KOMPARASI DETEKSI PENYAKIT GINJAL KRONIS MENGGUNAKAN ALGORITMA *SUPPORT VECTOR MACHINE* DAN *RANDOM FOREST*

Khasan Galuh Ramadhan[1], Gentur Wahyu Nyipto Wibowo[2], Nadia Annisa Maori[3]

[1,2,3] Program Studi Teknik Informatika, Universitas Islam Nahdlatul Ulama Jepara
Jl. Taman Siswa, Pekeng, Kauman, Tahunan, Kec. Tahunan, Kabupaten Jepara, Jawa Tengah 59451
[1] kgaluhr@gmail.com, [2] gentur@unisnu.ac.id, [3] nadia@unisnu.ac.id

Abstract

Chronic Kidney Disease (CKD) is a significant global health issue that often goes undetected until it reaches an advanced stage. CKD is also one of the leading causes of death among Non-Communicable Diseases (NCDs) globally.. Therefore, early detection of CKD is important to prevent the risk of complications.This study compares two classification algorithms, namely Support Vector Machine (SVM) and Random Forest (RF). The SVM algorithm is known for its high level of accuracy, efficient in memory usage, and capable of handling data with non-normal distributions. while the RF algorithm is known for its ability to overcome overfitting and produce more accurate predictions. The data used are CKD patient data obtained from the UCI Machine Learning Repository, consisting of 400 patient records with 25 features. Parameter optimization was carried out using GridSearchCV to obtain the best parameters for both algorithms.The Random Forest (RF) algorithm showed better performance than SVM, with an accuracy of 98.75%, a precision of 98.48%, a recall of 98.96%, and an F1-score of 98.70%.Random Forest (RF) algorithm is more effective in PGK classification on this dataset than SVM. This study emphasizes the importance of selecting an appropriate algorithm and optimizing parameters for developing a reliable and accurate classification model.

***Keywords** : Chronic Kidney Disease, Classification, Support Vector Machine, Random Forest, GridSearchCV*

Abstrak

Penyakit Ginjal Kronis (PGK) adalah masalah kesehatan global yang signifikan dan seringkali tidak terdeteksi hingga mencapai stadium lanjut. PGK juga menjadi salah satu Penyakit Tidak Menular (PTM) penyebab kematian terbanyak dalam lingkup global. Oleh karena itu, deteksi dini PGK sangat penting untuk mencegah risiko komplikasi. Studi ini membandingkan dua algoritma klasifikasi, yaitu *Support Vector Machine* (SVM) dan *Random Forest* (RF). Algoritma SVM dikenal karena tingkat akurasinya yang tinggi, efisien dalam penggunaan memori, dan kemampuannya untuk menangani data dengan distribusi yang tidak normal. Sementara itu, algoritma RF dikenal karena kemampuannya dalam mengatasi overfitting dan menghasilkan prediksi yang lebih akurat. Data yang digunakan dalam penelitian ini adalah data pasien PGK yang diperoleh dari UCI Machine Learning Repository, yang terdiri dari 400 catatan pasien dengan 25 fitur. Optimalisasi parameter dilakukan menggunakan *GridSearchCV* untuk mendapatkan parameter terbaik untuk kedua algoritma. Hasil penelitian ini, algoritma *Random Forest* (RF) menunjukkan performa yang lebih baik daripada algoritma SVM, dengan akurasi sebesar 98,75%, *precision* 98,48%, *recall* 98,96%, dan *F1-score* 98,70%. Algoritma *Random Forest* (RF) lebih efektif dalam klasifikasi PGK pada dataset ini dibandingkan SVM. Studi ini menekankan pentingnya pemilihan algoritma yang tepat dan optimasi parameter dalam mengembangkan model klasifikasi yang andal dan akurat.

Kata kunci : Penyakit Ginjal Kronis, Klasifikasi, *Support Vector Machine*, *Random Forest*, *GridSearchCV*

1. PENDAHULUAN

Penyakit Ginjal Kronis (PGK) adalah kondisi yang ditandai dengan kerusakan progresif pada ginjal, yang menyebabkan penurunan kapasitas fungsionalnya. Secara khusus, PGK didefinisikan sebagai penurunan berkelanjutan laju filtrasi glomerulus (GFR) hingga kurang dari 60 ml/menit/ 1,73 m^2, dengan penurunan fungsi ini biasanya berkembang selama sekitar tiga bulan

[1]. PGK merupakan gangguan pada ginjal yang dapat disebabkan oleh berbagai faktor. Penyakit ini sering kali muncul tanpa gejala yang khas di tahap awal, sehingga sering kali tidak terdeteksi sejak dini dan menyebabkan keterlambatan dalam penanganan pasien dengan gagal ginjal. Penting untuk mengenali tanda-tanda awal gagal ginjal guna memungkinkan penanganan lebih cepat atau setidaknya memperlambat perkembangan penyakit. Indikator awal penyakit ginjal kronis (PGK) mungkin termasuk sering buang air kecil, terutama buang air kecil di malam hari, dan adanya urin berbusa atau bercampur darah, mudah lelah, nyeri pada punggung bagian bawah, pembengkakan pada tubuh, mual, muntah, dan bau mulut yang tidak sedap [2].

