

Implementasi Algoritma *Support Vector Machine* Untuk Klasifikasi Status *Stunting* Pada Balita

Abdul Jalil^{1✉}, Ahmad Homaidi², Zaehol Fatah³

^{1, 2, 3} Prodi Teknologi Informasi, Fakultas Sains & Teknologi, Universitas Ibrahimy, Indonesia

Informasi Artikel	ABSTRAK
Riwayat Artikel Diserahkan : 26-06-2024 Direvisi : 08-07-2024 Diterima : 12-07-2024	Stunting dan kekerdilan balita yang disebabkan oleh kekurangan gizi jangka panjang adalah masalah kesehatan utama di Indonesia. Dalam penelitian ini, algoritma <i>Vector Support Machine</i> (SVM) digunakan untuk mengklasifikasikan status stunting balita dengan tingkat akurasi yang tinggi. Dataset sebanyak 6500 data untuk kasus stunting di Indonesia terdiri dari delapan atribut: "Jenis kelamin", "umur", "berat lahir", "berat badan", "tinggi badan", "aksi eksklusif", dan "stunting." Data ini dikumpulkan melalui website Kaggle data publik. Preprocessing termasuk perbaikan variabel data dan pembersihan suara data. Selanjutnya, dataset yang telah siap dibagi menjadi 20% data uji dan 80% data latih. Ketika digunakan untuk mengklasifikasikan kasus stunting pada balita, algoritma SVM dengan kernel "linear" menunjukkan nilai akurasi 82%, presisi 80%, dan recall 86%. Hasil ini menunjukkan bahwa algoritma SVM sangat baik untuk mengklasifikasikan kasus stunting pada balita. Selanjutnya, model klasifikasi stunting yang dibangun diterapkan pada aplikasi web yang menggunakan <i>framework Streamlit</i>.
Kata Kunci: Stunting; Data Mining; SVM; Kaggle	ABSTRACT <i>Stunting and stunted growth in toddlers caused by long-term malnutrition are major health problems in Indonesia. In this study, the Vector Support Machine (SVM) algorithm is used to classify the stunting status of toddlers with a high level of accuracy. A total of 6500 data with 8 attributes—gender, age, birth weight, body weight, height, exclusive breastfeeding, and stunting—was taken from the Kaggle website, which is open to the public. Setelah preprocessing, which involves enhancing data variables and eliminating noise, the data is prepared to be divided into 80% training and 20% test. With the use of a "linear" kernel and the SVM algorithm, classification was completed with an accuracy value of 82%, precision of 80%, and recall value of 86%. These findings show that the SVM algorithm is very good at categorizing stunting cases of toddlers. The resulting stunting classification model was then integrated into a web application using the Streamlit framework.</i>
Keywords: Stunting, Data Mining, SVM, Kaggle	
Corresponding Author: Abdul Jalil Prodi Teknologi Informasi, Fakultas Sains & Teknologi, Universitas Ibrahimy, Indonesia Jl. KHR. Syamsul Arifin No.1-2, Sukorejo, Sumberejo, Kec. Banyuputih, Kabupaten Situbondo, Jawa Timur 68374 Email: abduljalilarpal@gmail.com	

PENDAHULUAN

Hampir setiap negara di seluruh dunia, termasuk Indonesia, menghadapi masalah stunting atau balita kerdil, yang merupakan akibat dari kekurangan nutrisi bagi balita. Stunting adalah kondisi di mana balita mengalami kegagalan pertumbuhan karena kekurangan asupan gizi atau

kekurangan gizi jangka panjang. Akibatnya, balita lebih pendek dari seusianya. Data yang dikumpulkan oleh Badan Pusat Statistik menunjukkan bahwa masyarakat dan pemerintah Indonesia masih menghadapi masalah stunting sebagai tantangan yang signifikan. Stunting masih tinggi di Indonesia dan merupakan salah satu penyebab utama tingginya angka kematian dan kesakitan pada anak-anak di bawah usia lima tahun (Putri, 2020).

Istilah "stunting" digunakan untuk menggambarkan gangguan pertumbuhan fisik balita yang disebabkan oleh kekurangan gizi yang terus menerus atau kekurangan gizi yang berlangsung lama. Tidak hanya masalah fisik, stunting pada anak-anak memiliki konsekuensi yang signifikan dan luas. Stunting juga dapat memengaruhi kondisi psikologis anak, seperti kehilangan kepercayaan diri dan harga diri, menurut penelitian yang dilakukan oleh. Pada akhirnya, ini dapat mengganggu hubungan sosial anak dengan lingkungan sekitarnya. Penurunan kemampuan kognitif, gangguan perkembangan motorik, dan peningkatan risiko penyakit kronis di masa depan adalah semua efek dari stunting. Saat dewasa, anak-anak yang stunting cenderung memiliki tingkat kecerdasan yang lebih rendah, prestasi akademik yang kurang baik, dan produktivitas yang lebih rendah (Rafika & Gz, 2019).

