati pada Gambar 8. Grafik yang terdapat pada gambar tersebut menunjukkan hasil sebelum dan sesudah optimasi pada kedua model knn dan svm.



Gambar 8. Perbandingan hasil sebelum dan sesudah optimasi KNN dan SVM

Hasil pada pengujian optimasi model KNN dan SVM disajikan pada Gambar 4. Terlihat perbedaan hasil sebelum dan sesudah, hasil sebelum optimasi model KNN hanya memiliki accuracy sebesar 0.91 dan setelah dioptimasi accuracy meningkat 4% menjadi 0.95. Dan pada model SVM sebelum optimasi hanya memiliki accuracy sebesar 0.83 dan setelah dioptimasi accuracy meningkat 11% menjadi 0.94. Hal ini menunjukkan optimasi menggunakan grid search berhasil meningkatkan performa model. Setelah melihat perbandingan accuracy dari model sebelum dan sesudah optimasi pada Tabel 12 disajikan hasil perbandingan sebelum dan sesudah optimasi secara detail.

Perbandingan hasil optimasi antara *K-Nearest Neighbors* (KNN) dan *Support Vector Machine* (SVM) menunjukkan peningkatan kinerja yang

signifikan setelah penerapan *Grid Search Cross Validation* (GridSearchCV). Sebelum optimasi, model KNN memiliki presisi sebesar 0.85, recall sebesar 0.90, dan F1-Score sebesar 0.91, dengan akurasi sebesar 0.91.

Tabel 12. Perbandingan Hasil Optimasi

Model	Precision	Recall	F1-Score	Accuracy
KNN	0,85	0,90	0,91	0,91
KNN + GridSearchCV	0,91	0,98	0,94	0,95
SVM	0,78	0,89	0,84	0,83
SVM + GridSearchCV	0,90	0,99	0,94	0,94

Setelah melalui proses optimasi menggunakan GridSearchCV, terjadi peningkatan yang signifikan dalam presisi menjadi 0.91, recall menjadi 0.98, dan F1-score menjadi 0.94, dengan akurasi meningkat menjadi 0.95. Hal ini menunjukkan bahwa model KNN telah mampu memperbaiki kemampuannya dalam mengklasifikasikan data dengan tepat setelah optimasi, terutama dalam hal mengidentifikasi sampel positif.

Sementara itu, pada model SVM sebelum optimasi, presisi hanya sebesar 0.78, recall sebesar 0.89, dan F1-Score sebesar 0.84, dengan akurasi sebesar 0.83. Namun, setelah proses optimasi dengan GridSearchCV, terjadi peningkatan yang signifikan dalam presisi menjadi 0.90, recall menjadi 0.99, dan F1-score menjadi 0.94, dengan akurasi tetap sebesar 0.94. Hal ini menunjukkan bahwa model SVM telah mengalami peningkatan yang signifikan dalam kemampuannya untuk mengidentifikasi sampel positif dan mengklasifikasikan data dengan lebih akurat setelah melalui proses optimasi.

Kedua model, baik KNN maupun SVM, mengalami peningkatan kinerja setelah melalui proses optimasi menggunakan *Grid Search Cross Validation*. Namun, model KNN tampaknya menunjukkan peningkatan yang lebih besar dalam hal recall, sedangkan model SVM lebih meningkat dalam presisi.

4. DISKUSI

Pada Sebelumnya terdapat beberapa penelitian yang melakukan optimasi model untuk prediksi terhadap kesehatan. Salah satunya adalah penelitian yang optimasi algoritma KNN dengan gridsearchcv pada prediksi kanker paru-paru. Berdasarkan hasil yang telah dilakukan, hasil optimasi terbaik oleh KNN memperoleh nilai akurasi sebesar 96% [28]. Kemudian penelitian lain melakukan optimasi SVM dengan gridsearch untuk Prediksi Kualitas Udara. Hasil penelitian menunjukkan bahwa algoritma akurasi SVM mencapai 94% setelah dioptimasi [29].

