

Penerapan Support Vector Machine untuk Klasifikasi Tingkat Risiko Kebakaran Hutan

I Komang Galih Agustan^{a1}, I Gede Santi Astawa^{a2}

Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Udayana
Jalan Raya Kampus Udayana, Bukit Jimbaran, Kuta Selatan, Badung, Bali, Indonesia
¹galihagustan2@email.com
²santi.astawa@unud.ac.id

Abstract

Classifying forest fire risk levels is a critical step for disaster mitigation, yet it poses significant challenges due to data complexity and class imbalance. This study systematically applies and evaluates the performance of the Support Vector Machine (SVM) algorithm for the multi-class classification of fire risk (Low, Medium, High) using the standard UCI Forest Fires dataset. The methodology involved a comprehensive preprocessing imbalance and hyperparameter optimization of C and gamma using GridSearchCV with cross-validation. Experimental results show that the final, optimized SVM model only achieved an accuracy of 50% and a macro-average F1-Score of 40% on the test set. This limited performance, particularly the model's failure to reliably identify the 'High' risk class, indicates that the standard meteorological features within the dataset possess insufficient predictive power for the complex task of classifying fire severity, highlighting that model success is fundamentally dependent on feature richness over algorithmic optimization.

Keywords: Fire Forest, Support Vector Machine, Unbalanced Data, SMOTE, Classification, Hyperparameter Optimization.

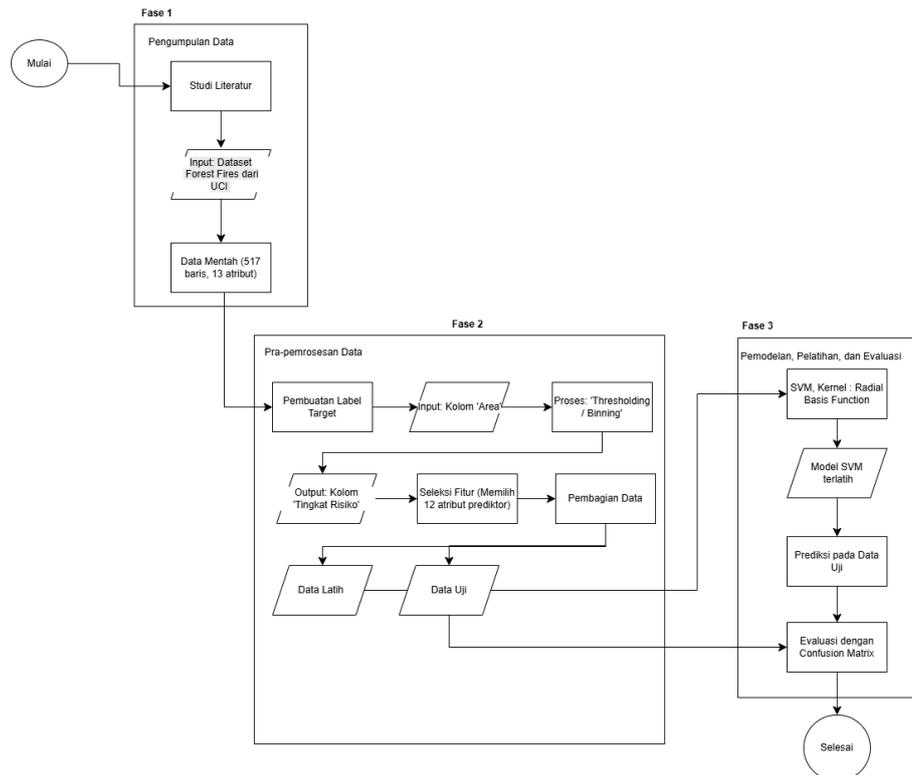
1. Pendahuluan

Kebakaran hutan dan lahan (karhutla) merupakan salah satu bencana yang menjadi ancaman nyata dan sering terjadi di Indonesia. Dampak yang ditimbulkan bersifat multidimensi, tidak hanya mengganggu keseimbangan lingkungan melalui polusi kabut asap dan perusakan ekosistem, tetapi juga meluas ke sektor kesehatan masyarakat dan menimbulkan kerugian ekonomi yang signifikan. Studi kasus di berbagai daerah rawan seperti Kabupaten Pulang Pisau dan Kabupaten Kapuas di Kalimantan Tengah menunjukkan bahwa karhutla adalah masalah nyata yang terjadi secara berulang dengan dampak kerusakan yang luas di tingkat lapangan. Mengingat besarnya skala masalah ini, pengembangan sebuah sistem peringatan dini yang efektif untuk mencegah kebakaran menjadi suatu kebutuhan yang sangat penting.[1]. Upaya untuk mengklasifikasi risiko kebakaran seringkali dihadapkan pada tantangan inkonsistensi dan subjektivitas jika hanya mengandalkan penilaian manual. Penilaian kualitas atau risiko secara manual dapat bersifat subjektif dan tidak konsisten informasinya. Oleh karena itu, diperlukan sebuah pendekatan yang lebih objektif dan konsisten berbasis data dan teknologi. Pemanfaatan *machine learning* menawarkan solusi untuk menganalisis berbagai faktor pemicu kebakaran secara algoritmik, sehingga dapat menghasilkan klasifikasi risiko yang lebih baik dan dapat dipertanggungjawabkan secara ilmiah. Salah satu algoritma *machine learning* yang telah terbukti baik dalam tugas klasifikasi adalah *Support Vector Machine* (SVM). Kemampuan SVM dalam menemukan *hyperplane* optimal untuk memisahkan kelas data membuatnya menjadi pilihan yang populer di banyak kajian ilmiah. Sebuah penelitian berhasil menggunakan SVM untuk memprediksi potensi kebakaran di lahan gambut Indonesia berdasarkan data cuaca dan Indeks Cuaca Kebakaran (FWI), dan mencapai akurasi (tinggi akan diubah). Selain itu, berbagai pendekatan komputasional lain seperti *clustering* menggunakan K-Means dan penentuan *thresholding* berbasis fungsi kepadatan probabilitas juga telah digunakan untuk menganalisis dan mengelompokkan data

kebakaran.[2]

