**Building of Informatics, Technology and Science (BITS)**
Volume 6, No 2, September 2024, Page: 1159−1168
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i2.5543

# Perbandingan Prediksi Penyakit Stunting Balita Menggunakan Algoritma Support Vektor Machine dan Random Forest

**Yunada Wiratama[1], RZ Abdul Aziz[2*]**

[1,2]Fakultas Ilmu komputer, Magister Teknik Informatika, IBI Darmajaya, Kota Bandar Lampung, Indonesia
Email: [1]yunadawiratama@gmail.com, [2,*]rz_aziz@darmajaya.ac.id
Submitted: **11/07/2024**; Accepted: **27/09/2024**; Published**: 30/09/2024**

**Abstrak**−Stunting pada balita merupakan masalah kesehatan yang serius, terutama di negara berkembang, dimana balita mengalami pertumbuhan yang terhambat akibat kekurangan gizi kronis. Kondisi ini tidak hanya mempengaruhi tinggi badan anak, tetapi juga perkembangan kognitif dan kesehatannya secara keseluruhan. Mengidentifikasi faktor risiko dan melakukan klasifikasi stunting dapat membantu dalam penanganan dan pencegahan masalah ini. Dalam penelitian ini, kami menerapkan dua metode machine learning untuk di bandingkan mana yang lebih baik klasifikasi dari dua metode ini, yaitu Random Forest dan Support Vector Machine (SVM), untuk melakukan klasifikasi penyakit stunting pada balita. Data yang digunakan merupakan data publik berjumlah 97.873 data. Setelah melalui tahap preprocessing, seperti pembersihan data, normalisasi, dan pembagian data, data dibagi menjadi set pelatihan dan pengujian. Model Random Forest dan SVM kemudian dilatih dengan menggunakan set pelatihan dan dievaluasi menggunakan metrik seperti akurasi, precision, dan recall. Hasil analisis menunjukkan bahwa kedua metode memiliki kinerja yang baik dalam mengklasifikasikan stunting pada balita, dengan hasil Random Forest mencapai akurasi 0,9997 dan SVM mencapai akurasi 0,9951. Temuan ini diharapkan dapat membantu dalam pengembangan strategi intervensi yang lebih efektif untuk mengatasi masalah stunting pada balita. Dengan adanya pendekatan ini, diharapkan dapat memberikan kontribusi yang signifikan dalam menurunkan prevalensi stunting di negara berkembang dan meningkatkan kualitas hidup anak-anak di masa mendatang. Selain itu, penelitian ini juga membuka peluang untuk eksplorasi lebih lanjut dalam penggunaan teknik machine learning lainnya untuk masalah kesehatan lainnya.

**Kata Kunci:** SVM,Random Forest,Stunting,Machine learning,Accuracy.

**Abstract**−Stunting in toddlers is a serious health problem, especially in developing countries, where toddlers experience stunted growth due to chronic malnutrition. This condition not only affects the child's height but also their cognitive development and overall health. Identifying risk factors and classifying stunting can help in addressing and preventing this issue. In this study, we applied two machine learning methods to compare which one performs better in classification, namely Random Forest and Support Vector Machine (SVM), to classify stunting in toddlers. The data used is public data consisting of 97,873 entries. After undergoing preprocessing steps such as data cleaning, normalization, and splitting, the data was divided into training and testing sets. The Random Forest and SVM models were then trained using the training set and evaluated using metrics such as accuracy, precision, and recall. The analysis results showed that both methods perform well in classifying stunting in toddlers, with Random Forest achieving an accuracy of 0.9997 and SVM achieving an accuracy of 0.9951. These findings are expected to aid in the development of more effective intervention strategies to address stunting in toddlers. With this approach, it is hoped to make a significant contribution to reducing the prevalence of stunting in developing countries and improving the quality of life for children in the future. Additionally, this research opens opportunities for further exploration of other machine learning techniques for other health issues.
**Keywords**: SVM,Random Forest,Stunting,Machine learning,Accuracy.

## 1. PENDAHULUAN

Stunting pada balita merupakan salah satu masalah kesehatan serius yang dihadapi oleh banyak negara berkembang, termasuk Indonesia [1]. Stunting adalah kondisi dimana tinggi badan seorang anak lebih rendah dari standar usianya, akibat kekurangan gizi kronis dan infeksi berulang selama periode paling awal pertumbuhan dan perkembangan [2]. Berdasarkan data terbaru dari Kementerian Kesehatan Republik Indonesia pada tahun 2023, prevalensi stunting pada balita masih berada pada angka yang mengkhawatirkan, yaitu sekitar 24,4% [3]. Kondisi ini tidak hanya mempengaruhi perkembangan fisik anak, tetapi juga berdampak pada perkembangan kognitif dan kesejahteraan secara keseluruhan [4].

Penanganan stunting memerlukan pendekatan yang komprehensif, salah satunya adalah melalui prediksi dini yang akurat untuk mengidentifikasi anak-anak yang berisiko tinggi mengalami stunting. Dengan prediksi dini, [5]. sebagai referensi peneliti yaitu: “Prediksi Stunting Pada Balita Dengan Menggunakan Algoritma Klasifikasi Naïve Bayes” [6], penelitian ini membahas tentang mengurangi angka stunting pada balita dengan bantuan *machine learning* dengan menggunakan data sebanyak 22855 data, intervensi dapat dilakukan lebih cepat dan efektif. Dalam konteks ini, teknologi machine learning menawarkan solusi yang potensial. Algoritma Support Vector Machine (SVM) dan Random Forest (RF) telah terbukti efektif dalam berbagai studi untuk klasifikasi dan prediksi berbasis data. Support Vector Machine (SVM) [7] adalah salah satu metode klasifikasi yang bekerja dengan cara mencari hyperplane terbaik yang memisahkan data ke dalam dua kelas.