Pada tahun 2018, data Riset Kesehatan Dasar (Riskesdas) menunjukkan bahwa "prevalensi penyakit ginjal kronis di Indonesia adalah 3,8%, meningkat 1,8 poin persentase dari tahun 2013, yang mana setara dengan sekitar 713.783 orang yang terkena penyakit ini". Maluku Utara menunjukkan tingkat prevalensi tertinggi sebesar 5,6%, sedangkan Kalimantan Utara memiliki tingkat prevalensi terendah sebesar 0,64%. Sulawesi Tengah melaporkan tingkat prevalensi sebesar 0,53%, dan daerah-daerah seperti Nusa Tenggara Barat, Jawa Barat, Jawa Tengah, Yogyakarta, dan Bali masing-masing mencatat tingkat prevalensi sebesar 0,48%. Jambi, Sulawesi Tenggara, Banten, dan Bangka Belitung memiliki tingkat prevalensi sebesar 0,32%, dan Sumatera Barat memiliki tingkat prevalensi sebesar 0,40%, dengan total 13.834 orang yang terkena di provinsi ini [3].

Meskipun pentingnya deteksi dini PGK diakui secara luas, tantangan utama adalah bagaimana meningkatkan akurasi dan efektivitas deteksi risiko penyakit ini. Banyak metode yang telah dikembangkan untuk deteksi dini, namun masih terdapat kesenjangan dalam hal akurasi dan kemampuan untuk menangani data yang memiliki *missing values* dan variabel *non-numerik*. Metode konvensional sering kali tidak mampu memberikan hasil yang memadai, sehingga diperlukan pendekatan yang lebih canggih.

Penelitian ini mengeksplorasi penggunaan algoritma *machine learning*, yaitu *Support Vector Machine* (SVM) dan *Random Forest* (RF), untuk mendeteksi PGK secara dini. Kedua algoritma ini dikenal mampu melakukan klasifikasi data dengan baik dan dapat mengatasi berbagai jenis data, termasuk data medis yang kompleks. Studi ini bertujuan untuk mengevaluasi dan membandingkan kemanjuran dua algoritma dalam mendeteksi PGK pada tahap awal, dengan tujuan mengidentifikasi metode yang lebih efektif dan akurat untuk deteksi dini. Diharapkan bahwa penerapan teknologi *machine learning* ini dapat meningkatkan akurasi dan efisiensi dalam mengidentifikasi faktor risiko PGK.

Berbagai penelitian telah dilakukan untuk mempelajari penyakit ginjal kronis (PGK), di mana para peneliti menggunakan berbagai pendekatan dan metode untuk mengatasi tantangan dalam mendiagnosis dan memprediksi PGK secara akurat. Dalam penelitian berjudul "Pendekatan Algoritma *Neural Network* dan *Genetic Algorithm* Untuk Prediksi Penyakit Ginjal Kronis" (2024), Hendy D. Siswaja dan Yudi Ramdhani memanfaatkan algoritma *Neural Network* yang dioptimalkan menggunakan *Genetic Algorithm* (GA) serta *k-fold Cross Validation* dengan nilai k sebagai kelipatan 10 untuk memprediksi apakah seorang pasien menderita PGK berdasarkan dataset hasil uji klinis pasien tersebut. Hasil penelitian ini menunjukkan bahwa "algoritma *Neural Network* yang disempurnakan dengan penerapan Algoritma Genetika (GA) mencapai tingkat akurasi sebesar 98,75%". Oleh karena itu, algoritma ini memiliki potensi untuk dikembangkan lebih lanjut sebagai aplikasi atau komponen dalam sistem kesehatan guna meningkatkan akurasi diagnosis PGK pada pasien [4].

Sebuah penelitian penting menyelidiki penerapan algoritma *Support Vector Machine* (SVM) dan pengembangan algoritma Random Forest, sebagaimana dilakukan oleh Mahendra Dwifebri Purbolaksono, Muhammad Irvan Tantowi, Adnan Imam Hidayat, dan Adiwijaya. Penelitian ini memberikan kontribusi terhadap pemahaman dan kemajuan teknik pembelajaran mesin. Penelitian ini berjudul "*Perbandingan Support Vector Machine dan Modified Balanced Random Forest dalam Deteksi Pasien Penyakit Diabetes*" (2021) dan berfokus pada deteksi dini diabetes melalui klasifikasi data pasien menggunakan pendekatan *machine learning*. Studi ini membandingkan SVM dengan *Modified Balanced Random Forest* (MBRF) dan menemukan bahwa MBRF mencapai kinerja terbaik dengan akurasi sebesar 97,8%, sedangkan SVM mencapai akurasi 87,94%. Temuan ini memberikan kontribusi penting dalam pengembangan metode klasifikasi yang lebih akurat untuk deteksi dini diabetes serta menekankan pentingnya pemilihan algoritma yang tepat dalam aplikasi kesehatan [5].