Meskipun berbagai pendekatan telah dieksplorasi, penerapan SVM secara langsung untuk mengklasifikasikan tingkat risiko kebakaran yang bersifat multi-kelas menggunakan data tabular yang berisi atribut meteorologi dan FWI masih menjadi area yang akan diteliti lebih lanjut. Oleh karena itu, penelitian ini bertujuan untuk menerapkan SVM untuk mengklasifikasikan tingkat risiko kebakaran hutan. Klasifikasi ini akan didasarkan pada dataset *Forest Fires* (Cortez and Morais, 2007) dari UCI Machine Learning Repository, sebuah repositori yang umum digunakan untuk penelitian machine learning.

2. Metode Penelitian



Gambar 1. Diagram Alur Penelitian

Alur penelitian ini disusun secara sistematis untuk menguji penerapan SVM, sebagaimana diilustrasikan pada Gambar 1. Proses diawali dengan tahap pengumpulan data, yaitu menggunakan dataset *Forest Fires* dari *UCI Machine Learning Repository*. Dataset tersebut kemudian melewati tahap pra-pemrosesan, di mana label target klasifikasi dibuat dari atribut 'area' melalui metode *thresholding*, sebelum akhirnya dibagi menjadi 80% data latih dan 20% data uji. Selanjutnya, data latih digunakan untuk membangun model dengan melatih algoritma SVM yang menggunakan kernel *Radial Basis Function* (RBF). Kinerja model yang telah terlatih kemudian diuji dan dievaluasi pada data uji dengan menghitung metrik akurasi dan menganalisis *confusion matrix* untuk menarik kesimpulan akhir.

2.1. Pengumpulan Data

Metode pengumpulan data yang digunakan dalam penelitian ini terdiri dari dua tahap utama, yaitu studi literatur untuk mengumpulkan landasan teori dan pengetahuan, serta penggunaan dataset sekunder untuk implementasi dan pengujian model.

a. Studi Literatur

Studi literatur dilakukan untuk mengumpulkan informasi, teori, dan hasil penelitian terdahulu yang berkaitan dengan topik penelitian. Tahap ini bertujuan untuk memahami permasalahan, mengeksplorasi metode metode yang pernah digunakan, dan memperkuat pemilihan algoritma *Support Vector Machine* (SVM) sebagai metode utama. Skala dan dampak dari kebakaran hutan dan lahan di Indonesia melalui laporan dan studi kasus spesifik di berbagai daerah seperti Kabupaten Pulang Pisau dan Kapuas, Kalimantan Tengah penting dalam memahami domain masalah. Peneliti merujuk kepada penelitian-penelitian sebelumnya yang juga menggunakan *machine learning* untuk menganalisis kebakaran hutan dan lahan. Beberapa pendekatan yang diidentifikasi antara lain adalah analisis *clustering* menggunakan K-means untuk menemukan pola, serta klasifikasi tingkat keparahan kebakaran menggunakan metode *thresholding* berbasis fungsi kepadatan probabilitas. Berdasarkan studi literatur yang komprehensif, peneliti memilih SVM karena keterkaitannya dengan jenis data dan tujuan penelitian, serta telah terbukti memiliki kinerja yang baik.[3]

b. Dataset Penelitian

Penelitian ini menggunakan data sekunder dalam bentuk tabular, yaitu dataset *Forest Fires* yang diperoleh dari UCI Machine Learning Repository. Dataset ini merupakan dataset yang umum digunakan dalam penelitian yang berkaitan dengan prediksi kebakaran hutan. Secara keseluruhan, dataset ini memuat 517 sampel kejadian kebakaran yang terjadi di wilayah timur laut Portugal, khususnya di Taman Nasional Montesinho. Dataset ini terdiri dari 13 atribut yang relevan untuk analisis risiko kebakaran. Atribut tersebut meliputi koordinat spasial (X dan Y) yang merepresentasikan posisi geografis pada peta t aman, atribut temporal seperti bulan (month) dan hari (day) terjadinya kebakaran, serta empat komponen sistem Fire Weather Index (FWI) yaitu FFMC, DMC, DC, dan ISI yang digunakan untuk mengukur tingkat risiko kebakaran. Selain itu, terdapat atribut meteorologi seperti temperatur (suhu dalam Celcius), RH (kelembapan relatif dalam persen), angin (kecepatan angin dalam km/jam), dan curah hujan (curah hujan dalam mm/m²). Atribut area yang menunjukkan total luas hutan yang terbakar (dalam hektar) menjadi target label dalam dataset ini. Dalam penelitian ini, atribut spasial, temporal, FWI, dan meteorologi akan digunakan sebagai fitur input dalam model *Support Vector Machine* (SVM). sementara itu, kolom area akan diproses lebih lanjut untuk diubah menjadi label klasifikasi berupa tiga kategori tingkat kebakaran: 'rendah', 'sedang', dan 'tinggi', yang akan dijelaskan lebih rinci di bagian pra-pemrosesan data.