Organisasi Kesehatan Dunia (WHO) telah menetapkan bahwa Global Nutrition Target akan fokus pada masalah stunting pada tahun 2025. Ini akan diterapkan pada program Sustainable Development Goals (SDGs) pada tahun 2030. Dengan tingkat stunting tertinggi 29,6% pada tahun 2017, Indonesia berada di posisi kelima dari 201 hingga 2030 (Purnamasari & Rahmawati, 2021). Ketidacukupan asupan nutrisi dari janin hingga bayi menyebabkan stunting. Menurut survei Status Gizi Indonesia (SSGI) tahun 2021, prevalensi stunting di Indonesia mencapai 24,4%, turun menjadi 21,6% pada tahun 2022 (Munira, 2023). Menurut penelitian dari Kesehatan Dasar (Riskades), prevalensi stunting adalah 37,2% pada tahun 2019, naik menjadi 35,6% pada tahun 2019 dan 36,8% pada tahun 2020 (Titimeidara & Hadikurniawati, 2021).

Akibatnya, sangat penting untuk menangani dan mencegah stunting pada balita. Untuk memastikan bahwa ibu hamil, ibu menyusui, dan anak balita memiliki akses yang lebih baik ke layanan kesehatan, pemerintah dan masyarakat harus bekerja sama. Masalah stunting hanya dapat diatasi secara efektif dan memberikan masa depan yang lebih cerah bagi generasi berikutnya melalui upaya yang komprehensif dan berkelanjutan. Pemeriksaan status stunting balita membutuhkan waktu yang lama karena prosedur manual yang tidak akurat dan mungkin berbahaya. Oleh karena itu, dibutuhkan sistem yang dapat mengklasifikasikan data status stunting balita yang bertumpuk (Purnamasari & Rahmawati, 2021).

Pembelajaran mesin (ML) dapat menjadi cara inovatif untuk mengklasifikasikan status stunting. Menurut (Andiani et al., 2019), tujuan utama proses klasifikasi adalah kelas label, dan tahap klasifikasi digunakan dalam proses pengkategorian untuk mengidentifikasi kelas data yang tidak memiliki label kelas yang diketahui. Proses klasifikasi dilakukan dalam dua tahap. Belajar dilakukan untuk menganalisis data pembelajaran dengan algoritma tertentu. Kemudian, klasifikasi dilakukan untuk memprediksi ketepatan klasifikasi dengan data pengujian. Pembelajaran mesin memiliki banyak algoritma untuk melakukan klasifikasi pada berbagai jenis data, seperti mengklasifikasikan stunting pada anak balita. Algoritma pembelajaran mesin memiliki kemampuan untuk membantu klasifikasi stunting dengan lebih cepat, akurat, dan efektif dibandingkan dengan metode konvensional. Salah satu algoritma klasifikasi yang paling umum dan digunakan dalam penelitian ini adalah support vector machine (SVM). Ini dipilih karena memiliki banyak keunggulan dan fitur yang relevan dengan masalah klasifikasi stunting (Salman, 2019).

Langkah berikutnya adalah menerapkan model pembelajaran mesin untuk klasifikasi stunting yang dibangun menggunakan algoritma vector support machine (SVM). Untuk menerapkan model klasifikasi stunting ke dalam aplikasi web yang dapat diakses melalui internet, penelitian ini menggunakan Streamlit, sebuah framework open-source yang populer dalam komunitas ilmu data dan pengajaran mesin. Pengguna, seperti tenaga kesehatan, peneliti, atau bahkan orang awam, dapat dengan mudah memasukkan informasi seperti jenis kelamin, usia, berat badan lahir, tinggi badan, dan riwayat pemberian ASI eksklusif. Aplikasi web akan memprediksi dan mengklasifikasikan stunting menggunakan model SVM yang telah dilatih berdasarkan data masukan.

METODE PENELITIAN

Proses yang dilalui dalam penelitian ini terdiri dari banyak tahapan yang saling terkait dan penting. Pengumpulan dataset, pra-pemrosesan (*preprocessing*) data, klasifikasi menggunakan algoritma *Support vector machine* (SVM), evaluasi kinerja model, dan penerapan model klasifikasi ke dalam aplikasi web menggunakan *framework Streamlit* adalah semua langkah dalam proses ini.

Setiap langkah dalam penelitian ini digambarkan dalam gambar yang menunjukkan alur proses yang sistematis dan terstruktur. Gambar-gambar ini memberikan gambaran yang jelas tentang proses yang dilakukan, mulai dari pengumpulan data hingga penerapan model dalam aplikasi web. Visualisasi ini membantu pembaca memahami metodologi dan pendekatan yang digunakan dalam penelitian, membuat hasil dan temuan yang diperoleh lebih mudah dipahami.