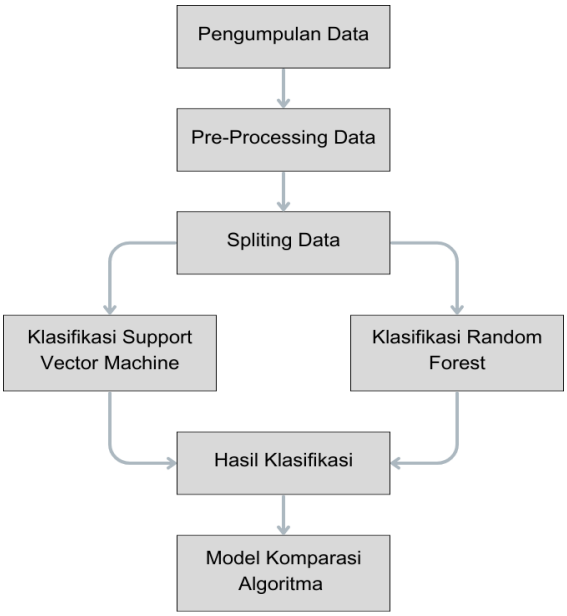
Meskipun berbagai studi telah mengaplikasikan algoritma *machine learning* untuk deteksi PGK, terdapat beberapa masalah dan celah penelitian yang masih belum teratasi. Seperti optimisasi parameter dan validasi model sering kali tidak dilakukan secara menyeluruh,

yang dapat mengurangi akurasi dan generalisasi model. Urgensi penelitian ini terletak pada kebutuhan untuk mengembangkan model prediktif yang lebih andal dan valid untuk deteksi dini PGK guna meningkatkan intervensi klinis dan manajemen pasien.

Penelitian ini mengkaji kinerja algoritma *Support Vector Machine* (SVM) dan *Random Forest* (RF) dalam klasifikasi data kesehatan untuk mendeteksi indikator awal Penyakit Ginjal Kronis (PGK). Evaluasi kedua algoritma tersebut akan dilakukan dengan menggunakan metrik seperti akurasi, *recall*, *precision*, dan *F1-score* untuk menilai efektivitasnya. Selain itu, penelitian ini juga mencakup optimisasi parameter secara mendalam dengan menggunakan *GridSearchCV* (*Grid Search Cross-Validation*) dan validasi model melalui *cross-validation* untuk memastikan model yang dihasilkan memiliki performa terbaik. Studi ini berfokus pada mengidentifikasi algoritma yang paling efektif untuk mendeteksi risiko PGK dan memberikan rekomendasi penerapan algoritma yang lebih efektif dalam program kesehatan untuk pencegahan dan penanganan PGK.

2. METODOLOGI PENELITIAN

Bagian ini menguraikan prosedur metodologis yang digunakan dalam penelitian, seperti yang diilustrasikan pada Gambar 1.



Gambar 1. Tahapan Penelitian

2.1. Pengumpulan Data

Dalam penelitian ini, data diperoleh dari *UCI Machine Learning Repository*, sebuah sumber data public yang terkenal. Dataset yang digunakan terdiri dari 400 sampel data pasien penderita penyakit ginjal kronis. Data ini menyediakan informasi penting yang memungkinkan analisis lebih lanjut menggunakan algoritma klasifikasi untuk mendeteksi penyakit ginjal kronis.

2.2. *Pre-processing* Data

Pre-processing data melibatkan pembersihan dan pengorganisasian data secara sistematis untuk mempersiapkannya bagi analisis selanjutnya. Proses ini memerlukan penghapusan *noise* yang tidak relevan, memastikan bahwa data disajikan secara terstruktur dan teratur, serta menangani ketidakkonsistenan atau gangguan apa pun dalam kumpulan data [6]. *Pre-processing* terdiri dari dua tahap utama: pembersihan data dan transformasi data. Tahap pembersihan data mencakup penghapusan *noise*, penghilangan data duplikat, pemeriksaan terhadap ketidakkonsistenan, serta perbaikan kesalahan dalam data. Transformasi data merupakan tahap akhir dalam *pre-processing* data. Tahap ini melibatkan penyesuaian dan modifikasi kumpulan data agar sesuai dengan persyaratan dan spesifikasi algoritma yang digunakan [7].

2.3. *Spliting Data*

Pada langkah berikutnya, penting untuk membagi kumpulan data menjadi dua subset yang berbeda: data pelatihan dan data pengujian. Proses ini, yang dikenal sebagai *splitting data*, dapat dijalankan secara efektif menggunakan pustaka "*scikit-learn*" dalam Python. Data pelatihan digunakan untuk membangun dan menyempurnakan model, sedangkan data pengujian, yang tetap tidak tersentuh selama fase pelatihan, digunakan untuk mengevaluasi akurasi model [8].

2.4. *Support Vector Machine* (SVM)

Support Vector Machine (SVM) pertama kali diperkenalkan oleh Vapnik pada tahun 1990an [9]. *Support Vector Machine* (SVM) adalah algoritma pembelajaran mesin yang tangguh yang digunakan untuk tugas klasifikasi dan regresi. Keunggulan algoritma SVM sendiri adalah memiliki tingkat akurasi yang tinggi, efisien dalam penggunaan memori, dan mampu menangani data yang distribusinya tidak normal [10]. Pada dasarnya, algoritma SVM bertujuan untuk mengidentifikasi *hyperplane* yang optimal batas keputusan yang secara maksimal memisahkan data ke dalam dua kategori yang berbeda.