2.2. Pra-pemrosesan Data

Pra-pemrosesan data adalah tahap krusial yang bertujuan untuk mentransformasikan data mentah menjadi format yang lebih bersih untuk dilatih oleh *machine learning*. Keberhasilan dari tahap ini akan memengaruhi kinerja akhir dari model klasifikasi. Dalam penelitian ini, ada dua proses utama yang dilakukan, yaitu kuantisasi label target dan pembagian data.

a. Kuantisasi Label Target

Atribut target asli dalam dataset, yakni area, bersifat numerik kontinu yang merepresentasikan luas hutan yang terbakar dalam satuan hektar. Namun, karena tujuan dari penelitian ini adalah membangun model klasifikasi, atribut tersebut perlu diubah menjadi kelas-kelas kategorikal. Proses transformasi ini dilakukan melalui teknik kuantisasi, yaitu metode yang membagi variabel kontinu menjadi sejumlah interval. Dalam penelitian ini, kuantisasi dilakukan menggunakan pendekatan statistik berbasis kuantil, yang memungkinkan pembentukan tiga kelas risiko dengan distribusi data yang relatif seimbang. Adapun aturan pembagiannya adalah sebagai berikut: kelas 'Rendah' mencakup semua data kejadian kebakaran yang berada dalam kuantil pertama, kelas 'Sedang' mencakup data yang berada antara kuantil pertama dan kuantil kedua, sedangkan kelas 'Tinggi' mencakup kuantil ketiga. Pendekatan ini dipilih untuk memastikan bahwa setiap kelas memiliki jumlah sampel yang representatif untuk mendukung *processing model* dan mendefinisikan tingkat risiko kebakaran secara objektif berdasarkan sebaran statistik yang ada di dalam dataset.

b. Penanganan Data Tidak Seimbang dengan SMOTE

Ketidakseimbangan kelas (*class imbalance*) adalah kondisi di mana distribusi sampel antar kelas dalam dataset sangat tidak merata. Berdasarkan analisis awal, dataset yang digunakan dalam penelitian ini menunjukkan ketidakseimbangan yang cukup mencolok, dengan kelas 'rendah' memiliki jumlah sampel yang jauh lebih banyak dibandingkan kelas 'sedang' dan 'tinggi'. Kondisi ini dapat menyebabkan model machine learning menjadi condong ke kelas mayoritas, sehingga performanya pada kelas minoritas menjadi kurang optimal. Untuk mengatasi permasalahan tersebut, penelitian ini memanfaatkan teknik oversampling yang dikenal sebagai SMOTE (Synthetic Minority Oversampling Technique). Berbeda dari pendekatan oversampling konvensional yang hanya menyalin ulang data, SMOTE menggunakan metode yang lebih inovatif untuk meningkatkan representasi kelas minoritas. Secara konsep, SMOTE bekerja dengan memilih sampel acak dari kelas minoritas, mengidentifikasi beberapa tetangga terdekatnya dalam kelas yang sama, lalu menghasilkan sampel sintesis baru sepanjang garis imajiner yang menghubungkan sampel asli dengan tetangga tersebut. Dengan demikian, jumlah data untuk kelas 'sedang' dan 'tinggi' dapat ditingkatkan secara efektif, memungkinkan model memahami karakteristik kelas minoritas lebih baik dan menghasilkan dataset latihan yang lebih seimbang. Sesuai dengan praktik terbaik, teknik SMOTE ini diterapkan hanya pada data latihan setelah proses pembagian data untuk mencegah terjadinya kebocoran data [4]

c. Pembagian Data

Setelah semua data memiliki label, dataset dibagi menjadi dua bagian terpisah secara acak, yakni data latihan dan data uji. Proporsi pembagian yang digunakan dalam penelitian ini adalah 80% untuk data latihan dan 20% untuk data uji

2.3. Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah salah satu algoritma *supervised learning* yang sangat andal dan populer untuk masalah klasifikasi. Prinsip dasar dari SVM adalah menemukan garis pemisah (*hyperplane*) terbaik yang dapat membedakan dua kelas atau lebih di dalam *feature space*. *Hyperplane* yang dianggap terbaik adalah yang mempunyai jarak atau margin terbesar ke titik data terdekat dari setiap kelas. Titik-titik data yang menjadi penentu dari *hyperplane* dan margin ini disebut sebagai *support vector*. [5] Dengan memaksimalkan margin, SVM bertujuan untuk membuat model yang memiliki kemampuan penyamarataan yang baik. Pada kenyataannya, kerap kali data tidak dapat dipisahkan secara linier. Untuk mengatasi masalah ini, SVM menggunakan sebuah teknik yang dikenal sebagai *kernel trick*. Konsep kernel memungkinkan SVM untuk memproyeksikan data ke ruang dimensi yang lebih tinggi, di mana data tersebut dapat dipisahkan secara linier. Terdapat beberapa jenis fungsi kernel yang umum digunakan, seperti Linear, Polynomial, dan *Radial Basis Function* (RBF). dalam penelitian ini, digunakan fungsi RBF. kernel RBF dipilih karena adaptabilitasnya dan kemampuannya dalam menangani hubungan yang kompleks dan non-linier antar fitur data. Kinerja dari SVM dengan kernel RBF sangat dipengaruhi oleh dua *hyperparameter* utama: C (Parameter Regularisasi) untuk mengontrol trade-off (kompromi) antara memaksimalkan margin dan meminimalkan kesalahan klasifikasi. Gamma (γ) untuk menentukan seberapa besar dampak dari satu sampel data pelatihan. [6]