Gambar 1. Alur Penelitian

Pengumpulan Data & Preprocessing

Peneliti menggunakan sumber data terbuka, atau open source, yang dapat diakses melalui situs web Kaggle.com. Kaggle adalah platform terkemuka yang memungkinkan akses ke berbagai jenis dataset untuk penelitian, pembelajaran mesin, dan proyek data science lainnya. Dalam penelitian ini, 6.500 set data digunakan, yang mencakup informasi tentang masalah *stunting* pada balita. Sebelum kumpulan data dapat digunakan untuk membangun model klasifikasi, tahapan pra-pemrosesan data, juga dikenal sebagai *preprocessing*, sangat penting untuk dilakukan. Pra-pemrosesan data adalah langkah penting dalam data mining karena bertujuan untuk memilah dan memperbaiki kualitas data agar menghasilkan kinerja model yang optimal (Putra et al., 2024).

Tahapan ini melakukan banyak hal, seperti membersihkan data (pembersihan data), menangani nilai yang hilang (nilai yang hilang), menghapus data yang tidak konsisten atau suara, dan mengubah data jika diperlukan. Pembagian kumpulan data, atau pembagian kumpulan data menjadi dua bagian, adalah tahapan penting dalam pra-pemrosesan data. Proses ini dilakukan dengan membagi dataset secara acak. 80% data digunakan sebagai pembelajaran, dan 20% sisanya digunakan sebagai pengujian. Data pelatihan sangat penting untuk pembangunan dan pelatihan model klasifikasi (Setyawan & Fauzi, 2023).

Algoritma *Support Vector Machine*

Untuk masalah klasifikasi dan prediksi, *Support vector machine* (SVM) adalah algoritma pembelajaran mesin supervised yang sangat populer. SVM (algoritma pembelajaran mesin yang diawasi) menggunakan pendekatan berbasis kernel atau penerjemah untuk menemukan hyperplane atau bidang pemisah yang optimal antara berbagai kelas dalam dataset. Tujuan utama dari SVM adalah untuk memaksimalkan jarak atau jarak antara hyperplane dengan data terdekat dari setiap kelas, sehingga diperoleh pemisahan yang jelas dan baik antara kelas-kelas tersebut (Isnain et al., 2021). Untuk menghubungkan data dari ruang input awal ke ruang fitur yang lebih besar, kernel digunakan. Metode optimasi digunakan untuk melatih algoritma SVM untuk menemukan hyperplane ideal yang memaksimalkan margin pemisah antara kelas-kelas. Pemisahan kelas dan generalisasi model pada data baru meningkat seiring dengan margin yang diperoleh (Rozaqi et al., 2021).

Dalam proses pelatihan SVM, algoritma menggunakan data pelatihan yang sudah diberi label kelas untuk membangun model klasifikasi. Proses pelatihan ini digunakan untuk menemukan hyperplane yang optimal dan mempelajari pola dan karakteristik yang membedakan kelas-kelas dalam dataset. Setelah model terbentuk, model tersebut dapat digunakan untuk memprediksi kelas dari data pengujian, yang merupakan data baru yang belum pernah dilihat

sebelumnya. Keunggulan utama SVM adalah kemampuan mereka untuk menangani data yang kompleks dan besar. SVM dapat mentransformasi data ke dalam ruang fitur yang lebih besar, di mana pemisahan kelas menjadi lebih mudah dengan menggunakan kernel yang sesuai. Ini termasuk masalah klasifikasi dan prediksi pada berbagai jenis data, termasuk data dengan dimensi yang tinggi, seperti data teks, gambar, atau sinyal (Suryati et al., 2023).

$$f(x_d) = \sum_{i=1}^{ns} a_i y_i \vec{x}_i \vec{x}_d + b$$

Keterangan:

ns = Jumlah support vector

a_i = Nilai bobot setiap titik data

y_i = Data kelas

$\vec{x}_i$ = Variabel *support vector*

$\vec{x}_d$ = Daya yang akan diklasifikasi

b = Nilai bias atau error

Evaluasi Model

Dalam proses mengevaluasi kinerja model klasifikasi, beberapa metrik penting digunakan untuk menentukan efektivitas model. Akurasi, presisi, dan ingat adalah metrik. Untuk menghitung nilai-nilai metrik ini, matriks yang disebut matriks kebingungan atau confusion digunakan. Matrik kekacauan sangat bermanfaat untuk evaluasi masalah klasifikasi, terutama dalam kasus biner seperti klasifikasi *stunting* yang dibahas dalam penelitian ini. Dengan membandingkan hasil prediksi model dengan label kelas yang sebenarnya, atau kebenaran tanah, matriks ini menyajikan informasi tentang jumlah prediksi yang benar dan salah dari model (Fadli & Saputra, 2023).

