Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 2, Desember 2025
e-ISSN 2685-998X
DOI 10.30865/json.v7i2.9299

Optimasi Linear Support Vector Machine untuk Deteksi Smishing Multi-Kelas pada Dataset Tidak Seimbang

Anggun Vannia, Muljono*

Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang, Indonesia
Email: [1]111202214106@mhs.dinus.ac.id, [2*]muljono@dsn.dinus.ac.id
Email Penulis Korespondensi: muljono@dsn.dinus.ac.id*
Submitted: **03/11/2025**; Accepted: **07/12/2025**; Published: **31/12/2025**

Abstrak–Serangan smishing (SMS phishing) menghadapi tantangan mendasar dalam deteksi berbasis *machine learning* akibat ketidakseimbangan distribusi kelas pada dataset dunia nyata, di mana *instance* kelas minoritas (smishing) justru paling kritis untuk diidentifikasi. Penelitian ini mengusulkan sebuah *framework robust* yang mengoptimasi *Linear Support Vector Machine* (SVM) dengan strategi *hybrid sampling* tiga tingkat untuk klasifikasi multi-kelas pada kondisi data tidak seimbang. *Framework* yang dikembangkan mengintegrasikan ekstraksi fitur hibrida TF-IDF dan *meta-features* dengan strategi penanganan ketidakseimbangan data yang komprehensif, yang meliputi *Random Oversampling* (ROS) untuk kelas minoritas, *Random Undersampling* (RUS) untuk kelas mayoritas, dan *Embedding MixUp* untuk augmentasi data level *embedding*. Optimasi parameter melalui *GridSearchCV* dengan validasi 5-*fold* berhasil menentukan konfigurasi optimal SVM Linear (C=0.5). Hasil evaluasi pada *test set* mendemonstrasikan kemampuan klasifikasi yang tinggi dan seimbang, dengan pencapaian akurasi 96,7% dan F1-*macro* 87,6%. Kinerja yang konsisten merata pada semua kelas ini tercermin dari *recall smishing* 84% sambil mempertahankan *recall ham* 99%. Temuan ini menegaskan bahwa kombinasi Linear SVM dan strategi *hybrid sampling* berhasil menghasilkan model deteksi *smishing* yang *robust*, seimbang, dan siap diimplementasikan dalam skenario dunia nyata.

Kata Kunci: *Smishing*; Deteksi; SVM Linear; *Hybrid Sampling*; Ketidakseimbangan Kelas

Abstract– Smishing (SMS phishing) attacks face fundamental challenges in machine learning-based detection due to class distribution imbalance in real-world datasets, where minority class instances (smishing) are the most critical to identify. This research proposes a robust framework that optimizes Linear Support Vector Machine (SVM) with a three-level hybrid sampling strategy for multi-class classification in imbalanced data conditions. The developed framework integrates TF-IDF hybrid feature extraction and meta-features with a comprehensive data imbalance handling strategy, which includes Random Oversampling (ROS) for minority classes, Random Undersampling (RUS) for majority classes, and Embedding MixUp for embedding-level data augmentation. Parameter optimization through GridSearchCV with 5-fold validation successfully determined the optimal configuration of Linear SVM (C=0.5). Evaluation results on the test set demonstrated high and balanced classification capabilities, achieving 96.7% accuracy and 87.6% F1-macro. This consistent performance across all classes is reflected in an 84% recall for smishing while maintaining a 99% recall for ham. These findings confirm that the combination of Linear SVM and hybrid sampling strategies successfully produces a robust, balanced smishing detection model that is ready for implementation in real-world scenarios.

Keywords: Smishing; Detection; Linear SVM; Hybrid Sampling; Class Imbalance

1. PENDAHULUAN

Di era digital, *phishing* telah berevolusi menjadi ancaman siber yang semakin *prevalen*, salah satunya melalui media pesan singkat (SMS). Serangan *phishing* melalui SMS ini secara spesifik dikenal sebagai *smishing* (*SMS phishing*). Modus kejahatan siber ini banyak memanfaatkan pesan singkat sebagai sarana untuk mengelabui korban dan mencuri data pribadi dengan menyamar sebagai lembaga resmi, seperti bank, instansi pemerintah, maupun perusahaan terpercaya [1]. Teknik serangan umumnya melibatkan pengiriman pesan yang dirancang untuk memicu respons cepat, mengarahkan korban ke tautan berbahaya, lampiran bermuatan *malware*, atau formulir palsu untuk mengekstrak informasi kredensial dan data sensitif [2]. Seluruh mekanisme ini mengandalkan rekayasa psikologis yang memanipulasi rasa urgensi dan kepercayaan korban.