SVM dikenal karena kemampuannya dalam menangani data berdimensi tinggi dan bekerja dengan baik pada dataset yang relatif kecil namun dengan jumlah fitur yang besar. Di sisi lain, Random Forest (RF) [8] adalah algoritma ensemble learning yang menggunakan kombinasi beberapa pohon keputusan (decision trees) untuk meningkatkan akurasi prediksi dan mengurangi risiko overfitting. RF memiliki keunggulan dalam menangani data yang kompleks

**Building of Informatics, Technology and Science (BITS)**
Volume 6, No 2, September 2024, Page: 1159−1168
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i2.5543

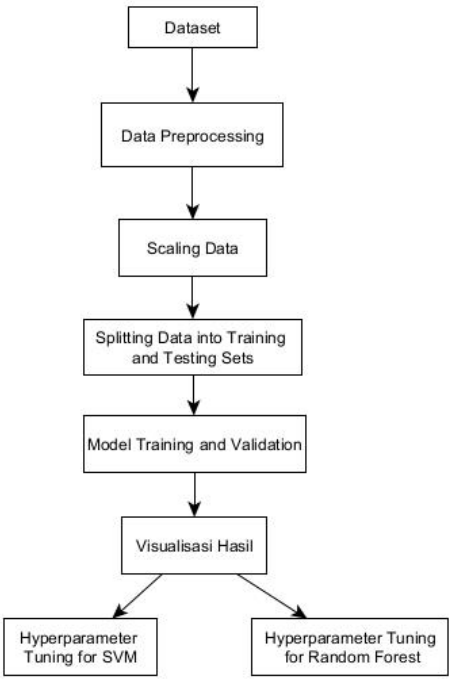
dan tidak terstruktur. Penelitian ini bertujuan untuk menerapkan algoritma SVM dan RF dalam memprediksi risiko stunting pada balita menggunakan data terbaru.

Dengan memanfaatkan algoritma-algoritma ini, diharapkan dapat diperoleh model prediksi yang akurat dan dapat digunakan sebagai alat bantu dalam pengambilan keputusan oleh pihak-pihak terkait dalam upaya penanggulangan stunting. Data yang digunakan dalam penelitian ini mencakup berbagai faktor yang berkontribusi terhadap stunting, termasuk status gizi, kesehatan ibu, kondisi ekonomi keluarga, dan lingkungan tempat tinggal. Melalui pendekatan ini, diharapkan penelitian ini tidak hanya memberikan kontribusi dalam pengembangan model prediksi stunting, tetapi juga memberikan wawasan baru tentang faktor-faktor yang paling signifikan mempengaruhi risiko stunting, sehingga intervensi yang dilakukan dapat lebih tepat sasaran dan efektif dalam jangka panjang

# 2. METODOLOGI PENELITIAN

## 2.1 Tahapan Penelitian

Pada tahapan penelitian terdiri dari beberapa proses tahapan peneltian dimana pada tahapan awal dilakukan pemilihan Dataset. Dataset[9] yang digunakan dalam penelitian ini berasal dari data publik yaitu tentang Stunting Toddler (Balita) Detection yang berjumlah 97873 data, yang mencakup informasi tentang umur (bulan), jenis kelamin, tinggi badan (cm), dan status gizi balita. Dataset ini dipilih karena representatif terhadap populasi balita, memastikan relevansi hasil analisis dalam konteks kesehatan masyarakat. Setelah itu dilakukan[10] pre-pemrosesan data kemudian pembagian data menjadi data training dan data testing, setelah itu melakukan pelatihan model menggunakan Support Vector machine dan Random Forest dan yang terakhir dilakukan evaluasi model[11] serta optimasi menggunakan hyperparameter untuk meningkatkan perfroma model. Berikut gambar 1 adalah alur tahapan penelitiannya.




**Gambar 1.** Tahapan Penelitian

## 2.2 Pre-pemrosesan Data

Pra-pemrosesan data dilakukan untuk memastikan integritas dan kualitas data sebelum analisis lebih lanjut. Langkah penting untuk memastikan integritas dan kualitas data sebelum dilakukan analisis lebih lanjut [12]. Langkah ini melibatkan identifikasi dan penanganan nilai yang hilang, serta normalisasi data untuk menstandarisasi rentang nilai fitur-fitur. Data yang mengandung nilai yang hilang dihapus, sementara normalisasi memungkinkan model untuk belajar dari data dengan lebih baik. Langkah-langkah yang dilakukan meliputi.

a. *Identifikasi dan Penanganan Nilai yang Hilang* : dimana dilakukan identifikasi terhadap nilai yang hilang dalam dataset untuk memastikan tidak ada data yang tidak lengkap [13]. Jika terdapat nilai yang hilang, strategi penanganan yang sesuai diterapkan seperti penghapusan baris atau pengisian nilai yang sesuai berdasarkan konteks data tersebut.
b. *Normalisasi Data :* Pada tahapan ini fitur-fitur seperti Umur (bulan) dan Tinggi Badan (cm) dinormalisasi untuk menstandarisasi rentang nilai mereka [14]. Normalisasi ini penting agar semua fitur memiliki skala yang serupa, memudahkan proses pembelajaran mesin dalam menemukan pola yang relevan dari data.

**Building of Informatics, Technology and Science (BITS)**
Volume 6, No 2, September 2024, Page: 1159−1168
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i2.5543

c. *Konversi Data Kategorikal ke Numerik* : Pada proses ini fitur Jenis Kelamin dan Status Gizi yang awalnya dalam bentuk kategorikal dikonversi menjadi bentuk numerik [15]. Misalnya, Jenis Kelamin yang semula 'laki-laki' dan 'perempuan' dikonversi menjadi nilai biner (0 dan 1), sedangkan Status Gizi seperti 'normal', 'stunting', 'stunted', 'severely stunted', dan 'tinggi' diassign ke nilai-nilai numerik yang sesuai (0, 1, 2, 3, dan 4).
d. *Penanganan Outlier* : Dalam tahapan ini dilakukan penanganan terhadap outlier yang dapat mempengaruhi kualitas hasil analisis [16]. Outlier yang terdeteksi diidentifikasi dan jika perlu, dilakukan strategi penanganan seperti penghapusan atau transformasi data untuk meminimalkan pengaruhnya terhadap model.

Pra-pemrosesan data dilakukan untuk memastikan kebersihan dan kualitas data sebelum digunakan untuk pelatihan model. Setelah melalui tahap-tahap pra-pemrosesan data di atas, dataset telah siap digunakan untuk proses selanjutnya, yaitu pembagian data, pelatihan model, dan evaluasi performa model. Tahap ini penting untuk memastikan data yang digunakan memiliki kualitas yang baik dan siap untuk dieksploitasi secara maksimal dalam analisis dan pemodelan.