Secara khusus, matriks confusion terdiri dari empat kuadran yang menunjukkan empat hasil potensial: nilai positif asli (TP), nilai positif palsu (FP), nilai negatif asli (TN), dan nilai negatif palsu (FN). Nilai positif asli (TP) menunjukkan jumlah kasus di mana model benar-benar memprediksi penurunan kelas. Nilai matrix confusion dapat digunakan untuk menghitung metrik evaluasi seperti akurasi, presisi, dan *recall*. *Recall* menunjukkan seberapa banyak kasus positif yang berhasil diidentifikasi oleh model. Sementara itu, akurasi menunjukkan seberapa banyak prediksi yang benar secara keseluruhan untuk kelas negatif dan positif (Harun & Ananda, 2021).

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

a. Accuracy

Accuracy adalah perbandingan antara jumlah prediksi yang benar dan total jumlah data yang diprediksi. Skor ini akan menunjukkan seberapa baik model melakukan prediksi secara keseluruhan.

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

b. Recall

Recall menghitung jumlah besar data positif yang berhasil teridentifikasi pada model. Ini menunjukkan seberapa baik model dapat menemukan kelas tertentu.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

c. Precision

Precision dapat menunjukkan seberapa akurat model dalam menentukan kelas. Ini diukur dengan banyaknya prediksi positif yang benar dari semua prediksi positif yang dilakukan.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (4)$$

Framework Streamlit

Setelah model klasifikasi *stunting* berhasil dibangun dan dievaluasi kinerjanya menggunakan metrik seperti akurasi, presisi, dan *recall*, langkah berikutnya adalah mengimplementasikan model ke dalam sebuah aplikasi web untuk membuatnya lebih mudah diakses dan digunakan oleh berbagai pihak. Dalam penelitian ini, kami menggunakan *Framework* Open-Source *Streamlit*, yang sangat populer dalam komunitas data science dan machine learning. *Framework* ini memungkinkan pengembang untuk mengubah skrip *Python* menjadi aplikasi web interaktif dengan antarmuka web. *Framework* ini sangat cocok untuk penelitian ini karena dimaksudkan untuk membantu membangun aplikasi data mining yang melibatkan analisis data science dan sistem pembelajaran mesin lainnya (Sarangpure et al., 2023).

Kemudahan membuat aplikasi web tanpa memahami teknologi web secara mendalam adalah salah satu keunggulan utama *Streamlit*. Pengembang dapat berkonsentrasi pada logika bisnis dan implementasi model pembelajaran mesin dengan *Streamlit*, karena *framework* ini secara otomatis menangani elemen antarmuka web. Selain itu, fitur-fitur yang sangat bermanfaat untuk aplikasi data mining seperti visualisasi data interaktif dan pemetaan interaktif, serta integrasi dengan berbagai pustaka populer dalam ekosistem *Python*, seperti *scikit-learn*, *pandas*, dan *NumPy*, memungkinkan pengembang untuk menyajikan hasil analisis data dan prediksi model machine learning dalam bentuk yang lebih interaktif dan mudah dipahami bagi pengguna (Syafarina & Zaenuddin, 2023).

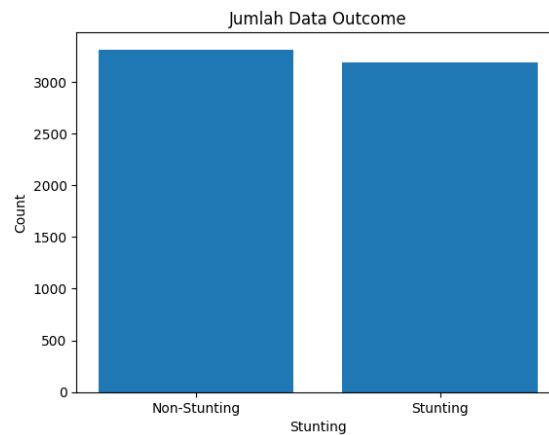
Peneliti dapat mengimplementasikan model klasifikasi *stunting* yang telah dibangun ke dalam aplikasi web yang interaktif dan mudah digunakan dengan *Streamlit*. Aplikasi dapat memprediksi apakah seorang balita mengalami *stunting* berdasarkan atribut yang dimasukkan, seperti jenis kelamin, usia, dan berat badan lahir. Implementasi model ke dalam aplikasi web ini meningkatkan aksesibilitas dan manfaat dari penelitian yang telah dilakukan, sehingga lebih banyak pihak yang berkepentingan, seperti peneliti, tenaga kesehatan, dan masyarakat umum, dapat memanfaatkannya (Syahbani & Ramadhan, 2023).