Berikut merupakan rumus Matematis SVM:

$$u = \sum_{i=1}^N y_i a_i K(x_i, x) - b \quad (1)$$

Keterangan:

u = hasil klasifikasi
 y_i = kelas ke - 1 (+1 atau - 1)
 a_i = support vector dari data ke - i
 $K(x_i, x)$ = fungsi kernel
 b = nilai bias

Berikut merupakan fungsi *kernel trick* pada SVM:

Tabel 1. Kernel Support Vector Machine (SVM)

No	Fungsi Kernel	Formula	Parameter
1	Linear	$K(x, y) = x \cdot y$	C dan γ
2	RBF	$K(x, y) = e^{-\gamma \ x - y\ ^2}$	C dan γ
4	Polynomial	$K(x, y) = (\gamma x \cdot y + r)^d$	C, γ , r, dan d

Keterangan:

x, y = fitur data

γ = control variabel

r = koefisien *polynomial*

d = derajat *polynomial*

2.4. Hyperparameter Tuning

Performa dari model SVM dengan kernel RBF sangat bergantung pada pemilihan nilai *hyperparameter*-nya, yaitu C dan gamma. Pemilihan kombinasi nilai yang tidak dapat menghasilkan model yang *underfitting* atau *overfitting*. Setiap algoritma memiliki set *hyperparameter* yang berbeda, maka *hyperparameter tuning* berperan dalam memaksimalkan performa model pada data validasi.[5]

2.5. K-Fold Cross-Validation

Untuk mendapatkan evaluasi kinerja model yang stabil dan dapat diandalkan selama proses *hyperparameter tuning*, penelitian ini menerapkan teknik validasi silang atau K-Fold Cross - Validation. Teknik ini digunakan untuk memastikan bahwa kinerja model tidak bergantung pada cara data dibagi secara acak, sehingga dapat menurunkan risiko *overfitting* dan memberikan estimasi kinerja yang lebih akurat. K-Fold Cross-Validation merupakan salah satu metode yang umum digunakan dalam studi klasifikasi untuk pembagian data. Dalam metode ini, data pelatihan dibagi menjadi k bagian yang sama besar (disebut *fold*). Model kemudian dilatih sebanyak k kali, di mana setiap iterasi menggunakan k-1-fold untuk pelatihan dan 1-fold sebagai data validasi. Dengan demikian, setiap bagian data pernah digunakan sebagai data validasi satu kali. Nilai evaluasi model diperoleh dari rata-rata hasil pada semua iterasi. Pada penelitian ini digunakan *5-fold Cross-Validation*, yang juga banyak diterapkan dalam studi klasifikasi menggunakan SVM. rata-rata skor dari lima iterasi inilah yang digunakan sebagai dasar dalam memilih kombinasi *hyperparameter* terbaik.[7]

2.6. Confusion Matrix

Untuk melakukan evaluasi kinerja model klasifikasi secara mendalam, penelitian ini menggunakan Confusion Matrix. *Confusion matrix*, adalah salah satu pendekatan yang dapat digunakan untuk melakukan pengukuran kinerja model dengan menyajikan data hasil prediksi dalam bentuk tabel dan membandingkannya dengan kelas atau label aktual. Manfaat utama dari *confusion matrix* adalah tidak hanya menunjukkan jumlah prediksi yang benar dan salah, tetapi juga menunjukkan jenis kesalahan spesifik yang dibuat oleh model.

Tabel 2. Confusion Matrix

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TP)	False Positives (FP)
Predicted Negative (0)	False Negatives (FN)	True Negatives (TN)

Terdapat empat komponen utama dalam *confusion matrix* yang menjadi dasar pengukuran yang terdiri dari *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), *False Negative* (FN). TP adalah jumlah data dari kelas positif yang berhasil diprediksi dengan benar oleh model sebagai kelas positif. TN merujuk kepada jumlah data dari kelas negatif yang berhasil diprediksi dengan benar oleh model sebagai kelas negatif. FP mengacu kepada jumlah data dari kelas negatif yang salah diprediksi oleh model sebagai kelas positif. FN menunjukkan jumlah data dari kelas positif yang salah diprediksi oleh model sebagai kelas negatif. Nilai-nilai inilah yang kemudian sebagai dasar untuk menghitung metrik evaluasi kinerja yang lebih kuantitatif, seperti akurasi, F1-score, presisi, dan *recall*, yang direpresentasikan sebagai berikut.[2]

a. Akurasi

Akurasi adalah alat ukur untuk rasio total prediksi yang benar terhadap keseluruhan data uji, artinya *True Positives* (TP) dan *True Negatives* (TN) disini dilibatkan. Metrik ini memberikan gambaran umum tentang seberapa sering model membuat prediksi yang benar.