Dari hasil pemrosesan yang dilakukan dapat dilihat bahwa data yang digunakan dalam penelitian ini telah bersih dari nilai yang hilang, dinormalisasi untuk fitur-fitur numerik, dan dikonversi ke format yang sesuai untuk analisis lanjutan menggunakan metode Support Vector Machine (SVM) dan Random Forest. Tahapan preprocessing data yang cermat sangat penting untuk memastikan hasil analisis yang akurat sehinggga data dapat proses ke tahap selanjutnya.

### 2.3 Scaling Data

Pada tahapan selanjutnya adalah scaling data merupakan proses untuk menyesuaikan skala atau rentang nilai dari fitur-fitur numerik dalam dataset [17]. Tujuannya adalah untuk memastikan bahwa semua fitur memiliki rentang nilai yang serupa, sehingga tidak ada fitur yang dominan hanya karena memiliki skala yang lebih besar. Ini penting karena beberapa algoritma machine learning, seperti SVM dan Random Forrest, sangat sensitif terhadap skala data. berikut adalah tahapan yang dilakukan pada scaling data.

a. Pemilihan Fitur yang akan di Scaling : Pada tahap ini, fitur yang dipilih untuk dilakukan scaling adalah 'Umur (bulan)' dan 'Tinggi Badan (cm)' [17]. Fitur ini dipilih karena keduanya merupakan variabel numerik yang penting dalam analisis terkait status gizi balita
b. Inisialisasi dan Fitting Standard Scaler : Pertama-tama, dilakukan inisialisasi dari objek Standard Scaler dari library scikit-learn [18]. Standard Scaler digunakan untuk mentransformasi fitur-fitur numerik dengan cara mengubah distribusi data sehingga memiliki mean 0 dan standard deviation 1. Berikut adalah proses pemrogramnya. Pada proses diatas fitting dilakukan dengan memanggil method fit_transform() pada objek scaler, yang menghitung mean dan standard deviation dari fitur-fitur yang dipilih ('Umur (bulan)' dan 'Tinggi Badan (cm)') dari dataset X, dan kemudian melakukan transformasi data untuk mengubah skala fitur-fitur tersebut.
c. Penggantian Nilai Infinite atau NaN: Pada tahapan ini setelah proses scaling, dilakukan penanganan terhadap nilai infinite atau NaN dalam dataset. Ini penting untuk memastikan kebersihan dataset dan mencegah terjadinya masalah saat proses training model machine learning. Pada langkah ini memastikan bahwa data yang digunakan untuk training model tidak mengandung nilai yang tidak valid, seperti nilai infinit atau NaN.
d. Pengecekan Data Setelah Scaling: Setelah scaling selesai, penting untuk memverifikasi hasilnya dengan memeriksa data secara keseluruhan [19]. Berikut adalah tahapan program pengecekan scaling data.

Setelah langkah pemrosesan scaling data dilajankan dengan Fitur-fitur 'Umur (bulan)' dan 'Tinggi Badan (cm)' dalam dataset telah diubah skala nya sehingga memiliki rata-rata nol dan variansi satu, sesuai dengan proses yang dilakukan oleh StandardScaler. Kemduain data yang telah di-scale kemudian siap untuk digunakan dalam proses training model machine learning, seperti SVM dan Random Forest. Dan yang terakhir scaling data membantu memastikan bahwa semua fitur memberikan kontribusi yang seimbang terhadap analisis selanjutnya dan meningkatkan konsistensi serta akurasi model yang dibangun.

### 2.4 Splitting Data into Training and Testing Sets

Splitting Data into Training and Testing Sets dilakan untuk pengembangan model machine learning. Pada tahap ini, dataset yang telah dimiliki dibagi menjadi dua subset utama yaitu data training dan data testing. Data training digunakan untuk melatih model machine learning, sedangkan data testing digunakan untuk menguji atau mengevaluasi performa model yang telah dilatih. Berikut adalah tahapan Splitting Data into Training and Testing Sets dalam peneltian ini.

a. Definisi Features dan Target Variable : Pada tahap ini, dilakukan pengaturan features (variabel independen) dan target variable (variabel dependen) yang akan digunakan untuk training model dan evaluasi performanya.

b. Pembagian Data menjadi Data Training dan Data Testing: Pada proses ini data dibagi menjadi data training dan data testing menggunakan fungsi train_test_split() dari scikit-learn. Data testing digunakan untuk menguji kinerja model yang sudah dilatih.
c. Penanganan Nilai Infinite atau NaN setelah Pembagian Data: Setelah pembagian data, penting untuk memastikan tidak ada nilai infinite atau NaN dalam data training dan data testing yang akan digunakan untuk training dan evaluasi model.

**Building of Informatics, Technology and Science (BITS)**
Volume 6, No 2, September 2024, Page: 1159−1168
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i2.5543

d. Pengecekan Shapes dari Training dan Testing Data: Pada proses terakhir dari Splitting Data into Training and Testing Sets dilakukan pengecekan untuk memastikan bahwa data training dan testing telah dibagi dengan benar dan siap digunakan untuk tahap selanjutnya dalam proses pembuatan model machine learning.

### 2.5 Model Training and Validation

Pada tahapan [20]Model Training and Validation dilakukan pemrosesan pada pelatihan model machine learning menggunakan data training yang telah dipisahkan sebelumnya. Setelah pelatihan selesai maka dilakukan validasi model menggunakan data testing yang juga telah dipisahkan sebelumnya. Proses ini untuk mengukur seberapa baik model yang telah dilatih mampu menggeneralisasi pola dari data training ke data yang belum pernah dilihat sebelumnya. Dan berikut adalah tahapan pada model training dan validasi dimana pada tahapan ini akan dilakukan model training dan validasi pada support vector machine dan juga random forest.