Eskalasi serangan *smishing* tidak hanya menimbulkan ancaman teoritis, tetapi telah terkonfirmasi melalui data empiris yang mengkhawatirkan. Laporan terkini mengungkapkan peningkatan signifikan serangan SMS *phishing* di Indonesia, dengan kenaikan mencapai 37% pada tahun 2023 [3]. Kerugian finansial yang material pun tidak terelakkan, menimpa level individu, korporasi, hingga instansi pemerintah. Di saat yang sama, pola serangan global turut menunjukkan peningkatan kecanggihan, di mana pesan *phishing* dirancang semakin adaptif dengan konteks linguistik dan budaya lokal untuk meningkatkan tingkat keberhasilan [4]. Oleh karena itu, menjadi suatu keharusan untuk mengembangkan sistem deteksi *smishing* yang tidak hanya akurat, tetapi juga adaptif terhadap pola serangan yang dinamis dan efisien dari sisi komputasi.

Dalam upaya memenuhi kebutuhan tersebut, berbagai pendekatan deteksi telah dikembangkan. Namun, metode yang ada masih menghadapi sejumlah permasalahan mendasar. Permasalahan dalam deteksi *smishing* terletak pada ketidakmampuan model dalam mengakomodasi variasi linguistik yang luas, keterbatasan ruang lingkup pesan singkat, serta distribusi kelas yang tidak seimbang secara signifikan. Kondisi ketidakseimbangan ini mengakibatkan model lebih cenderung bias terhadap kelas mayoritas, sehingga menurunkan kinerja deteksi

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 2, Desember 2025
e-ISSN 2685-998X
DOI 10.30865/json.v7i2.9299

pada kelas minoritas yang justru memiliki tingkat urgensi lebih tinggi. Di sisi lain, metode deteksi yang ada saat ini belum berhasil menerapkan strategi penyeimbangan data secara efektif tanpa mengurangi kualitas representasi konteks pesan.

Menanggapi kebutuhan tersebut, komunitas peneliti *cybersecurity* telah gencar mengembangkan solusi deteksi otomatis yang memanfaatkan teknik *machine learning*. Pendekatan machine learning dirancang untuk menciptakan algoritma prediksi yang akurat, yang performanya sangat dipengaruhi oleh kualitas dan jumlah data yang digunakan untuk melatih sistem [5][6]. Perkembangan algoritma machine learning terus menunjukkan peningkatan kemampuan dalam mengenali pola serangan yang semakin kompleks [7]. Perkembangan ini terlihat dari peralihan pendekatan yang dimulai dari model klasik sederhana hingga arsitektur yang semakin canggih. Pada tahap awal, penelitian banyak berfokus pada penerapan algoritma klasik dengan fitur dasar. Salah satu yang menonjol adalah *Support Vector Machine* (SVM), yang dikombinasikan dengan fitur *n-gram* dan *kernel* RBF, dilaporkan mampu mencapai akurasi 87,3% dengan validasi silang *k-fold* [8]. Dalam konteks ini, SVM terus menjadi pilihan karena keunggulan fundamentalnya, yaitu kemampuannya membangun *decision boundary* yang *robust* hanya dengan mengandalkan *support vectors*, menjadikannya efisien dan *powerful* untuk menangani data berdimensi tinggi [9].