HASIL DAN PEMBAHASAN

Persiapan Dataset & Preprocessing

Platform Kaggle.com menyediakan berbagai sumber data publik untuk data penelitian ini. Dengan menyediakan akses terbuka terhadap berbagai jenis dataset untuk digunakan dalam proyek penelitian dan kompetisi, Kaggle adalah komunitas online terkemuka untuk para praktisi data science. Penelitian ini menggunakan 6.500 daftar data, yang terdiri dari beberapa variabel penting yang berkaitan dengan subjek penelitian. Dari 6.500 rekaman data, 3.250 merupakan kasus *stunting* positif dan 3.250 lainnya merupakan kasus *stunting* negatif atau non-*stunting*. Informasi yang dimasukkan ke dalam variabel-variabel ini mencakup informasi seperti karakteristik demografi, kondisi kesehatan, dan variabel lain yang dapat memengaruhi kejadian *stunting* pada anak-anak. Dataset terbuka seperti ini memungkinkan para peneliti untuk mengeksplorasi dan menganalisis lebih banyak data. Peneliti dapat memastikan kualitas dan integritas data yang digunakan dalam penelitian dengan menggunakan sumber data yang kredibel

dan terpercaya seperti Kaggle. Ini membuat hasil analisis dan kesimpulan yang diperoleh lebih akurat dan terpercaya.



Gambar 2. Data Status Stunting

Dataset yang digunakan dalam penelitian ini terdiri dari 6.500 rekaman data, masing-masing dengan 8 variabel atau kolom yang menunjukkan atribut atau fitur yang terkait dengan status *stunting* pada individu yang menderitanya. Variabel-variabel ini memiliki informasi penting yang digunakan sebagai penentu dalam menentukan dan mengklasifikasikan *stunting*.

Adanya variabel yang beragam dan lengkap memungkinkan peneliti untuk melakukan analisis yang mendalam untuk menemukan pola-pola dan hubungan antara berbagai faktor dengan kejadian *stunting*. Hal ini memungkinkan peneliti untuk memperoleh pemahaman yang lebih baik tentang faktor risiko dan penyebab *stunting*, serta mengembangkan model prediksi atau klasifikasi yang akurat untuk mendeteksi kasus *stunting* di masa mendatang.

Tabel 1. Head Data Stunting

Sex	Age	Berat Lahir	Tinggi Lahir	Berat Badan	Tinggi Badan	ASI Eksklusif	Stunting
0	56	2,9	50	11	90	1	No
0	20	3,3	49	11,1	80,5	0	No
1	4	2,8	48	6,5	63	0	No
0	14	2	49	7	71	1	No

Keterangan:

Sex = Jenis Kelamin

Age = Umur (bulan)

Birth_weight = Berat Lahir

Birth_length = Tinggi Lahir

Body_weight = Berat Badan

Body_length = Tinggi Badan

Asi_Eksklusif = ASI Langsung

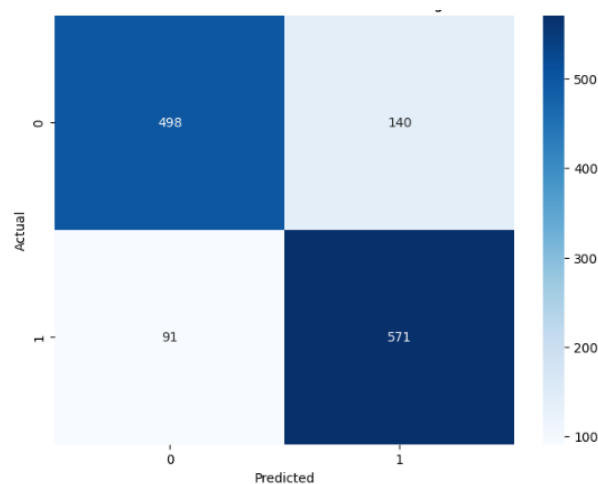
Sebelum dilakukan proses pembelajaran mesin (machine learning), data mentah yang diperoleh harus melalui beberapa tahapan pra-pemrosesan untuk memastikan kualitas dan kesiapan data. Salah satu tahapan penting adalah penanganan noise, yaitu mengatasi masalah adanya nilai-nilai yang hilang atau tidak lengkap dalam dataset. Rekaman data yang mengandung noise tersebut ditangani dengan melakukan pembersihan (cleaning) dan imputasi nilai yang hilang menggunakan metode yang sesuai. Selanjutnya, untuk mempersiapkan data bagi algoritma pembelajaran mesin, peneliti memberikan label "*stunting*" pada setiap rekaman data. Label ini menjadi target atau kelas yang ingin diprediksi oleh model, yaitu apakah seorang individu

mengalami *stunting* atau tidak berdasarkan karakteristik yang diwakili oleh variabel-variabel dalam dataset.