a. Support Vector Machine
   Pada Support Vector Machine digunakan untuk melakukan klasifikasi status gizi berdasarkan fitur-fitur yang tersedia. Yaitu dengan inisialisasi model menggunakan SVC() dari library sklearn.svm untuk membuat model SVM tanpa parameter spesifik. Kemduain setelah itu dialkukan pelatihan model SVM menggunakan data training (X_train, y_train) yang telah dipisahkan sebelumnya. Setelah itu dilakukan prediksi model SVM untuk memprediksi label dari data testing (X_test). Hasil prediksi disimpan dalam variabel y_pred_svm.
   Setelah pemrosesan selanjutnya adalah dilakukan evaluasi model untuk mengukur performa model svm yang dilatih dengan melakukan Classification report untuk menampilkan nilai precision, recall, dan F1-score untuk setiap kelas yang diprediksi. Kemudian Confusion matrix untuk menunjukkan jumlah prediksi yang benar dan salah untuk setiap kelas. Dan yang terakhir adalah Accuracy score untuk menghitung akurasi model, yaitu persentase prediksi yang benar dari total prediksi.
b. Random Forest
   Pada tahap pemrosesan Model Training and Validation Random Forest dilakukan klasifikasi status gizi berdasarkan fitur-fitur yang tersedia. Diantaranya pada inisialisasi model digunakan *RandomForestClassifier()* dari library *sklearn.ensemble* untuk membuat model Random Forest tanpa parameter spesifik. Kemudian pada pelatihan modelnya, Model Random Forest dilatih menggunakan data training (X_trai`, y_train) yang telah dipisahkan sebelumnya. Setelah dilatih, model Random Forest digunakan untuk memprediksi label dari data testing (X_test). Hasil prediksi disimpan dalam variabel y_pred_rf.

Setelah pemrosesan selanjutnya adalah dilakukan evaluasi model untuk mengukur performa model random forest dilakukan evaluasi model untuk mengukur performa model Random Forest yang telah dilatih dengan Classification report untuk menampilkan nilai precision, recall, dan F1-score untuk setiap kelas yang diprediksi. Kemudian Confusion matrix untuk menunjukkan jumlah prediksi yang benar dan salah untuk setiap kelas. Dan yang terakhir Accuracy score untuk menghitung akurasi model, yaitu persentase prediksi yang benar dari total prediksi.

### 2.4 Visualisasi Hasil

Pada tahap visualisasi hasil dilakukan prediksi menggunakan confusion matrix. Visualisasi ini untuk melihat performa model secara lebih intuitif, dengan menampilkan jumlah prediksi yang benar dan salah untuk setiap kelas. Pada tahapan ini sangat penting untuk menganalisis dan memahami bagaimana model pada pemrosesan bekerja. Confusion matrix adalah lebih efektif karena menunjukkan distribusi prediksi dan kesalahan model. Langkah pertama yang dilakukan adalah dengan melakukan pengimporan library yang di perlukan yaitu *matplotlib.pyplo* dan *seaborn.*

Pada Visualisasi Confusion Matrix untuk SVM dan Random forest dilakukan pembuatan plot menggunakan fungsi heatmap dari seaborn untuk membuat heatmap dari confusion matrix kemudain pengaturan tampilan yang disusun dalam dua subplots, satu untuk SVM dan satu untuk Random Forest, agar mudah untuk dibandingkan.

### 2.4 Hyperparameter Tuning SVM dan Random Forest

Hyperparameter tuning adalah proses untuk meningkatkan performa model. Dalam penelitian ini menggunakan Randomized Search untuk mengoptimalkan hyperparameter dari Support Vector Machine (SVM) dan Random Forest (RF). Tujuannya adalah untuk menemukan kombinasi hyperparameter yang memberikan performa terbaik. Hyperparameter tuning dilakukan untuk mencari kombinasi terbaik dari parameter yang tidak dipelajari oleh model selama training. Dengan menggunakan teknik Randomized Search dapat secara efisien menjelajahi ruang parameter yang besar dan menemukan pengaturan terbaik yang meningkatkan kinerja model. Berikut adalah tahapan Hyperparameter tuning dari algortima SVM.

a. Mendefinisikan Parameter Distribution SVM : Mendefinisikan distribusi parameter yang akan dicari oleh Randomized Search. Distribusi parameter untuk SVM termasuk C dan gamma, yang keduanya menggunakan distribusi reciprocal untuk menjelajahi rentang yang besar. Kernel rbf dipilih karena umumnya memberikan performa yang baik untuk SVM.
b. Randomized Search untuk SVM sekaligus hasil : Membuat objek RandomizedSearchCV dengan model SVM, parameter distribution, dan jumlah iterasi pencarian sebanyak 20. Randomized Search kemudian di-fit pada data training untuk menemukan kombinasi parameter terbaik. Pada hasil adalah Proses tuning untuk menghasilkan parameter terbaik untuk SVM.

**Building of Informatics, Technology and Science (BITS)**
Volume 6, No 2, September 2024, Page: 1159−1168
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i2.5543

c. Evaluasi Model SVM dengan Hyperparameter Terbaik: pada tahapan ini melakukan Re-training Model SVM yang di training ulang menggunakan parameter terbaik yang ditemukan. Kemudian dilakukan Evaluasi Model dan kemudian diuji pada data testing dan hasilnya dievaluasi.

Pada pemrosesan selanjutnya yaitu Hyperparameter tuning untuk Random Forest yang tahapanya sama seperti pada pemrosesan Hyperparameter tuning SVM. Berikut adalah tahapan pemrosesan Hyperparameter tuning Random Forest.

a. Mendefinisikan Parameter Distribution untuk Random Forest : Parameter yang dicari meliputi 'n_estimators', 'max_features', dan 'max_depth'. Distribusi parameter ini memungkinkan pencarian yang luas dan efisien dalam ruang parameter.
b. Randomized Search untuk Random Forest : Pada tahapan ini membuat objek RandomizedSearchCV dengan model Random Forest, parameter distribution, dan jumlah iterasi pencarian. Randomized Search kemudian di-fit pada data training. Setelah itu Proses tuning menghasilkan parameter terbaik untuk Random Forest.
c. Evaluasi Model Random Forest dengan Hyperparameter Terbaik: melakukan Re-training Model pada Random Forest yang di training ulang menggunakan parameter terbaik yang ditemukan. Kemduain melakukan Evaluasi Model yang diuji pada data testing dan hasilnya dievaluasi.