Beberapa penelitian terdahulu telah menyoroti pentingnya optimasi parameter dalam meningkatkan kinerja model klasifikasi. Penggunaan teknik Grid Search CV secara signifikan meningkatkan akurasi dan stabilitas model SVM dan KNN. Selain itu, peran penting presisi dan *recall* dalam evaluasi kinerja model klasifikasi. Peran penting presisi dan *recall* adalah dua metrik evaluasi yang penting dalam kinerja model klasifikasi. Presisi mengukur akurasi prediksi positif, sementara *recall* mengukur kemampuan model untuk mengidentifikasi sebagian besar kasus positif yang sebenarnya. Presisi memastikan bahwa prediksi positif yang dibuat oleh model relevan dan akurat, sedangkan *recall* memastikan bahwa sebagian besar kasus positif yang sebenarnya dapat terdeteksi oleh model. Keduanya bekerja bersama-sama untuk memberikan gambaran yang komprehensif tentang kemampuan model dalam melakukan prediksi yang akurat dan relevan. Temuan ini konsisten dengan hasil penelitian ini, di mana penelitian ini terdapat peningkatan yang signifikan dalam presisi dan *recall* setelah melalui proses optimasi. Namun, perbaikan yang dicapai berbeda-beda di antara kedua model pada penelitian ini, dengan model KNN menunjukkan peningkatan yang lebih besar dalam *recall*, sementara model SVM lebih meningkat dalam presisi. Selain itu penelitian ini memiliki beberapa perbedaan dari penelitian sebelumnya yaitu dengan teknik preprocessing, pada data penelitian ini terdapat data yang tidak seimbang sehingga harus dilakukan SMOTE, jika tidak diberi imbalance data maka precision, recall, dan f1-score tidak menunjukkan hasil yang bagus.

5. KESIMPULAN

Hasil pengujian optimasi model KNN dan SVM menunjukkan peningkatan kinerja yang signifikan setelah penerapan *Grid Search Cross Validation* (GridSearchCV). Peningkatan tersebut tercermin dari perbedaan hasil sebelum dan setelah optimasi, di mana akurasi model KNN meningkat 4% dari 0.91 menjadi 0.95, sementara akurasi model SVM meningkat 11% dari 0.83 menjadi 0.94. Dalam mengoptimalkan kinerja model klasifikasi KNN dan SVM menggunakan *Grid Search Cross Validation*, kedua model menunjukkan peningkatan yang signifikan. Namun, perbaikan yang dicapai berbeda-beda di antara keduanya. Model KNN mengalami peningkatan yang lebih besar dalam *recall*, menunjukkan kemampuan yang lebih baik dalam mengidentifikasi sampel positif, sementara model SVM menunjukkan peningkatan yang lebih besar dalam presisi, menunjukkan kemampuan yang lebih baik dalam memastikan prediksi positif relevan. Pemilihan model yang optimal akan tergantung pada kebutuhan spesifik dan preferensi dalam menangani tipe data dan masalah klasifikasi yang dihadapi. Jika prioritas adalah untuk mengidentifikasi sebanyak mungkin sampel positif, maka model KNN mungkin menjadi pilihan yang lebih baik. Namun, jika penting untuk memastikan bahwa prediksi positif yang dibuat relevan, model SVM mungkin lebih sesuai.

DAFTAR PUSTAKA

[1] N. Permatasari, "Perbandingan Stroke Non Hemoragik dengan Gangguan Motorik Pasien Memiliki Faktor Resiko Diabetes Melitus dan Hipertensi," *Jurnal Ilmiah Kesehatan Sandi Husada*, vol. 9, no. 1, pp. 298-304, 2020, doi: 10.35816/jiskh.v11i1.273.

[2] A. Byna dan M. Basit, "Penerapan Metode Adaboost Untuk Mengoptimasi Prediksi Penyakit Stroke Dengan Algoritma Naïve Bayes," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 9, no. 3, pp. 407-411, 2020, doi: 10.32736/sisfokom.v9i3.1023.

[3] P. A. Setiawan, "Diagnosis dan Tatalaksana Stroke Hemoragik," *Jurnal Medika Hutama*, vol. 3, no. 1, pp. 1660-1665, 2021.

[4] K. Akmal, A. Faqih dan F. Dikananda, "Perbandingan Metode Algoritma Naive Bayes Dan K-Nearest Neighbors Untuk Klasifikasi Penyakit Stroke," *Jurnal Mahasiswa Teknik Informatika*, vol. 7, no. 1, pp. 470-477, 2023, doi: 10.36040/jati.v7i1.6367.