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

b. Presisi

Presisi menakar tingkat ketepatan dari prediksi yang diklasifikasikan sebagai kelas positif. Metrik ini berguna untuk mengetahui seberapa banyak prediksi positif yang dibuat oleh model benar-benar relevan.

$$Recall = \frac{TP}{TP + FP} \tag{3}$$

c. Recall

Recall menakar kemampuan model untuk menelaah kembali semua sampel yang seharusnya masuk dalam kelas positif. Metrik ini sangat krusial untuk mengetahui seberapa banyak kasus positif yang sebenarnya yang berhasil dideteksi oleh model.

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

d. F1-Score

F1-Score adalah rata-rata dari presisi dan *recall*. Metrik ini memberikan satu nilai tunggal yang memadukan kedua metrik tersebut. F1-Score sangat berguna, terutama dalam kasus dimana distribusi data antar kelas tidak selaras.

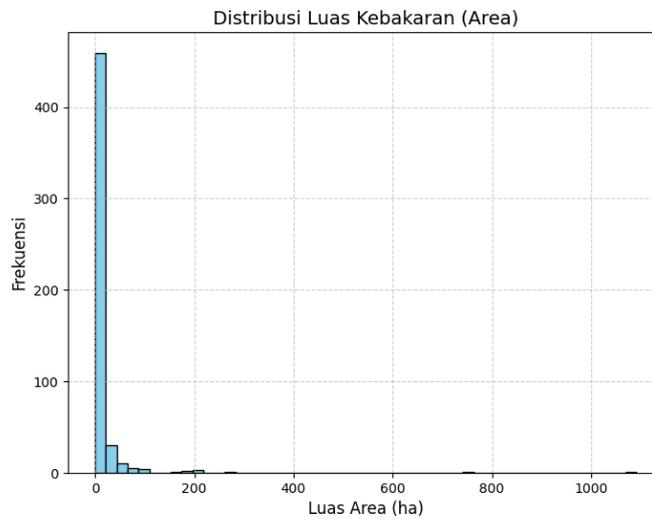
$$F1 - Score = 2 \times \tag{5}$$

3. Hasil dan Diskusi

Di bab ini akan disajikan hasil dari dua skenario: pengujian model SVM dengan penanganan data tidak seimbang (*class weighting*) sebagai model dasar, dan pengujian modeel setelah penerapan SMOTE serta optimasi *hyperparameter*

3.1. Analisis Karakteristik Dataset

Tahap awal dalam analisis hasil adalah memahami karakteristik dari dataset yang digunakan. Setelah dilakukan proses pra-pemrosesan untuk membentuk kelas target, dilakukan visualisasai untuk melihat distribusi jumlah sampel pada setiap kelas. Gambar 2. Menyajikan distribusi kelas tingkat risiko kebakaran pada dataset asli.

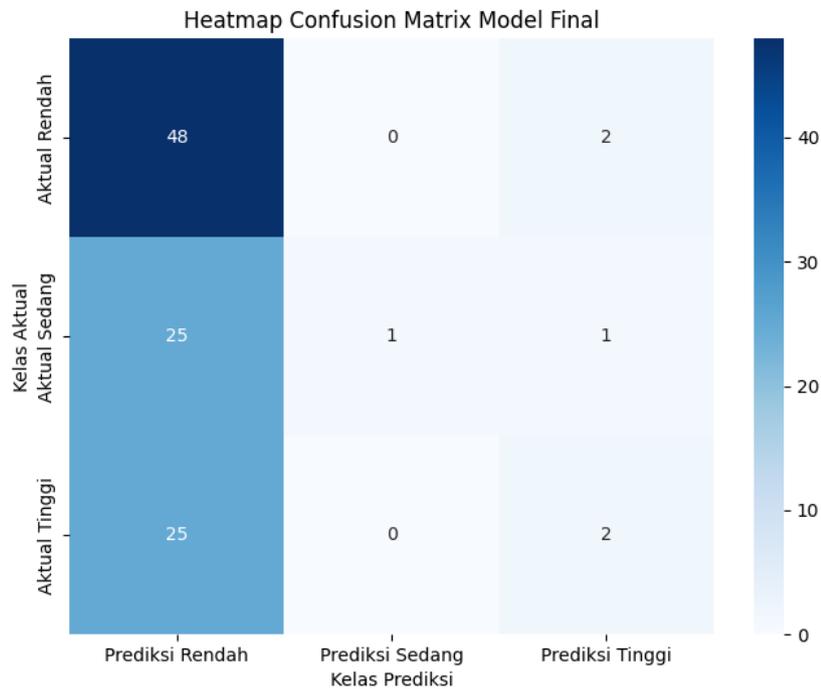


Gambar 2. Distribusi Jumlah Sampel untuk Setiap Kelas Risiko

Dari Gambar 2, terlihat dengan jelas bahwa terdapat ketidakseimbangan kelas yang signifikan dalam dataset. Kelas 'rendah' merupakan kelas mayoritas dengan jumlah sampel yang jauh lebih dominan dibandingkan dengan dua kelas lainnya. Sementara itu, kelas 'sedang' dan 'tinggi' merupakan kelas minoritas dengan jumlah sampel yang relatif lebih sedikit. Distribusi yang tidak seimbang ini merupakan tantangan fundamental dalam pemodelan klasifikasi. Jika tidak ditangani, model *machine learning* akan cenderung menjadi bias dan lebih memihak pada kelas mayoritasnya, sambil mengabaikan pola dari kelas minoritas. Fenomena ini yang menjadi justifikasi utama mengapa teknik penanganan data tidak seimbang, seperti SMOTE, perlu diterapkan pada tahap pra-pemrosesan sebelum data digunakan untuk melatih model.