### 2.5 *Support Vektor Machine* (SVM)

Support Vector Machine (SVM) merupakan salah satu algoritma machine learning yang paling populer untuk klasifikasi [12]. Selama dekade terakhir, SVM menjadi metode yang kuat untuk pola klasifikasi, memiliki tingkat keberhasilan yang tinggi saat diterapkan diberbagai bidang[13]. Sehingga banyak dari kalangan komunitas machine learning berminat untuk mempelajari dan mengembangkan SVM karena kinerjanya yang sangat baik dalam berbagai masalah pembelajaran[14]. SVM adalah metode learning machine yang bertujuan untuk menemukan hyperplane terbaik yang memisahkan dua buah kelas pada ruang input ( input space). Algoritma klasifikasi SVM menggunakan data training untuk membentuk model klasifikasi, model yang terbentuk digunakan sebagai prediksi kelas data baru yang belum pernah ada sebelumnya yang disebut dengan testing data.

### 2.6 *Random Forest*

*Random Forest* (RF) ialah metode yang dapat menaikannilaiakurasi, sehinggasimpul anak untuk setiap node yangdilakukan secara acakdapatmeningkat, dan diperlukanuntuk membuatpohon keputusan yang terdiri dari internal node,root node, dan leaf nodedengan cara mengambil atribut maupun data secara acak menurutketetapanyang berlaku[15].Random Forestmerupakan algoritma machine learningyang digunakan sebagai klasifikasi, bertugas untuk mengelompokkan data yang bergantung pada kecenderungannya, berisi kumpulan dari decision treeyang beroperasi menjadi suatu gabungan fungsional, dan dapat berjalan efisien pada data yang jumlahnya banyak[16]. Algoritma training untuk random forestmenggunakan bootstrap aggregating (bagging). Proses latih dilakukan dengan mengambil satu set data latih yang kemudian akan dimasukkan ke dalam suatu tree. Pemilihan atribut akan dipecahdandiambil secara acak dalam sebuah node. Baggingmelakukan pemilihan sampel berulang kali, dengan penggantian.

### 2.7 Confusion matrix

Cara kerjaconfusion matrix dengan mengolah data untuk membandingkan hasil prediksi dengan label sesungguhnya. Evaluasi dengan confusion matrix menghasilkan nilai akurasi, presisi dan re-call. Pada evaluasi klasifikasi terdapat empat kemungkinan yang bisa terjadi dari hasil klasifikasi suatu data. Jika data positif dan diprediksi positif maka dihitung sebagai true positive dan jika data positif diprediksi negatif maka akan dihitung sebagai false negative. Pada data negatif jika diprediksi negatif dihitung Sebagai true negative dan jika diprediksi positif maka akan dihitung sebagai false positive seperti terlihat pada tabel 1 berikut.

**Tabel 1.** *Confusion matrix*

| **Actual** | **prediction** | |
|---|---|---|
| | Positif | Negative |
| **Positif** | True Positif (TP) | True Negative (TN) |
| **Negative** | False Positif (FP) | Fasle Negatif (FN) |

## 3. HASIL DAN PEMBAHASAN

Dalam hasil dan pembahasan akan membahas hasil dari serangkaian proses analisis data yang telah dilakukan untuk mengklasifikasikan status gizi anak berdasarkan dataset yang tersedia. Proses ini mencakup import library dan load dataset, preprocessing data, scaling data, splitting data, training dan validasi model, visualisasi hasil, serta tuning hyperparameter untuk model *Support Vector Machine* (SVM) dan *Random Forest* (RF). Berikut adalah penjelasan mendetail dari setiap langkah dan hasilnya.

**Building of Informatics, Technology and Science (BITS)**
Volume 6, No 2, September 2024, Page: 1159−1168
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i2.5543

### 3.1 Pengimporan Dataset

Langkah pertama dalam proses ini adalah mengimpor library yang diperlukan dan memuat dataset. Dataset yang digunakan berisi informasi tentang umur, jenis kelamin, tinggi badan, dan status gizi anak-anak. Gambar 2 berikut adalah tampilan dari isi dataset tersebut

| | Umur (bulan) | Jenis Kelamin | Tinggi Badan (cm) | Status Gizi |
|---|---|---|---|---|
| 0 | 0 | laki-laki | 44.591973 | stunted |
| 1 | 0 | laki-laki | 56.705203 | tinggi |
| 2 | 0 | laki-laki | 46.863358 | normal |
| 3 | 0 | laki-laki | 47.508026 | normal |
| 4 | 0 | laki-laki | 42.743494 | severely stunted |
| ... | ... | ... | ... | ... |
| 97868 | 49 | laki-laki | 107.000000 | normal |
| 97869 | 49 | laki-laki | 101.000000 | normal |
| 97870 | 49 | laki-laki | 92.000000 | stunted |
| 97871 | 49 | laki-laki | 89.700000 | severely stunted |
| 97872 | 49 | laki-laki | 89.500000 | severely stunted |

[97873 rows x 4 columns]

**Gambar 2.** Dataset

Dalam hasil dataset yang di ambil terdapat data yang digunakan berisi 97873 baris dan 4 kolom dengan distribusi sebagai berikut:

a. Umur (bulan): 0 hingga 49 bulan
b. Jenis Kelamin: Laki-laki dan perempuan
c. Tinggi Badan (cm): 40.01 cm hingga 120.00 cm
d. Status Gizi: Stunted, tinggi, normal, severely stunted

### 3.2 Data Preprocessing

Pada tahap preprocessing, data dibersihkan dan dipersiapkan untuk analisis lebih lanjut. Langkah-langkah yang dilakukan termasuk pemeriksaan nilai yang hilang, statistik deskriptif, dan pengkodean variabel kategoris. Berikut pada gambar 3 merupakan hasil dari pemrosesan data preprocessing yang telah dilakukan.

```
Missing values in the dataset:
 Umur (bulan)         0
Jenis Kelamin        0
Tinggi Badan (cm)    0
Status Gizi          0
dtype: int64
Dataset description:
        Umur (bulan)  Tinggi Badan (cm)
count  97873.000000       97873.000000
mean      24.376754          84.467260
std       14.261700          15.910840
min        0.000000          40.010437
25%       12.000000          73.900000
50%       24.000000          85.400000
75%       37.000000          96.100000
max       49.000000         120.000000
Unique values in 'Status Gizi':
 ['stunted' 'tinggi' 'normal' 'severely stunted']
Data after cleaning:
    Umur (bulan)  Jenis Kelamin  Tinggi Badan (cm)  Status Gizi
0             0              0          44.591973            2
1             0              0          56.705203            4
2             0              0          46.863358            0
3             0              0          47.508026            0
4             0              0          42.743494            3
Data shape after cleaning: (97873, 4)
```