[5] Hozairi, Anwari dan S. Alim, "IMPLEMENTASI ORANGE DATA MINING UNTUK KLASIFIKASI KELULUSAN MAHASISWA DENGAN MODEL K-NEARESTNEIGHBOR, DECISION TREE SERTA NAIVE BAYES," *Jurnal Ilmiah NERO*, vol. 6, no. 2, pp. 133-144, 2021, doi: 10.21107/nero.v6i2.237.

[6] A. O. C. Pratiwi, "Klasifikasi Jenis Anggur Berdasarkan Bentuk Daun Menggunakan Convolutional Neural Network Dan K-Nearest Neighbor.," *Jurnal Ilmiah Teknik Informatika Dan Komunikasi*, vol. 3, no. 2, pp. 201-224, 2023, doi: 10.55606/juitik.v3i2.535.

[7] R. P. Kurniadi, R. Saedudin dan V. P. Widartha, "PERBANDINGAN AKURASI ALGORITMA K-NEAREST NEIGHBOR DAN LOGISTIC REGRESSION UNTUK KLASIFIKASI PENYAKIT DIABETES," Universitas Telkom, S1 Sistem Informasi, Bandung, 2021.

[8] S. Chowdhury dan M. P. Schoen, "Research Paper Classification using Supervised Machine Learning Techniques," *2020 Intermountain Engineering, Technology and Computing (IETC)*, pp. 1-6, 2020, doi: 10.32628/IJSRCSEIT.

[9] H. S. Wafa, A. I. Hadiana dan F. R. Umbara, "Prediksi Penyakit Diabetes Menggunakan Algoritma Support Vector Machine (SVM)," *INFORMATICS AND DIGITAL EXPERT (INDEX)*, vol. 4, no. 1, pp. 40-45, 2022.

[10] S. Dwiasnati dan Y. Devianto, "Optimasi Prediksi Bencana Banjir menggunakan Algoritma SVM untuk penentuan Daerah Rawan Bencana Banjir," *Prosiding SISFOTEK*, vol. 5, no. 1, pp. 202-207, 2021.

[11] G. N. AHMAD, H. FATIMA, SHAFIULLAH, A. S. SAIDI dan IMDADULLAH, "Efficient Medical Diagnosis of Human Heart Diseases Using Machine Learning Techniques With and Without GridSearchCV," *IEEE*, vol. 20, pp. 80151-80173, 2022, doi: 10.1109/ACCESS.2022.3165792.

[12] R. Alshammri, G. Alharbi, E. Alharbi dan I. Almubark, "Machine learning approaches to identify Parkinson's disease using voice signal features," *Frontiers*, vol. 6, 2023, doi: 10.3389/frai.2023.1084001.

[13] N. Chandrasekhar dan S. Peddakrishna, "Enhancing Heart Disease Prediction Accuracy through Machine Learning Techniques and Optimization," *Processes*, vol. 11, no. 4, p. 1210, 2023, doi: 10.3390/pr11041210.

[14] Sulistiana dan M. A. Muslim, "Support Vector Machine (SVM) Optimization Using Grid Search and Unigram to Improve E-Commerce Review Accuracy," *Jornal Of Soft Computing*, vol. 1, no. 1, pp. 8-15, 2020, doi: 10.52465/joscex.v1i1.3.

[15] T. A. Assegie, "An optimized K-Nearest Neighbor based breast cancer detection," *Journal of Robotics and Control*, vol. 2, no. 3, pp. 115-118, 2021.

[16] T. S. R. P. S. G. N. K. K. Tsehay Admassu Assegie, "Early Prediction of Gestational Diabetes with Parameter-Tuned K-Nearest Neighbor Classifier," *Jurnal of Robotics and Control*, vol. 4, no. 4, pp. 452-457, 2023.

[17] S. A. S. E. M. S. L. R. M. M. I. R. A. C. a. N. B. Alexey N. Beskopylny, "Concrete Strength Prediction Using Machine Learning Methods CatBoost, k-Nearest Neighbors, Support Vector Regression," *applied sciences*, vol. 12, no. 21, p. 10864, 2022, doi: 10.3390/app122110864.