3.2. Hasil Pengujian Model SVM Standar

Eksperimen pertama dilakukan dengan menerapkan model SVM standar (menggunakan kernel RBF dengan parameter default) pada dataset tanpa perlakuan penyeimbangan kelas. Tujuan dari pengujian ini adalah untuk mengetahui kinerja dasar dari algoritma dan mengidentifikasi potensi masalah yang muncul dari karakteristik data asli



Gambar 3. Confusion Matrix Model SVM Standar

Perilaku model yang paling menonjol dari *confusion matrix* ini adalah kegagalan totalnya dalam mengidentifikasi kelas ‘tinggi’. Dari 27 kasus, hanya 2 yang berhasil diprediksi dengan benar, 25 dari 27 kelas masuk ke kategori kelas ‘rendah’. Model juga masih kesulitan dengan kelas ‘sedang’, hanya berhasil menebak 1 dari 35 kasus dengan benar. Sebaliknya, model menunjukkan kecenderungan kuat untuk memprediksi kelas ‘rendah’.

Tabel 3. Laporan Klasifikasi Model SVM Standar

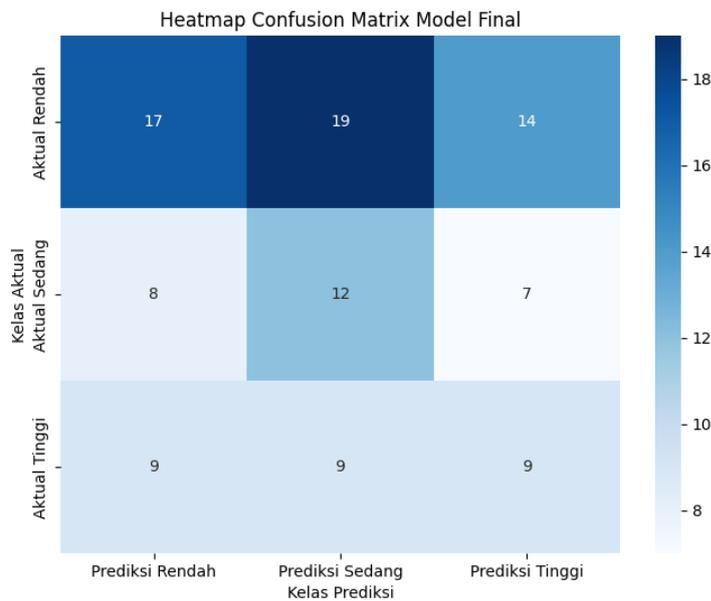
	Precision	Recall	F1-Score	Support
Rendah	0.49	0.96	0.65	50
Sedang	1.00	0.07	0.07	27
Tinggi	0.40	0.07	0.12	27
Accuracy			0.49	104
Macro Avg	0.63	0.36	0.28	104
Weighted Avg	0.60	0.49	0.36	104

Laporan klasifikasi ini mengonfirmasi temuan dari *confusion matrix*. Nilai recall yang tinggi untuk kelas ‘rendah’ menunjukkan bahwa model mampu memprediksi kelas ‘rendah’ dengan hampir sempurna. Namun model gagal untuk kelas ‘sedang’ dan ‘tinggi’ dengan skor hampir nol. Ini indikator atau pertanda bahwa model melewati lebih dari 90% kejadian kebakaran yang sebenarnya terjadi, sehingga model svm standar sangat tidak disarankan untuk memprediksi risiko, mengingat dataset cenderung *skewed* ke kanan.

3.3. Hasil Pengujian Model Awal

Skenario pengujian awal dilakukan untuk mengetahui kinerja dasar model SVM setelah masalah

ketidakseimbangan kelas (area) coba diatasi menggunakan parameter `class_weight = 'balanced'`. Pada tahap ini, model dilatih dengan parameter `C` dan `gamma default` dari library Scikit-learn. Evaluasi dilakukan pada data uji untuk mendapatkan gambaran awal kemampuan model SVM sebelum pengoptimasian. Hasil dari evaluasi model awal dilampirkan dalam bentuk Confusion Matrix pada Gambar 4 dan Laporan Klasifikasi pada Tabel 4.



Gambar 4. Confusion Matrix Model Awal

Hasil pada *confusion matrix* menunjukkan perubahan perilaku model yang drastis dibandingkan model standar. Model tidak lagi cenderung menebak 'Rendah' secara drastis. Terlihat bahwa model kini lebih baik dalam membuat prediksi untuk kelas 'Sedang' (12 prediksi benar) dan mulai mengidentifikasi beberapa kasus 'Tinggi' (9 prediksi benar), meskipun masih banyak terjadi kesalahan klasifikasi di antara ketiga kelas.