**Gambar 3.** Hasil Preprocessing

Dari hasil preprocessing yang telah dilakukan Tidak ada nilai yang hilang dalam dataset. Kemudian Distribusi umur, tinggi badan, dan status gizi anak-anak telah dianalisis.selanjutnya Kolom Jenis Kelamin dan Status Gizi diubah menjadi nilai numerik untuk memudahkan proses analisis berikutnya. Dan kemudian setelah tahapan preprocessing Jumlah baris dan kolom adalah 97873 x 4. Setelah itu Jenis Kelamin: 0 untuk laki-laki, 1 untuk perempuan. Dan yang terakhir pada label diberi nilai numerik Status Gizi: 0 untuk normal, 1 untuk tinggi, 2 untuk stunted, 3 untuk severely stunted untuk memudahkan dalam proses analisis berikutnya.

**Building of Informatics, Technology and Science (BITS)**
Volume 6, No 2, September 2024, Page: 1159−1168
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i2.5543

### 3.3 Scaling Data

Pada tahap Scaling Data fitur numerik seperti umur dan tinggi badan di-scale menggunakan StandardScaler. Scaling membantu untuk menormalisasi data sehingga setiap fitur memiliki skala yang sama, yang penting untuk kinerja model machine learning. Gambar 4 berikut adalah hasil dari pemrosesannya.

```
Data after scaling:
    Umur (bulan)  Jenis Kelamin  Tinggi Badan (cm)  Status Gizi
0              0              0          44.591973            2
1              0              0          56.705203            4
2              0              0          46.863358            0
3              0              0          47.508026            0
4              0              0          42.743494            3
Data shape after scaling: (97873, 4)
```

**Gambar 4.** Hasil Scaling Data

Dari hasil scaling data yang telah di proses dapat dilihat bahwa Data umur dan tinggi badan telah di-scale kemdian Dataset setelah scaling tetap memiliki 97873 baris dan 4 kolom.

### 3.4 Splitting Data into Training and Testing Sets

Dalam hasil Splitting Data into Training and Testing Sets dataset dibagi menjadi set training dan set testing dengan perbandingan 80:20. Set training digunakan untuk melatih model, sedangkan set testing digunakan untuk mengevaluasi kinerja model. Gambar 5 berikut adalah hasil dari pemrosesannya.

```
Training data shape: (78298, 3) (78298,)
Testing data shape: (19575, 3) (19575,)
```

**Gambar 5.** Splitting Data into Training and Testing Sets

Dari hasil pemrosesan Splitting Data into Training and Testing Sets pada gambar 27 dapat dilihat Training data: 78298 data dan Testing data: 19575 data

### 3.5 Model Training and Validation

Pada pemrosesan model training and validation Dua model machine learning yaitu SVM dan Random Forest, dilatih menggunakan set training dan divalidasi menggunakan set testing. Berikut gambar 6 adalah hasil dari pemrosesanya.

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

           0       0.99      0.99      0.99     10488
           2       0.97      0.94      0.96      2247
           3       0.98      0.99      0.98      3400
           4       0.99      0.98      0.99      3440

    accuracy                           0.99     19575
   macro avg       0.98      0.98      0.98     19575
weighted avg       0.99      0.99      0.99     19575

SVM Confusion Matrix:
 [[10428    24     3    33]
 [   52  2120    75     0]
 [    0    34  3366     0]
 [   53     0     0  3387]]
SVM Accuracy Score: 0.9860025542784163
Random Forest Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     10488
           2       1.00      1.00      1.00      2247
           3       1.00      1.00      1.00      3400
           4       1.00      1.00      1.00      3440

    accuracy                           1.00     19575
   macro avg       1.00      1.00      1.00     19575
weighted avg       1.00      1.00      1.00     19575

Random Forest Confusion Matrix:
 [[10484     4     0     0]
 [    0  2246     1     0]
 [    0     1  3399     0]
 [    0     0     0  3440]]
Random Forest Accuracy Score: 0.9996934865900383
```

**Gambar 6.** Hasil Model Training and Validation

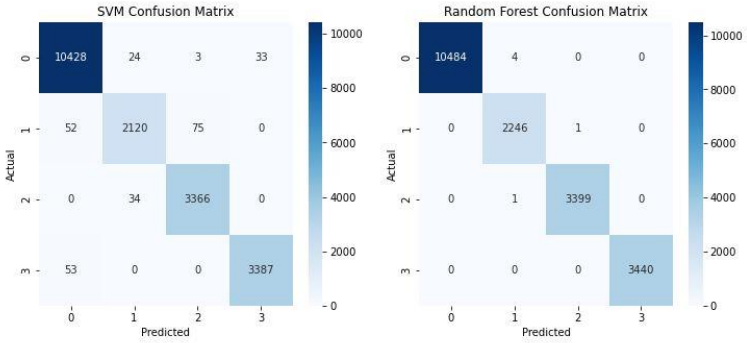
Dari hasil pemrosesan Model Training and Validation dapat dilihat Accuracy Score SVM adalah 0.9860 dengan Precision, Recall, dan F1-score tinggi untuk semua kelas. Kemudian Confusion Matrix menunjukkan sebagian besar prediksi benar. Dan Tingkat kesalahan rendah menunjukkan model SVM cukup andal. Kemduain pada Hasil Random Forest Accuracy Score sebesar 0.9997 dengan Precision, Recall, dan F1-score sangat tinggi untuk semua

**Building of Informatics, Technology and Science (BITS)**
Volume 6, No 2, September 2024, Page: 1159−1168
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i2.5543

kelas. Kemudian Confusion Matrix hampir sempurna. Dan Model Random Forest menunjukkan kinerja luar biasa dengan kesalahan minimal

### 3.6 Visualisasi Hasil

Pada tahap ini, hasil dari model SVM dan Random Forest divisualisasikan dalam bentuk grafik, seperti confusion matrix dan grafik evaluasi lainnya. Berikut adalah hasilnya.




**Gambar 7.** Visualisasi Hasil

Dalam visualisasi pada masing masing algoritma Confusion matrix menunjukkan distribusi prediksi yang akurat dari kedua model kemdian Visualisasi ini membantu dalam memahami kekuatan dan kelemahan masing-masing model.

### 3.7 Hyperparameter Tuning for SVM and Random Forest

Dalam hasil Proses tuning hyperparameter untuk SVM dan Random Forest dilakukan menggunakan Randomized Search untuk menemukan kombinasi parameter terbaik dan hasilnya sebagai berikut.