Setelah proses pelabelan dan pembersihan selesai, dataset dibagi menjadi dua bagian: data pelatihan dan data pengujian. Fungsi "train_test_split" digunakan untuk melakukan proses pembagian ini, dengan parameter "test_size=0.2" dan "stratify=Y". Parameter "test_size=0.2" menunjukkan bahwa 20% dari total data akan digunakan sebagai data pemeriksaan, dan 80% sisanya akan digunakan sebagai data pelatihan. Parameter "stratify=Y" memastikan bahwa distribusi kelas (penurunan dan non-penurunan) pada data penilaian dan Proses pembagian ini menghasilkan 5.200 rekaman data, yang merupakan 80% dari 6.500 data, yang akan digunakan sebagai data pelatihan untuk melatih model algoritma Support vector machine (SVM). Sementara itu, 1.300 rekaman data, yang merupakan 20% dari total data, disisihkan sebagai data pengujian, yang akan digunakan untuk mengevaluasi kinerja model yang telah dilatih.

Klasifikasi Algoritma *Support Vector Machine*

Setelah proses pra-pemrosesan selesai setelah dataset dikumpulkan, algoritma vector support machine (SVM) digunakan untuk melakukan klasifikasi. Ini dilakukan menggunakan bahasa pemrograman Python dan lingkungan Jupyter Notebook. Peneliti memanfaatkan pustaka atau bibliotek seperti scikit-learn, numpy, dan pandas yang tersedia dalam ekosistem Python untuk menerapkan SVM ini. Nilai kernel adalah fungsi yang digunakan untuk memetakan data dari ruang input (ruang fitur) ke ruang fitur yang lebih besar. Tujuan penerapan kernel linear dengan parameter "kernel = linear" dalam penelitian ini adalah untuk menemukan bidang pemisah atau hyperplane terbaik dalam ruang fitur awal tanpa perlu mengubahnya ke ruang fitur yang lebih kompleks. Peneliti dapat menggunakan matrix confusion untuk menghitung metrik evaluasi seperti akurasi, presisi, dan recall untuk menilai kinerja model secara keseluruhan. Informasi ini sangat penting untuk menginterpretasikan hasil klasifikasi dan menentukan apakah model SVM dengan kernel linear yang dibangun cukup baik atau apakah penyesuaian tambahan diperlukan.



Gambar 3. Nilai *Confusion matrix* Algoritma SVM

Setelah algoritma *Support vector machine* (SVM) menghasilkan *confusion matrix* dari hasil klasifikasi, langkah selanjutnya adalah melakukan perhitungan untuk metrik akurasi, presisi, dan *recall*, yang merupakan metrik evaluasi yang sangat penting. Ketiga metrik ini memberikan data yang lebih mendalam tentang bagaimana model berhasil mengklasifikasikan kasus *stunting* dan *non-stunting* dengan benar. Tabel menunjukkan perhitungan akurasi, presisi, dan *recall* model SVM dengan kernel linear dalam penelitian ini.

Tabel tersebut menunjukkan nilai-nilai metrik tersebut untuk kedua kelas *stunting* dan *non-stunting*. Tabel ini memungkinkan peneliti untuk dengan mudah membandingkan kinerja model saat mengklasifikasikan setiap kelas dan menemukan kelebihan dan kekurangan model. Tabel ini

juga memungkinkan peneliti untuk melakukan analisis lebih lanjut tentang keuntungan dari presisi dan *recall* serta membuat keputusan apakah model perlu diubah untuk meningkatkan metrik tertentu atau untuk mencapai keseimbangan yang ideal antara keduanya, sesuai dengan tujuan dan konteks penelitian.

Dengan informasi tentang *confusion matrix*, akurasi, presisi, dan *recall* peneliti dapat mengevaluasi kinerja model SVM dengan kernel linear secara menyeluruh dan membuat kesimpulan yang tepat tentang keefektifan model dalam mengklasifikasikan kasus *stunting* pada dataset yang digunakan.

$$\text{Akurasi} = \frac{498+571}{498+571+91+140} = 89\%$$

$$\text{Presisi} = \frac{571}{571+140} = 80\%$$

$$\text{Recall} = \frac{571}{571+91} = 86\%$$

Klasifikasi *Stunting* menggunakan *Streamlit*

Setelah model klasifikasi *stunting* berhasil dibangun dan diuji kinerjanya menggunakan algoritma *Support vector machine* (SVM) dengan kernel linear, langkah selanjutnya adalah mengimplementasikan model tersebut dalam bentuk aplikasi web sehingga lebih banyak orang dapat menggunakannya. Untuk mencapai tujuan ini, peneliti menggunakan modul *pickle* yang tersedia dalam bahasa pemrograman *Python*. Modul ini adalah komponen bawaan *Python* yang memungkinkan untuk serialisasi dan deserialisasi objek *Python*, termasuk model pembelajaran mesin yang telah dilatih. Model SVM yang telah dibangun dapat disimpan dalam format biner yang efisien dan mudah diakses kembali saat dibutuhkan dengan menggunakan *pickle*.