Hyperplane yang optimal ini dicirikan oleh margin terbesar, yang merupakan jarak antara titik data terdekat dari setiap kategori, yang dikenal sebagai *support vector*. Efektivitas SVM sebagian besar dipengaruhi oleh fungsi kernel, yang mengubah data ke dalam ruang berdimensi lebih tinggi untuk memfasilitasi pemisahan [11]. Tujuannya adalah untuk menentukan *hyperplane* yang memaksimalkan margin ini. Untuk representasi matematis terperinci dari SVM, lihat Persamaan 1 yang disediakan di bawah ini [6]:

$$f(x) = w^{t}\Phi(x) + b \quad (1)$$

Keterangan:

b = Bias

$x = (x_1, x_2, \ldots, x_n)$ = Variabel Input

2.5. *Random Forest* (RF)

Metode *Random Forest* yang pertama kali diperkenalkan oleh Ho pada tahun 1995, merupakan kemajuan signifikan dalam domain pembelajaran mesin, khususnya dalam mengatasi tantangan pengenalan pola [12]. Teknik ini dibangun berdasarkan metodologi *Classification and Regression Tree* (CART) yang mendasar. *Random Forest* menggunakan strategi yang dikenal sebagai agregasi bootstrap (atau bagging) yang dipadukan dengan pemilihan fitur acak. Metode ini menghasilkan kumpulan pohon keputusan, di mana setiap pohon dibangun dengan memilih fitur secara acak dan membagi simpul. Prediksi akhir diperoleh dengan menggabungkan keluaran dari masing-masing pohon ini, dengan suara mayoritas menentukan hasil klasifikasi atau regresi akhir [13]. Dengan menggabungkan sejumlah pohon keputusan, algoritma RF dapat menangani overfitting dan menghasilkan prediksi yang lebih akurat [14]. Dalam model RF, node dikategorikan ke dalam tiga jenis berbeda: root node, internal node, dan leaf node. Root node berfungsi sebagai titik awal pohon keputusan, yang dianalogikan sebagai batang pohon. Internal node, yang diposisikan di sepanjang cabang, berfungsi untuk membagi data dan memiliki setidaknya satu masukan dan minimal dua keluaran. Sebaliknya, leaf node, yang terletak di ujung terminal cabang, menerima satu masukan dan tidak menghasilkan keluaran [15].

2.6. Model Komparasi Algoritma *Support Vector Machine* dan *Random Forest*

Pada tahap ini, dilakukan analisis komparatif terhadap dua model algoritma: *Support Vector Machine* (SVM) dan *Random Forest* (RF), untuk mengevaluasi kemanjurannya dalam mendeteksi penyakit ginjal kronis. Kinerja masing-masing model dinilai berdasarkan metrik evaluasi, termasuk akurasi, *precicison*, *recall*, dan *F1-score*. Metrik ini berfungsi sebagai indikator kecakapan model dalam menghasilkan hasil yang akurat dan andal.

3. HASIL DAN PEMBAHASAN

3.1. Pengumpulan Data dan Analisis Data

Pada penelitian ini, data diambil dari laman "https://archive.ics.uci.edu/dataset/336/chronic+kidney+disease". Data tersebut dimuat menggunakan pustaka Pandas, yang mempermudah proses pembacaan file CSV. Selanjutnya, data dianalisis dan hasil dari analisis ini disajikan pada Tabel I berikut.

TABEL I. HASIL ANALISIS DATA

No	Atribut	Dekskripsi	Kelas	Jenis	Missing Values	Nilai
1.	age	*Age*	Fitur	Numerik	9	age in years
2.	bp	*Blood Pressure*	Fitur	Numerik	12	in mm/Hg
3.	sg	*Specific Gravity*	Fitur	Kategorik	47	(1.005, 1.010, 1.015, 1.020, 1.025)
4.	al	*Albumin*	Fitur	Kategorik	46	(0, 1, 2, 3, 4, 5)
5.	su	*Sugar*	Fitur	Kategorik	49	(0, 1, 2, 3, 4, 5)
6.	rbc	*Red Blood Cells*	Fitur	Kategorik	152	(normal, abnormal)
7.	pc	*Pus Cell*	Fitur	Kategorik	65	(normal, abnormal)
8.	pcc	*Pus Cell Clumps*	Fitur	Kategorik	4	(present, notpresent)
9.	ba	*Bacteria*	Fitur	Kategoris	4	(present, notpresent)
10.	bgr	*Blood Glucose Random*	Fitur	Numerik	44	in mgs/dl
11.	bu	*Blood Urea*	Fitur	Numerik	19	in mgs/dl
12.	sc	*Serum Creatinine*	Fitur	Numerik	17	in mgs/dl