```
[CV] END C=7.762771535562091, gamma=0.15317914902340293, kernel=rbf; total time=  35.5s
[CV] END C=7.762771535562091, gamma=0.15317914902340293, kernel=rbf; total time=  35.4s
[CV] END C=7.762771535562091, gamma=0.15317914902340293, kernel=rbf; total time=  35.4s
[CV] END C=252.83369130515717, gamma=0.17916899708157977, kernel=rbf; total time=  13.2s
[CV] END C=252.83369130515717, gamma=0.17916899708157977, kernel=rbf; total time=  13.2s
[CV] END C=252.83369130515717, gamma=0.17916899708157977, kernel=rbf; total time=  13.2s
[CV] END C=252.83369130515717, gamma=0.17916899708157977, kernel=rbf; total time=  13.5s
[CV] END C=252.83369130515717, gamma=0.17916899708157977, kernel=rbf; total time=  13.6s
Best Parameters for SVM: {'C': 669.0361942484437, 'gamma': 0.5743043196363885, 'kernel': 'rbf'}

Optimized SVM Accuracy Score: 0.9950957854406131
```

**Gambar 8.** Hyperparameter Tuning for SVM

Dari hasil Hyperparameter Tuning for SVM pada gambar 30 dapat dilihat hasil dari C adalah 699.063 sementara nilai Gamma nya 0.574 dengan kernel rbf kemudian pada model SVM dengan parameter terbaik menghasilkan akurasi yang lebih tinggi yaitu 0.9951. Selanjutnya Hyperparameter Tuning for Random Forest sebagai berikut yang terlihat pada gambar 9.

```
[CV] END ..max_depth=15, max_features=None, n_estimators=401; total time=  23.1s
[CV] END ..max_depth=15, max_features=None, n_estimators=401; total time=  23.1s
[CV] END ..max_depth=15, max_features=None, n_estimators=401; total time=  23.2s
[CV] END ..max_depth=41, max_features=None, n_estimators=196; total time=  11.3s
[CV] END ..max_depth=41, max_features=None, n_estimators=196; total time=  11.4s
[CV] END ..max_depth=41, max_features=None, n_estimators=196; total time=  11.3s
[CV] END ..max_depth=41, max_features=None, n_estimators=196; total time=  11.2s
[CV] END ..max_depth=41, max_features=None, n_estimators=196; total time=  11.3s
[CV] END ..max_depth=36, max_features=log2, n_estimators=487; total time=  18.7s
[CV] END ..max_depth=36, max_features=log2, n_estimators=487; total time=  18.5s
[CV] END ..max_depth=36, max_features=log2, n_estimators=487; total time=  18.2s
[CV] END ..max_depth=36, max_features=log2, n_estimators=487; total time=  18.9s
[CV] END ..max_depth=36, max_features=log2, n_estimators=487; total time=  18.6s
[CV] END ..max_depth=45, max_features=log2, n_estimators=196; total time=   7.4s
[CV] END ..max_depth=45, max_features=log2, n_estimators=196; total time=   7.4s
[CV] END ..max_depth=45, max_features=log2, n_estimators=196; total time=   7.7s
[CV] END ..max_depth=45, max_features=log2, n_estimators=196; total time=   7.9s
[CV] END ..max_depth=45, max_features=log2, n_estimators=196; total time=   8.0s
Best Parameters for Random Forest: {'max_depth': 19, 'max_features': 'sqrt', 'n_estimators': 311}