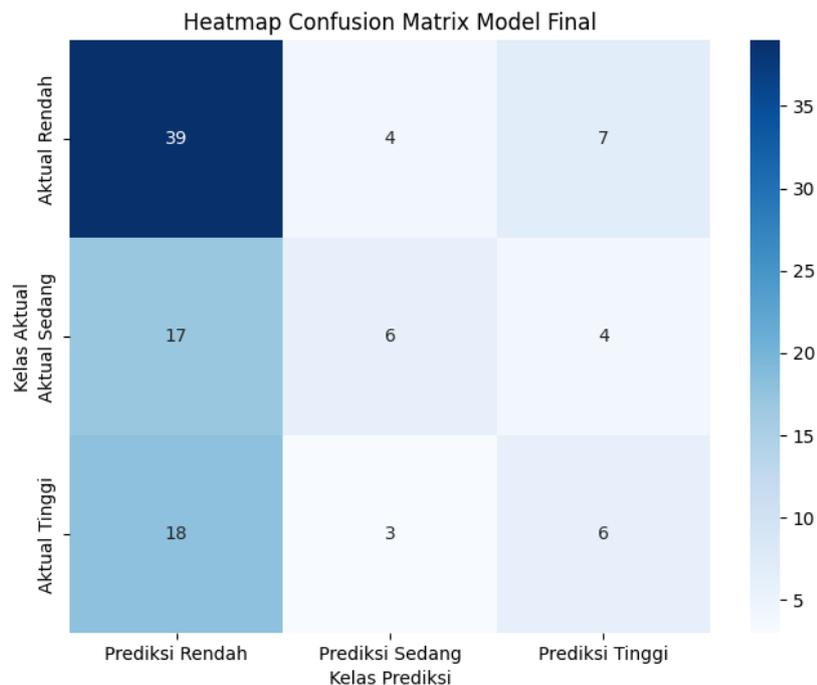
Tabel 4. Laporan Klasifikasi Model Awal

	Precision	Recall	F1-Score	Support
Rendah	0.50	0.34	0.40	50
Sedang	0.30	0.44	0.36	27
Tinggi	0.30	0.33	0.32	27
Accuracy			0.37	104
Macro Avg	0.37	0.37	0.36	104
Weighted Avg	0.40	0.37	0.37	104

Laporan klasifikasi mengonfirmasi bahwa metode `class_weight` berhasil mengurangi bias. Nilai recall untuk kelas "Sedang" meningkat signifikan menjadi 0,44, dan untuk kelas "Tinggi", meningkat dari 0,07 menjadi 0,32. Ini menunjukkan bahwa model tidak lagi "buta" terhadap kelas minoritas. Namun, peningkatan ini harus dibayar dengan harga: akurasi keseluruhan turun sedikit menjadi 37%, karena model tidak lagi menerima "poin mudah" dengan menebak kelas mayoritas. Skor F1 untuk semua kelas juga tetap rendah.

3.4. Hasil Pemodelan Kombinasi SMOTE dan GridSearchCV

Eksperimen final dilakukan dengan menerapkan pendekatan yang paling menyeluruh untuk memaksimalkan performa model. Pendekatan ini melibatkan teknik oversampling menggunakan SMOTE pada data latih untuk menciptakan dataaset yang seimbang, yang kemudian dilanjutkan dengan proses optimasi *hyperparameter* menggunakan GridSearchCV untuk mencari kombinasi C dan gamma yang optimal. Model terbaik kemudian dievaluasi pada data uji.



Gambar 5. Confusion Matrix Model Kombinasi SMOTE dan GridSearchCV

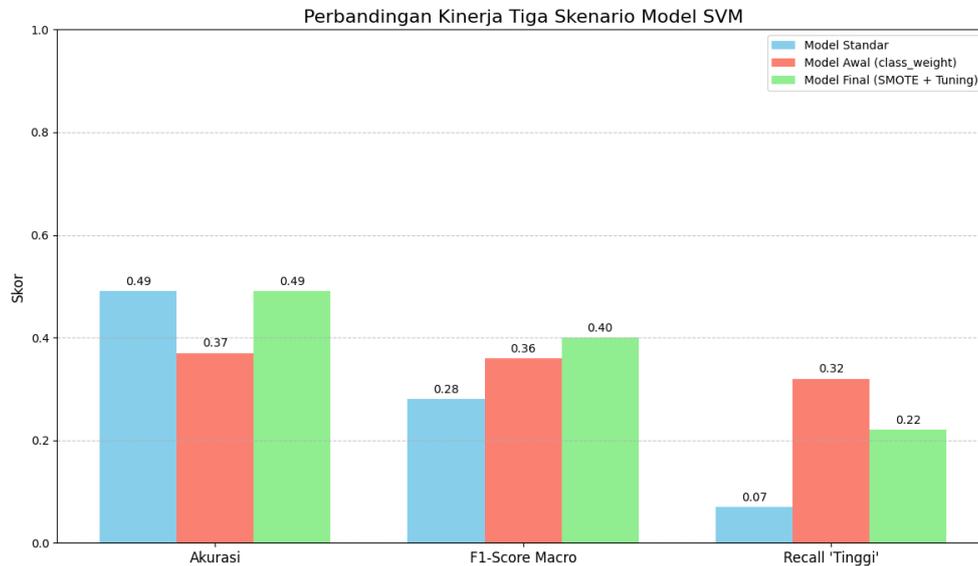
Tabel 5. Laporan Klasifikasi Model Kombinasi SMOTE dan GridSearchCV

	Precision	Recall	F1-Score	Support
Rendah	0.53	0.78	0.63	50
Sedang	0.46	0.22	0.30	27
Tinggi	0.35	0.22	0.27	27
Accuracy			0.49	104
Macro Avg	0.45	0.41	0.40	104
Weighted Avg	0.46	0.49	0.45	104