13.	sod	*Sodium*	Fitur	Numerik	87	in mEq/L
14.	pot	*Potassium*	Fitur	Numerik	88	in mEq/L
15.	hemo	*Hemoglobin*	Fitur	Numerik	52	in gms
16.	pcv	*Packed Cell Volume*	Fitur	Numerik	70	cells / cumm
17.	wc	*White Blood Cell Count*	Fitur	Numerik	105	cells / cumm
18.	rc	*Red Blood Cell Count*	Fitur	Numerik	130	in millions/cmm
19.	htn	*Hypertension*	Fitur	Kategorik	2	(yes, no)
20.	dm	*Diabetes Mellitus*	Fitur	Kategorik	2	(yes, no)
21.	cad	*Coronary Artery Disease*	Fitur	Kategorik	2	(yes, no)
22.	appet	*Appetite*	Fitur	Kategorik	1	(good, poor)
23.	pe	*Pedal Edema*	Fitur	Kategorik	1	(yes, no)
24.	ane	*Anemia*	Fitur	Kategorik	1	(yes, no)
25.	class	*Classification*	Target	Kategorik	0	(ckd, notckd)

3.2. *Pre-processing* Data

Pada tahap *pre-processing* data, terdapat beberapa proses yang dilakukan. Proses pertama adalah mengubah tipe data pada kolom numerik menjadi tipe data float, bertujuan untuk memudahkan operasi matematis dan statistik pada kolom-kolom tersebut.

Proses berikutnya adalah menangani *missing value* atau nilai yang hilang. Pada kolom numerik digunakan *Iterative Imputer*, metode ini melakukan iterasi melalui kolom-kolom yang hilang dan menggunakan regresi untuk memperkirakan nilai yang hilang, memberikan estimasi yang lebih akurat. Dan untuk kolom kategorik digunakan *Simple Imputer,* untuk mengisi nilai yang hilang dengan nilai yang paling sering muncul di setiap kolom kategorik.

Proses berikutnya pada tahapan *pre-processing* adalah transformasi data yang bertujuan untuk mengubah data kategorik menjadi nilai numerik. Pada kolom kategorik yang berisi kelas target, nilai atribut "class" diubah menjadi "0" dan "1".

3.3. *Spliting* Data

Sebelum menerapkan algoritma klasifikasi, kumpulan data dibagi menjadi dua subset: 80% untuk pelatihan dan 20% untuk pengujian. Pendekatan ini memastikan bahwa model dievaluasi berdasarkan data yang belum pernah ditemui sebelumnya, sehingga memberikan penilaian yang akurat terhadap kinerjanya.

3.4. *Support Vector Machine* (SVM)

Saat menggunakan algoritma *Support Vector Machine* (SVM) untuk tugas klasifikasi, beberapa langkah prosedural harus dipatuhi. Berikut ini memberikan gambaran terperinci tentang tahapan yang terlibat dalam proses klasifikasi SVM:

1. Menentukan Kernel dan Nilai-nilai Parameter

Dalam klasifikasi dengan algoritma *Support Vector Machine* (SVM), digunakan kernel linear, dan parameter grid ditetapkan untuk *C* (cost) dan γ (gamma), yang merupakan *hyperparameter* dalam algoritma SVM, seperti yang ditampilkan pada Tabel II berikut:

TABEL II. PARAMETER *SUPPORT VECTOR MACHINE*

Parameter	Nilai Parameter
C (cost)	(0.1, 1, 10, 100)
γ (gamma)	(1, 0.1, 0.01, 0.001)

2. Training Model dan Optimasi Hyperparameter

Model SVM diinisialisasi, dan *GridSearchCV (Grid Search Cross-Validation)* digunakan untuk mencari *hyperparameter* terbaik berdasarkan parameter grid yang telah didefinisikan. Pencarian ini menghasilkan model terbaik dengan parameter *C* (*cost*) bernilai 0.1 dan γ (*gamma*) bernilai 0.1.

3. Metrik Evaluasi Algoritma SVM

Untuk mengevaluasi kinerja proses klasifikasi pada data uji, digunakan matriks confusi. Matriks ini merupakan representasi tabular yang menggambarkan keakuratan klasifikasi dengan merinci jumlah contoh yang diidentifikasi dengan benar versus contoh yang salah diklasifikasikan [16]. Untuk penilaian komprehensif metrik kinerja algoritma SVM,

silakan lihat Tabel III yang disediakan di bawah ini:

TABEL III. METRIK EVALUASI ALGORITMA SVM

	Predicted Positive Class	***Predicted Negative Class***
Actual Positive Class	46	2
Actual Negative Class	0	32

4. Hasil Pengujian Algoritma SVM

Berdasarkan kernel, optimasi parameter, dan metrik evaluasi yang telah ditentukan, performa klasifikasi algoritma SVM yang telah dioptimalkan dapat dihitung. Hasil pengujian ini ditampilkan pada Gambar 2 berikut:

```
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.96      0.98        48
           1       0.94      1.00      0.97        32

    accuracy                           0.97        80
   macro avg       0.97      0.98      0.97        80
weighted avg       0.98      0.97      0.98        80

Akurasi SVM : 97.50%
Precision SVM : 97.06%
Recall SVM : 97.92%
F1 Score SVM : 97.42%
```

Gambar 2. Hasil Pengujian Algoritma SVM

3.5. *Random Forest* (RF)

Dalam klasifikasi menggunakan algoritma *Random Forest* (RF), juga dilakukan beberapa tahapan. Berikut adalah tahapan dalam mengklasifikasikan menggunakan algoritma RF:

1. Menentukan Parameter

Dalam Klasifikasi Algoritma *Random Forest* (RF) menentukan parameter grid yang didefinisikan dengan "*n_estimators*, *max_depth*, *min_samples_split*, dan *min_samples_leaf*", yang merupakan *hyperparameter* dalam algoritma RF. sebagaimana ditunjukkan pada Tabel IV berikut:

TABEL IV. PARAMETER *RANDOM FOREST* (RF)

Parameter	**Nilai Parameter**
n_estimators	(50, 100, 200)
max_depth	(none, 5, 10)
min_samples_split	(2, 5, 10)
min_samples_leaf	(1, 2, 4)

2. Training Model dan Optimasi Hyperparameter

Model *Random Forest* (RF) diinisialisasi, dan *GridSearchCV* digunakan untuk menemukan *hyperparameter* terbaik berdasarkan parameter grid yang telah ditentukan. Proses ini menghasilkan model terbaik dengan nilai parameter *n_estimators* = 50, *max_depth* = none, *min_samples_split* = 2, dan *min_samples_leaf* = 1.

3. Metrik Evaluasi Algoritma SVM

Metrik evaluasi untuk algoritma RF ditunjukkan pada Tabel V berikut:

TABEL V. METRIK EVALUASI ALGORITMA RF

	Predicted Positive Class	***Predicted Negative Class***
Actual Positive Class	47	1
Actual Negative Class	0	32

4. Hasil Pengujian Algoritma SVM

Berdasarkan optimasi parameter, dan metrik evaluasi yang telah ditetapkan, performa klasifikasi algoritma RF yang telah dioptimalkan dapat dihitung, dengan hasil pengujian yang ditampilkan pada Gambar 3 berikut.

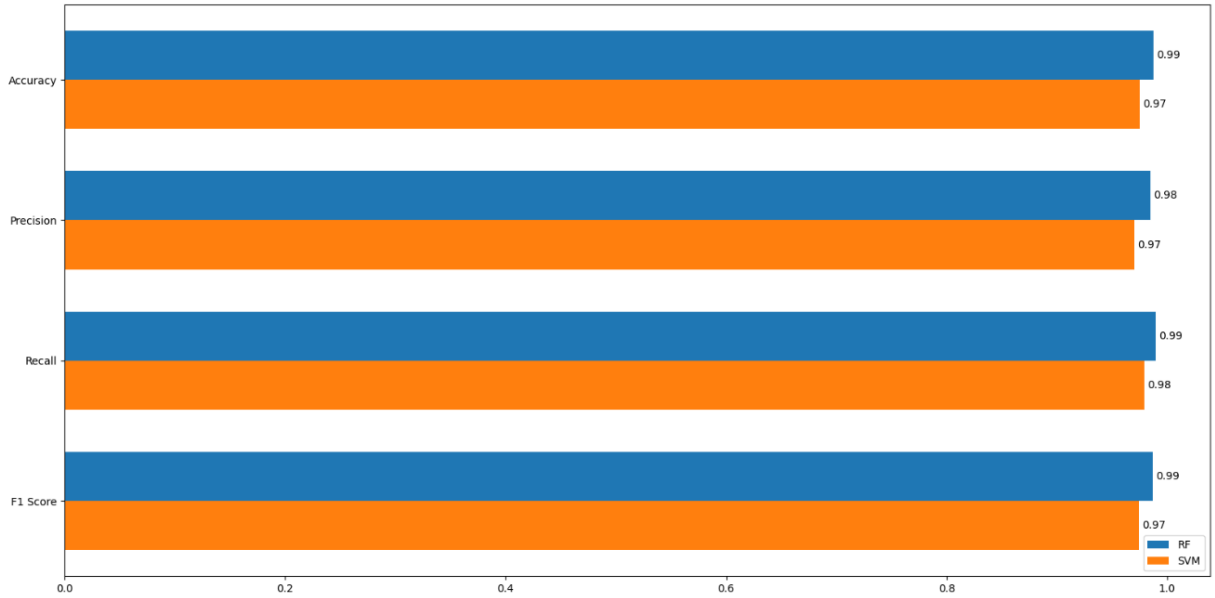
```
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.98      0.99        48
           1       0.97      1.00      0.98        32

    accuracy                           0.99        80
   macro avg       0.98      0.99      0.99        80
weighted avg       0.99      0.99      0.99        80

Akurasi RF : 98.75%
Precision RF : 98.48%
Recall RF : 98.96%
F1 Score RF : 98.70%
```

Gambar 3. Hasil Pengujian Algoritma RF

3.6. Model Komparasi Algoritma *Support Vector Machine* dan *Random Forest*

Dalam studi ini, dilakukan komparasi hasil dari pengujian dua algoritma *machine learning* yang digunakan, yaitu *Support Vector Machine* (SVM) dan *Random Forest* (RF). Penilaian peforma kedua algoritma didasarkan pada metrik evaluasi yaitu akurasi, *precision, recall*, dan *F1-score*, dengan menggunakan nilai parameter terbaik. Parameter yang digunakan dalam pengujian algoritma SVM yaitu *C* (*cost*) = 0.1 dan γ (*gamma*) = 0.1, sedangkan nilai parameter yang digunalkan

pada algoritma RF yaitu *n_estimators* = 50, *max_depth* = none, *min_samples_split* = 2, dan *min_samples_leaf* = 1. Diagram pada gambar 4 menunjukkan perbedaan performa klasifikasi antara kedua algoritma tersebut.