Optimized Random Forest Accuracy Score: 0.9997445721583652
```

**Gambar 9.** Hyperparameter Tuning for Random Fores

**Building of Informatics, Technology and Science (BITS)**
Volume 6, No 2, September 2024, Page: 1159−1168
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i2.5543

Dari hasil Hyperparameter Tuning for Random Fores pada gambar 31 dapat dilihat hasil dari Max depth adalah 19 kemudian Max features adalah sqrt dan N estimators adalah 311 kemudian Model Random Forest dengan parameter terbaik menghasilkan akurasi yang hampir sempurna yaitu 0.9997

## 4. KESIMPULAN

Dalam penelitian ini, kami berhasil mengklasifikasikan status gizi anak-anak dengan tingkat akurasi yang tinggi menggunakan model SVM dan Random Forest. Proses hyperparameter tuning lebih lanjut meningkatkan kinerja kedua model. Model Random Forest menunjukkan hasil yang luar biasa dengan akurasi hampir sempurna, sementara SVM juga menunjukkan kinerja yang sangat baik. Proses preprocessing, yang meliputi pembersihan data, normalisasi, dan scaling, serta pembagian data menjadi set pelatihan dan pengujian, dilakukan dengan hati-hati untuk memastikan integritas data. Langkah-langkah ini sangat penting untuk menghindari bias dan overfitting dalam model. Selain itu, proses training dan validasi dilakukan secara ekstensif untuk mengevaluasi performa model pada data yang belum pernah dilihat sebelumnya. Hyperparameter tuning, yang melibatkan penyesuaian parameter-parameter model untuk mendapatkan hasil terbaik, memainkan peran krusial dalam mencapai akurasi tinggi. Kombinasi dari semua tahapan ini menghasilkan model yang sangat andal dalam mengklasifikasikan status gizi anak-anak. Hasil analisis menunjukkan bahwa perbandingan kedua metode memiliki kinerja yang baik dalam mengklasifikasikan stunting pada balita akan tetapi disini metode random forest memiliki akurasi lebih baik di bandingankan SVM, dengan hasil Random Forest mencapai akurasi 0,9997 dan SVM mencapai akurasi 0,9951. Temuan ini diharapkan dapat membantu dalam pengembangan strategi intervensi yang lebih efektif untuk mengatasi masalah stunting pada balita. Dengan adanya pendekatan ini, diharapkan dapat memberikan kontribusi yang signifikan dalam menurunkan prevalensi stunting di negara berkembang dan meningkatkan kualitas hidup anak-anak di masa mendatang. Selain itu, penelitian ini juga membuka peluang untuk eksplorasi lebih lanjut dalam penggunaan teknik machine learning lainnya untuk masalah kesehatan lainnya.

## REFERENCES

[1] D. R. H. Sitompul, D. J. Ziegel, and E. Indra, "Perbandingan Algoritma K-Nearest Neighbors (K-NN) dan Random forest terhadap Penyakit Gagal Jantung," *Jurnal Teknlogi Informatika dan Komputer MH. Thamrin*, vol. 9, no. 1, pp. 471–486, 2023.

[2] S. Lonang and D. Normawati, "Klasifikasi Status Stunting Pada Balita Menggunakan K-Nearest Neighbor Dengan Feature Selection Backward Elimination," *Jurnal Media Informatika Budidarma*, vol. 6, no. 1, pp. 49–56, 2022.

[3] S. Handayani, "Selamatkan Generasi Bangsa Dari Bahaya Stunting: Save The Nation's Generation From The Dangers of Stunting," *Journal of Midwifery Science and Women's Health*, vol. 3, no. 2, pp. 87–92, 2023.

[4] N. O. Nirmalasari, "Stunting pada anak: Penyebab dan faktor risiko stunting di Indonesia," *Qawwam*, vol. 14, no. 1, pp. 19–28, 2020.

[5] A. P. Mardin, R. Z. A. Aziz, and A. Kurniawan, "Performance Analysis of Graph Database and Relational Database," in Proceeding International Conference on Information Technology and Business, 2020, pp. 89–94.

[6] H. H. Sutarno, R. Latuconsina, and A. Dinimaharawati, "Prediksi Stunting Pada Balita Dengan Menggunakan Algoritma Klasifikasi K-Nearest Neighbors," *eProceedings of Engineering*, vol. 8, no. 5, 2021.

[7] M. S. Hasibuan and R. Z. A. Aziz, "Detection of learning styles with prior knowledge data using the SVM, K-NN and Naïve Bayes algorithms," Jurnal Infotel, vol. 14, no. 3, pp. 209–213, 2022.

[8] M. S. Hasibuan, R. Z. abdul Aziz, D. Naista, and N. A. Syafira, "Implementation Of A Classification Algorithm To Detect Felder-Silverman Learning Style" 2023.

[9] M. S. Hasibuan and R. Z. A. Aziz, "Detection of learning styles with prior knowledge data using the SVM, K-NN and Naïve Bayes algorithms," Jurnal Infotel, vol. 14, no. 3, pp. 209–213, 2022.

[10] E. T. Handayani and A. Sulistiyawati, "Analisis Setimen Respon Masyarakat Terhadap Kabar Harian Covid-19 Pada Twitter Kementerian Kesehatan Dengan Metode Klasifikasi Naive Bayes," *Jurnal Teknologi Dan Sistem Informasi*, vol. 2, no. 3, pp. 32–37, 2021.

[11] A. M. Pravina, I. Cholisoddin, and P. P. Adikara, "Analisis sentimen tentang opini maskapai penerbangan pada dokumen twitter menggunakan algoritme support vector machine (svm)," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 3, no. 3, pp. 2789–2797, 2019.

[12] A. Zulfiani and C. Fauzi, "Penerapan Algorimta Backpropagation Untuk Prakiraan Cuaca Harian Dibandingkan Dengan Support Vector Machine dan Logistic Regression," JURNAL MEDIA INFORMATIKA BUDIDARMA, vol. 7, no. 3, pp. 1229–1237, 2023.

[13] F. S. Pamungkas and I. Kharisudin, "Analisis Sentimen dengan SVM, NAIVE BAYES dan KNN untuk Studi Tanggapan Masyarakat Indonesia Terhadap Pandemi Covid-19 pada Media Sosial Twitter," in *PRISMA, Prosiding Seminar Nasional Matematika*, 2021, pp. 628–634.

[14] D. Darwis, E. S. Pratiwi, and A. F. O. Pasaribu, "Penerapan Algoritma Svm Untuk Analisis Sentimen Pada Data Twitter Komisi Pemberantasan Korupsi Republik Indonesia," *Jurnal Ilmiah Edutic: Pendidikan dan Informatika*, vol. 7, no. 1, pp. 1–11, 2020.

[15] M. C. Mihaescu and P. S. Popescu, "Review on publicly available datasets for educational data mining," *Wiley Interdiscip Rev Data Min Knowl Discov*, vol. 11, no. 3, p. e1403, 2021.

[16] A. Izzah and R. Widyastuti, "Prediksi Harga Saham Menggunakan Improved Multiple Linear Regression untuk Pencegahan Data Outlier," *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, pp. 141–150, 2017.

**Building of Informatics, Technology and Science (BITS)**
Volume 6, No 2, September 2024, Page: 1159−1168
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i2.5543

[17] W. A. Arifin, I. Ariawan, A. A. Rosalia, L. Lukman, and N. Tufailah, "Data scaling performance on various machine learning algorithms to identify abalone sex," *Jurnal Teknologi dan Sistem Komputer*, vol. 10, no. 1, pp. 26–31, 2022.
[18] M. Tangkelangi, S. W. Djami, and A. Rantesalu, "Pemeriksaan Kadar Total Protein dan Albumin Sebelum dan Sesudah Pemberian Makanan Tambahan Pada Balita Stunting di Kelurahan Penfui, Kota Kupang," *Jurnal Nusantara Berbakti*, vol. 1, no. 4, pp. 116–121, 2023.
[19] V. N. M. Kusman, V. Metayani, and O. Karnalim, "Prediksi Analisis Sentimen Data Debat Pemilihan Presiden 2024 Menggunakan Support Vector Machine (SVM)," *Explore IT: Jurnal Keilmuan dan Aplikasi Teknik Informatika*, vol. 16, no. 1, pp. 1–5, 2024.
[20] B. Rakajati and E. Y. Hidayat, "Perbandingan Metode Naïve Bayes dan Support Vector Machine Pada Klasifikasi 22 Bahasa Daerah," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 8, no. 1, pp. 221–230, 2024.