Gambar 4. Tampilan Klasifikasi *Stunting* berbasis web

Peneliti kemudian menggunakan *Streamlit*, sebuah *framework* open-source *Python* yang memungkinkan pembuatan aplikasi web yang cepat dan mudah, setelah model disimpan dalam format *pickle*. Selain menyediakan antarmuka pemrograman yang mudah digunakan, *Streamlit* memungkinkan peneliti untuk mengintegrasikan model pembelajaran mesin yang telah dibangun dengan antarmuka pengguna yang interaktif. Pengguna hanya perlu memasukkan perintah "*streamlit run stunting.py*" pada prompt atau terminal untuk menjalankan aplikasi web ini. Perintah ini akan memuat dan menjalankan file *Python* "*stunting.py*", yang berisi kode antarmuka web dan memanggil model SVM yang telah disimpan dalam format *pickle*." Pengguna diminta untuk memasukkan nilai-nilai dari sejumlah variabel atau fitur yang terkait dengan masalah *stunting* dalam aplikasi web yang dibangun untuk mengimplementasikan model klasifikasi *stunting*. Variabel-variabel ini adalah faktor-faktor yang dipertimbangkan oleh model saat memprediksi dan mengklasifikasikan apakah seorang balita menderita *stunting* atau tidak. Pengguna pertama kali

harus memasukkan variabel jenis kelamin (sex) balita, yang biasanya diwakili dengan nilai biner 0 atau 1.

Selanjutnya, pengguna harus memasukkan umur (umur) balita dalam satuan bulan. Selain itu, untuk model diperlukan data tentang panjang badan saat lahir (*Birth_Length*) dan berat badan saat lahir (*Birth_Weight*). Selain itu, pengguna juga diminta untuk memasukkan data tentang berat badan (*Body_Weight*) dan tinggi badan (*Body_Length*) balita saat ini. Variabel-variabel ini menunjukkan pertumbuhan dan perkembangan balita dari lahir hingga saat ini. Terakhir, sebagai bagian penting dari pencegahan *stunting*, pengguna diminta untuk memberikan informasi tentang riwayat pemberian ASI eksklusif (ASI_Eksklusif) kepada balita. Biasanya, variabel ini dikodekan dalam bentuk biner, dengan nilai 0 menunjukkan bahwa balita tidak mendapatkan ASI eksklusif, dan nilai 1 menunjukkan bahwa balita mendapatkan ASI eksklusif.

KESIMPULAN DAN SARAN

Kesimpulan

Studi ini mencoba mengklasifikasikan status *stunting* anak balita dengan menggunakan algoritma *Support vector machine* (SVM). Dataset yang digunakan untuk penelitian ini berasal dari platform open source Kaggle.com dan terdiri dari 6.500 rekaman data yang mencakup delapan variabel atau fitur yang terkait dengan masalah *stunting*. Dari 6.500 rekaman data tersebut, 3.250 menunjukkan kasus *stunting* positif, sedangkan 3.250 lainnya menunjukkan kasus *stunting* negatif atau non-*stunting*. Sebelum digunakan untuk membangun model klasifikasi, dataset terlebih dahulu dibagi menjadi dua bagian: data pelatihan dan data uji. Sebagian besar data, atau 5.200 rekaman data, digunakan sebagai data pelatihan untuk melatih model SVM. Sisa 20%, atau 1.300 rekaman data, digunakan sebagai data pengujian untuk mengevaluasi kinerja model yang telah dilatih.

Peneliti menggunakan algoritma *switching multiple variables* (SVM) dengan kernel linear untuk membuat model klasifikasi *stunting*. Tujuan dari pemilihan kernel linear ini adalah untuk menemukan bidang pemisah atau *hyperplane* terbaik dalam ruang fitur awal tanpa perlu mengubah ke ruang fitur yang lebih kompleks. Hasil evaluasi menunjukkan bahwa model klasifikasi *stunting* yang dibangun dengan algoritma SVM dengan kernel linear memiliki kinerja yang sangat baik. Selain memiliki akurasi yang tinggi sebesar 82%, model ini juga memiliki nilai presisi sebesar 80%, yang menunjukkan bahwa dari kasus yang diprediksi sebagai *stunting* oleh model adalah benar positif, dan nilai *recall* model adalah 86%, yang menunjukkan bahwa klasifikasi memiliki nilai yang bagus.

Saran

Saran yang dapat dipertimbangkan antara lain Pertama, eksplorasi algoritma lain atau optimasi parameter pada algoritma SVM dapat dilakukan untuk meningkatkan akurasi model klasifikasi *stunting*. Kedua, penambahan fitur atau variabel yang relevan seperti data riwayat kesehatan ibu, pola makan, dan kondisi lingkungan dapat dipertimbangkan untuk memperkaya model dan meningkatkan kinerjanya.