Gambar 4. Diagram Klasifikasi Perbandingan SVM dan RF

Secara visual pada Gambar 4, menyajikan perbandingan kinerja antara algoritma *Random Forest (RF)* dan *Support Vector Machine (SVM)* dalam mengklasifikasikan data penyakit ginjal kronis. Berdasarkan gambar 4, algoritma *Random Forest* (warna biru) secara konsisten menunjukkan nilai yang lebih tinggi pada semua metrik evaluasi dibandingkan dengan *SVM* (warna oranye). Berdasarkan diagram pada Gambar 4, perbedaan nilai parameter evaluasi dapat disajikan kedalam bentuk tabel, seperti yang ditunjukkan pada Tabel VI berikut ini:

TABEL VI. PERFORMA KLASIFIKASI *SUPPORT VECTOR MACHINE* DAN *RANDOM FOREST*

Parameter evaluasi	***Support Vector Machine***	***Random Forest***
Akurasi	97.50%	98.75%
Precision	97.06%	98.48%
Recall	97.92%	98.96%
F1-score	97.42%	98.70%

Hasil dari perbandingan antara algoritma *Support Vector Machine* (SVM) dengan algoritma *Random Forest* (RF), mengindikasikan bahwa algoritma *Random Forest* lebih baik dalam mengklasifikasikan data penyakit ginjal kronis pada dataset yang digunakan dalam penelitian ini.

4. KESIMPULAN DAN SARAN

Penelitian ini bertujuan untuk mengevaluasi kemanjuran dua algoritma klasifikasi yang berbeda antara *"Support Vector Machine"* (SVM) dan *"Random Forest"* (RF) dalam mendeteksi penyakit ginjal kronis (PGK). Analisis hasil menghasilkan beberapa wawasan yang signifikan:

1. *Pre-processing* Data: Tahap ini sangat penting untuk memastikan integritas dan kebersihan data, yang pada gilirannya meningkatkan keakuratan hasil prediktif. Langkah-langkah *pre-processing* meliputi penanganan nilai yang hilang, normalisasi data, dan transformasi fitur kategori.
2. Performa Algoritma:
   a. *Support Vector Machine (SVM)*: Algoritma SVM dengan optimasi *hyperparameter* melalui *GridSearchCV* menghasilkan akurasi sebesar 97.50%, *precision* 97.06%, *recall* 97.92%, dan *F1-score* 97.42%.
   b. *Random Forest (RF)*: Algoritma *Random Forest* (RF), yang dioptimalkan melalui penyetelan hiperparameter, mencapai akurasi 98,75%, *precicison* 98,48%, *recall* 98,96%, dan *F1-score* 98,70%.
3. Pemilihan Model yang Optimal: Setelah mengevaluasi akurasi, *precision*, dan *F1-score*, algoritma *Random Forest* (RF) muncul sebagai yang lebih unggul dibandingkan dengan *Support Vector Machine* (SVM).

"Model RF menunjukkan kinerja yang lebih tinggi di semua metrik, menjadikannya pilihan yang lebih efektif untuk mengklasifikasikan penyakit ginjal kronis dalam kumpulan data ini.

Kesimpulan dari penelitian ini adalah bahwa Algoritma *Random Forest* (RF) menunjukkan kinerja yang unggul dalam mengklasifikasikan penyakit ginjal kronis jika dibandingkan dengan *Support Vector Machine* (SVM). RF menunjukkan akurasi, *precision, recall*, dan *F1-score* yang lebih baik. Studi ini menggarisbawahi pentingnya memilih algoritme yang tepat dan mengoptimalkan parameternya untuk mencapai hasil prediktif yang tepat dan andal.

Untuk memperkuat validitas hasil, disarankan agar algoritma *Random Forest (RF)* dan *Support Vector Machine* (SVM) diaplikasikan pada dataset yang lebih besar dan beragam, yang memungkinkan analisis lebih komprehensif serta memastikan keandalan model dalam berbagai kondisi. Selain itu, penggunaan metode ensemble learning dengan menggabungkan RF dan SVM atau algoritma lainnya dapat dipertimbangkan untuk mendapatkan hasil yang lebih stabil. Walaupun optimasi *hyperparameter* telah dilakukan, disarankan untuk terus mengeksplorasi kombinasi parameter yang lebih luas menggunakan teknik optimasi lain seperti Random Search atau Bayesian Optimization, guna menemukan konfigurasi model yang mungkin lebih optimal.

DAFTAR PUSTAKA:

[1] Y. Widjaja *et al.*, "Peningkatan Kewaspadaan Masyarakat Terhadap Penyakit Ginjal Kronis Dengan Edukasi Gaya Hidup dan Skrining Fungsi Ginjal," *Communnity Development Journal*, vol. 4, no. 6, pp. 12147–12153, 2023, doi: doi.org/10.31004/cdj.v4i6.22087.

[2] L. C. Afrilia, J. T. Atmojo, and A. S. Mubarok, "Efektifitas Continuous Ambulatory Peritoneal Dialysis (CAPD) pada Penderita Gagal Ginjal: Litelatur Review," *Journal of Language and Health*, vol. 5, no. 2, pp. 401–412, 2024, doi: doi.org/10.37287/jlh.v5i2.3521.