Berdasarkan evaluasi akhir, model final ini berhasil mencapai akurasi sebesar 49%. Laporan klasifikasi pada Tabel 5 menunjukkan bahwa model memiliki performa terbaiknya pada kelas 'rendah' dengan F1-Score sebesar 63%, yang terutama disebabkan oleh tingginya nilai recall 78%. Namun, kendala utama yang sebelumnya ditemukan tetap ada pada kelas-kelas minoritas. F1-Score untuk kelas 'sedang' hanya mencapai 30%, dan performa model pada kelas 'tinggi' bahkan lebih rendah dengan F1-Score 27%. Nilai recall untuk kelas "tinggi" yang hanya sebesar 22% mengindikasikan bahwa model gagal mengenali sebagian besar kasus kebakaran tingkat

tinggi. Hasil dari eksperimen akhir ini sangat signifikan. Meskipun terjadi sedikit peningkatan akurasi, ketidakmampuan model dalam mendeteksi kelas risiko tinggi, bahkan setelah penerapan SMOTE dan upaya optimasi lainnya.

3.5. Perbandingan Kinerja Model



Gambar 6. Perbandingan Kinerja Model

Untuk merangkum hasil dari ketiga skenario eksperimen, Gambar 6 menyajikan perbandingan kinerja model secara langsung. Grafik tersebut menunjukkan adanya perbaikan bertahap pada setiap langkah intervensi. Model standar awal menunjukkan bias yang parah dengan kegagalan total dalam mengenali kelas 'tinggi'. Penerapan *class_weight* berhasil mengatasi bias ini dan meningkatkan Recall untuk kelas 'tinggi' menjadi 0.32, meskipun dengan konsekuensi penurunan akurasi. Pendekatan final yang menggunakan SMOTE dan optimasi *hyperparameter* berhasil mencapai kinerja seimbang terbaik di antara ketiganya, dengan F1-Score Macro tertinggi 0.40 dan Recall 'tinggi' mencapai 0.22.

4. Kesimpulan

Penelitian ini telah berhasil menerapkan dan mengevaluasi algoritma *Support Vector Machine* (SVM) untuk tugas klasifikasi tingkat risiko kebakaran hutan multikelas ('Rendah', 'Sedang', 'Tinggi'). Melalui serangkaian eksperimen yang sistematis, mulai dari model SVM standar, penerapan penyeimbang kelas dengan *class_weight*, hingga pendekatan yang lebih menyeluruh menggunakan SMOTE dan optimasi *hyperparameter* melalui GridSearchCV, model terbaik dikembangkan dan dievaluasi. Hasil evaluasi akhir menunjukkan bahwa model SVM yang paling optimal hanya mampu mencapai akurasi sebesar 49% dengan F1-Score rata-rata 40%. Kinerja yang masih rendah ini, terutama pada kegagalan model untuk mengidentifikasi kelas risiko 'tinggi' secara andal, yakni skor recall yang hanya 22%, membawa kepada kesimpulan bahwa fitur-fitur meteorologi standar yang ada dalam dataset *Forest Fires* memiliki daya prediksi yang terbatas dan tidak memadai untuk secara akurat mengklasifikasikan tingkat keparahan kebakaran. Dengan demikian, penelitian ini menunjukkan bahwa meskipun SVM dapat diterapkan, keberhasilan klasifikasi untuk masalah yang kompleks ini sangat bergantung pada kekayaan informasi dari fitur yang digunakan, bukan hanya pada optimasi algoritma belaka.

Daftar Pustaka

- [1] M. R. F. H. Y. F. R. Adithea Loren, "Analisis Faktor Penyebab Kebakaran Hutan dan Lahan Serta Upaya Pencegahan yang Dilakukan Masyarakat di Kecamatan Basarang Kabupaten

- Kapuas Kalimantan Tengah,” Nov. 2015.
- [2] I. M. W. W. A. A. I. N. E. K. I. W. S. Putu Ayu Novia Aryanti, “Penerapan Machine Learning Dalam Analisis Sentimen dan Pemodelan Topik Data Opini Kendaraan Listrik,” Dec. 2024.
- [3] J. Elektronik Ilmu Komputer Udayana *et al.*, “Diagnosis Penyakit Retinopati Diabetes Menggunakan SVM dengan Optimasi Algoritma Genetika”.
- [4] E. Dazhi, J. Liu, M. Zhang, H. Jiang, and K. Mao, “RE-SMOTE: A Novel Imbalanced Sampling Method Based on SMOTE with Radius Estimation,” *Computers, Materials and Continua*, vol. 81, no. 3, pp. 3853–3880, 2024, doi: 10.32604/cmc.2024.057538.
- [5] J. V Bagade *et al.*, “An ensemble approach for detection of diabetes using SVM and DT,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 38, no. 1, pp. 1–1, 2025, doi: 10.11591/ijeecs.v38.i1.pp1-1x.
- [6] M. Desiawan and A. Solichin, “SVM Optimization with Grid Search Cross Validation for Improving Accuracy of Schizophrenia Classification Based on EEG Signal,” *Jurnal Teknik Informatika*, vol. 17, no. 1, pp. 10–20, May 2024, doi: 10.15408/jti.v17i1.37422.
- [7] J. Elektronik *et al.*, “Deteksi Rasa Buah Jeruk Siam Kintamani Menggunakan SVM dengan Optimasi Algoritma Genetika”.
- [8] P. Cortez and A. Morais. A data mining approach to predict forest fires using meteorological data. In J. Neves, M.F. Santos, and J. Machado, editors, Proceedings of the 13th Portuguese Conference on Artificial Intelligence, pages 512–523, 2007.