[18] FEDESORIANO, "Stroke Prediction Dataset," Kaggle, 2021. Tersedia pada: < https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset > [Diakses 20 Januari 2024].

[19] A. K. M. &. A. A. K. Saleh H. Alhathloul, "Low visibility event prediction using random forest and K-nearest neighbor methods," *Theor Appl Climatol*, vol. 155, pp. 1289-1300, 2023, doi: 10.1007/s00704-023-04697-6.

[20] M. S. Irwanto, F. A. Bachtiar dan N.

Yudistira, "KLASIFIKASI AKTIVITAS MANUSIA MENGGUNAKAN ALGORITME COMPUTED INPUT WEIGHT EXTREME LEARNING MACHINE DENGAN REDUKSI DIMENSI PRINCIPAL COMPONENT ANALYSIS," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 9, no. 6, pp. 1195-1202, 2022, doi: 10.25126/jtiik.2022965504.

[21] R. N. Ikhsani dan F. F. Abdulloh, "Optimasi SVM dan Decision Tree Menggunakan SMOTE Untuk Mengklasifikasi Sentimen Masyarakat Mengenai Pinjaman Online," *Jurnal Media Informatika Budidarma*, vol. 7, no. 4, pp. 1667-1677, 2023, doi: 10.30865/mib.v7i3.6368.

[22] Rahmadini, E. E. LorencisLubis, A. Priansyah, Y. R.W.N dan T. Meutia, "PENERAPAN DATA MINING UNTUK MEMPREDIKSI HARGA BAHAN PANGAN DI INDONESIA MENGGUNAKAN ALGORITMA K-NEAREST NEIGHBOR," *Jurnal Mahasiswa Akuntansi Samudra*, vol. 4, no. 4, pp. 223 -235, 2023.

[23] A. S. Abiyyu dan K. M. Lhaksmana, "Perbandingan Metode Seleksi Fitur untuk Mengoptimasi Model Support Vector Machine dalam Memprediksi Turnover Pegawai," *e-Proceeding of Engineering*, vol. 10, no. 2, p. 1921, 2023.

[24] C. Chazar dan Widhiaputra, "Machine Learning Diagnosis Kanker Payudara Menggunakan Algoritma Support Vector Machine," *INFORMASI (Jurnal Informatika dan Sistem Informasi)*, vol. 12, no. 1, pp. 67-80, 2020, doi: 10.37424/informasi.v12i1.48.

[25] Y. N. Fuadah, M. A. Pramudito dan K. M. Lim, "An Optimal Approach for Heart Sound Classification Using Grid Search in Hyperparameter Optimization of Machine Learning," *bioengineering*, vol. 10, no. 1, p. 45, 2023, doi: 10.3390/bioengineering10010045.

[26] C. Bigoni, A. Cadic-Melchior, T. Morishita dan F. C. Hummel, "Optimization of phase prediction for brain-state dependent stimulation: a grid-search approach," *Journal of Neural Engineering*, vol. 20, no. 1, 2023, doi: 10.1088/1741-2552/acb1d8.

[27] M. Azhari, Z. Situmorang dan R. Rosnelly, "Perbandingan Akurasi, Recall, dan Presisi Klasifikasi pada Algoritma C4.5, Random Forest, SVM dan Naive Bayes," *Jurnal Media Informatika Budi Dharma*, vol. 5, no. 2, pp. 640-651, 2021, doi: 10.30865/mib.v5i2.2937.

[28] S. T. Kusuma dan T. B. Sasongko, "Optimasi K-Nearest Neighbor dengan Grid Search CV pada Prediksi Kanker Paru-Paru," *Indonesian Journal of Computer Science*, vol. 12, no. 4, pp. 2162-2171, 2023, doi: 10.33022/ijcs.v12i4.3267.

[29] P. Elisa dan A. R. Isnain, "COMPARISON OF RANDOM FOREST, SUPPORT VECTOR MACHINE AND NAIVE BAYES ALGORITHMS TO ANALYZE SENTIMENT TOWARDS MENTAL HEALTH STIGMA," *Jurnal Teknik Informatika (JUTIF)*, vol. 5, no. 1, pp. 321-329, 2024, doi: 10.52436/1.jutif.2024.5.1.1817.