[3] E. D. Kusumaningrum and S. Sariono, "Evidence Base Nursing Penggunaan Mobile Health Dialysis untuk Evaluasi Mandiri Adekuasi Hemodialisis pada Pasien Gagal Ginjal Kronik," *Malahayati Nursing Journal*, vol. 5, no. 8, pp. 2580–2588, Aug. 2023, doi: 10.33024/mnj.v5i8.9519.

[4] H. D. Siswaja and Y. Ramdhani, "Pendekatan Algoritma Neural Network dan Genetic Algorithm Untuk Prediksi Penyakit Ginjal Kronis," *JURNAL RESPONSIF*, vol. 6, no. 2, pp. 232–239, 2024, doi: doi.org/10.51977/jti.v6i2.1778.

[5] M. D. Purbolaksono, M. Irvan Tantowi, A. Imam Hidayat, and A. Adiwijaya, "Perbandingan Support Vector Machine dan Modified Balanced Random Forest dalam Deteksi Pasien Penyakit Diabetes," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 2, pp. 393–399, Apr. 2021, doi: 10.29207/resti.v5i2.3008.

[6] A. Setiawan, R. Febrio Waleska, M. Adji Purnama, and L. Efrizoni, "Komparasi Algoritma K-Nearest Neighbor (K-NN), Support Vector Machine (SVM), Dan Decision Tree Dalam Klasifikasi Penyakit Stroke," *Jurnal Informatika & Rekayasa Elektronika)*, vol. 7, no. 1, p. 3, 2024, doi: doi.org/10.36595/jire.v7i1.1161.

[7] M. Nurkholifah, Jasmarizal, Y. Umar, and Rahmaddeni, "Analisa Performa Algoritma Machine Learning Dalam Prediksi Penyakit Liver," *Jurnal Indonesia : Manajemen Informatika dan Komunikasi*, vol. 4, no. 1, pp. 164–172, Jan. 2023, doi: 10.35870/jimik.v4i1.149.

[8] Firman Akbar and Rahmaddeni, "Komparasi Algoritma Machine Learning Untuk Memprediksi Penyakit Alzheimer," *Jurnal Komputer Terapan*, vol. 8, no. 2, pp. 236–245, Nov. 2022, doi: 10.35143/jkt.v8i2.5713.

[9] C. Chairunnisa, I. Ernawati, and M. M. Santoni, "Klasifikasi Sentimen Ulasan Pengguna Aplikasi PeduliLindungi di Google Play Menggunakan Algoritma Support Vector Machine dengan Seleksi Fitur Chi-Square," *Informatik : Jurnal Ilmu Komputer*, vol. 18, no. 1, pp. 69–79, 2022, doi: 10.52958/iftk.v17i4.4594.

[10] I. Siti Aisah, B. Irawan, and T. Suprapti, "ALGORITMA SUPPORT VECTOR MACHINE (SVM) UNTUK ANALISIS SENTIMEN ULASAN APLIKASI AL QUR'AN DIGITAL," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 6, pp. 3759–3765, Feb. 2024, doi: 10.36040/jati.v7i6.8263.

[11] V. Singh, V. K. Asari, and R. Rajasekaran, "A Deep Neural Network for Early Detection and Prediction of Chronic Kidney Disease," *Diagnostics*, vol. 12, no. 1, p. 116, Jan. 2022, doi: 10.3390/diagnostics12010116.

[12] N. Giarsyani, A. F. Hidayatullah, and R. Rahmadi, "Komparasi Algoritma Machine Learning dan Deep Learning Untuk Named Entity Recognition : Studi Kasus Data Kebencanaan," *Jurnal Informatika dan Rekayasa Elektronik*, vol. 3, no. 1, pp. 48–57, 2020.

[13] C. Z. V. Junus, T. Tarno, and P. Kartikasari, "Klasifikasi Menggunakan Metode Support Vector Machine dan Random Forest Untuk Deteksi Awal Risiko Diabetes Melitus," *Jurnal Gaussian*, vol. 11, no. 3, pp. 386–396, Jan. 2023, doi: 10.14710/j.gauss.11.3.386-396.

[14] H. Mulyo and N. A. Maori, "Peningkatan Akurasi Prediksi Pemilihan Program Studi Calon Mahasiswa Baru Melalui Optimasi Algoritma Decision Tree Dengan Teknik Pruning dan Ensemble," *Jurnal Disprotek*, vol. 15, no. 1, pp. 15–25, 2024, doi: doi.org/10.34001/jdpt.v15i1.

[15] G. A. Sandag, "Prediksi Rating Aplikasi App Store Menggunakan Algoritma Random Forest," *CogITo Smart Journal*, vol. 6, no. 2, pp. 167–178, Dec. 2020, doi: 10.31154/cogito.v6i2.270.167-178.

[16] D. Normawati and S. A. Prayogi, "Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," *Jurnal Sains Komputer & Informatika (J-SAKTI*, vol. 5, no. 2, pp. 697–711, 2021, doi: dx.doi.org/10.30645/j-sakti.v5i2.369.