REFERENSI

- Andiani, L., Ilmu Komputer, J., & Palupi Rini, D. (2019). Analisis Penyakit Jantung Menggunakan Metode KNN Dan Random Forest. In *Prosiding Annual Research Seminar* (Vol. 5, Issue 1).
- Fadli, M., & Saputra, R. A. (2023). Klasifikasi Dan Evaluasi Performa Model Random Forest Untuk Prediksi Stroke. *Jurnal Teknik*, 12(2).
- Harun, A., & Ananda, D. P. (2021). Analisa Sentimen Opini Publik Tentang Vaksinasi Covid-19 di Indonesia Menggunakan *Naïve bayes* dan *Decission Tree*: *Analysis of Public Opinion Sentiment About Covid-19 Vaccination in Indonesia Using Naïve Bayes and Decission Tree*. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 1(1), 58–64.

- Isnain, A. R., Sakti, A. I., Alita, D., & Marga, N. S. (2021). Sentimen Analisis Publik Terhadap Kebijakan Lockdown Pemerintah Jakarta Menggunakan Algoritma Svm. *Jurnal Data Mining Dan Sistem Informasi*, 2(1), 31–37.
- Munira, S. L. (2023). Hasil Survei Status Gizi Indonesia (SSGI) 2022. *Jakarta: Badan Kebijakan Pembangunan Kesehatan Kementerian Kesehatan RI*.
- Purnamasari, M., & Rahmawati, T. (2021). Hubungan Pemberian Asi Eksklusif dengan Kejadian Stunting Pada Balita Umur 24-59 Bulan. *Jurnal Ilmiah Kesehatan Sandi Husada*, 10(1), 290–299. <https://doi.org/10.35816/jiskh.v10i1.490>
- Putra, F., Tahiyat, H. F., Ihsan, R. M., Rahmaddeni, R., & Efrizoni, L. (2024). Penerapan Algoritma K-Nearest Neighbor Menggunakan Wrapper Sebagai Preprocessing untuk Penentuan Keterangan Berat Badan Manusia: *Application of K-Nearest Neighbor Algorithm Using Wrapper as Preprocessing for Determination of Human Weight Information*. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 4(1), 273–281.
- Putri, O. N. (2020). Implementasi Metode CNN Dalam Klasifikasi Gambar Jamur Pada Analisis Image Processing (Studi Kasus: Gambar Jamur Dengan Genus Agaricus Dan Amanita).
- Rafika, M., & Gz, S. (2019). Dampak Stunting Pada Kondisi Psikologis Anak. *Buletin Jagaddhita*, 1(1).
- Rozaqi, A. J., Sunyoto, A., & rudyanto Arief, M. (2021). Deteksi Penyakit Pada Daun Kentang Menggunakan Pengolahan Citra dengan Metode Convolutional Neural Network. *Creative Information Technology Journal*, 8(1), 22–31.
- Salman, N. (2019). Algoritma k-Nearest Neighbor Berbasis Backward Elimination Pada Client Telemarketing. *SISITI: Seminar Ilmiah Sistem Informasi Dan Teknologi Informasi*, 8(2), 141–150.
- Sarangpure, N., Dhamde, V., Roge, A., Doye, J., Patle, S., & Tamboli, S. (2023). Automating the machine learning process using PyCaret and Streamlit. *2023 2nd International Conference for Innovation in Technology (INOCON)*, 1–5.
- Setyawan, D. A., & Fauzi, D. N. (2023). Implementasi Fungsi Dispersion Ratio Pada Proses Splitting Atribut Algoritma Decision Tree. *Madani: Jurnal Ilmiah Multidisiplin*, 1(1).
- Suryati, E., Styawati, S., & Aldino, A. A. (2023). Analisis Sentimen Transportasi Online Menggunakan Ekstraksi Fitur Model Word2vec Text Embedding Dan Algoritma Support Vector Machine (SVM). *Jurnal Teknologi Dan Sistem Informasi*, 4(1), 96–106.
- Syafarina, G. A., & Zaenuddin, Z. (2023). Implementasi Framework Streamlit Sebagai Prediksi Harga Jual Rumah Dengan Linear Regresi. *METIK JURNAL*, 7(2), 121–125. <https://doi.org/10.47002/metik.v7i2.608>
- Syahnani, M. F. N., & Ramadhan, N. G. (2023). Klasifikasi Gerakan Yoga dengan Model Convolutional Neural Network Menggunakan Framework Streamlit. *Jurnal Media Informatika Budidarma*, 7(1), 509–519.
- Titimeidara, M. Y., & Hadikurniawati, W. (2021). Implementasi Metode Naive Bayes Classifier Untuk Klasifikasi Status Gizi Stunting Pada Balita. *Jurnal Ilmiah Informatika*, 9(01), 54–59.