Sejalan dengan itu, beberapa penelitian mengusulkan arsitektur yang lebih komprehensif. Sebagian besar pendekatan tidak hanya mengandalkan konten teks, tetapi juga berbagai modul analisis. Seperti model *Smishing Detector* berbasis *Naive Bayes* yang mengintegrasikan empat modul analisis seperti SMS *Content*, URL, *Source Code*, dan APK, sehingga berhasil mendongkrak akurasi hingga 96,29% [10]. Strategi peningkatan kinerja model kini lebih difokuskan pada optimalisasi menyeluruh terhadap algoritma yang ada melalui kombinasi teknik yang sinergis. Sebagai contoh, penerapan *hybrid* optimisasi (*Reptile + Prairie Dogs*) dan teknik *oversampling* dilaporkan terbukti mampu lebih meningkatkan performa sistem deteksi yang tetap berbasis pada SVM [11].

Meskipun optimasi tersebut mampu meningkatkan performa, akar permasalahan seperti ketidakseimbangan kelas pada *dataset* smishing seringkali tidak tertangani secara komprehensif. Kondisi ini dapat menyebabkan bias model terhadap kelas mayoritas sehingga menurunkan kinerja deteksi pada kelas minoritas. Sebagai algoritma yang populer digunakan, kinerja SVM pada *dataset* tidak seimbang seringkali terbatas akibat kecenderungannya mendominasi kelas mayoritas [12]. Penelitian lain mencoba mengatasi keterbatasan ini. Misalnya, [13] mengusulkan pendekatan *hybrid* yang mengintegrasikan teknik *resampling*, reduksi fitur, dan optimisasi parameter berbasis *Improved Simulated Annealing* (ISA). Meskipun berhasil meningkatkan akurasi hingga 89,65% pada berbagai dataset publik, pendekatan tersebut belum secara spesifik menguji efektivitasnya dalam konteks deteksi *smishing* multi-kelas.

Penelitian terdahulu telah berkontribusi signifikan melalui pemanfaatan SVM, integrasi berbagai modul analisis, dan penerapan metode optimisasi yang beragam. Namun, sebagian besar studi masih menunjukkan beberapa keterbatasan penting. Strategi penyeimbangan data yang digunakan umumnya hanya bertumpu pada satu teknik, seperti oversampling atau undersampling, sehingga belum mampu mengatasi ketidakseimbangan kelas secara menyeluruh. Selain itu, mayoritas model hanya bergantung pada satu jenis fitur dan belum memanfaatkan kombinasi TF-IDF dengan meta-fitur pesan yang berpotensi memberikan representasi yang lebih kaya. Pendekatan hybrid yang telah diusulkan sebelumnya pun belum dievaluasi secara khusus pada konteks deteksi smishing multi-kelas. Kondisi tersebut menegaskan adanya celah yang memerlukan solusi yang lebih komprehensif dan terstruktur.

Masalah inti dari penelitian ini terletak pada rendahnya kinerja deteksi smishing pada kelas minoritas akibat ketidakseimbangan kelas dan keterbatasan representasi fitur. Kondisi ini penting untuk diatasi karena kesalahan mendeteksi pesan berbahaya dapat berdampak langsung pada kerugian finansial dan keamanan pengguna. Untuk menjawab gap tersebut, penelitian ini mengusulkan optimasi *Linear Support Vector Machine* (SVM) melalui strategi hybrid sampling yang dirancang khusus untuk data teks pesan. Kerangka kerja yang dikembangkan mengintegrasikan ekstraksi fitur hibrida yang menggabungkan TF-IDF dan meta-fitur, dengan strategi penyeimbangan data yang komprehensif dalam penyeimbangan distribusi kelas. Strategi *hybrid sampling* ini menerapkan tiga teknik yaitu *Random Oversampling* (ROS) pada kelas minoritas untuk memperbanyak representasi pattern penting [14], Random Undersampling (RUS) pada kelas mayoritas untuk mengurangi bias tanpa menghilangkan informasi kritis [15], dan augmentasi data level *embedding* menggunakan teknik *MixUp* untuk menciptakan variasi sampel sintetik di ruang *embedding* [16].

Pendekatan *hybrid* ini dipilih karena kemampuannya dalam menyeimbangkan distribusi kelas tanpa mengorbankan informasi penting pada kedua kelas mayoritas dan minoritas [14]. Selanjutnya, optimasi parameter model dilakukan menggunakan *GridSearchCV* dengan validasi 5-*fold* berbasis metrik F1-*macro* guna menentukan konfigurasi paling optimal, sehingga dihasilkan model yang tidak hanya akurat tetapi juga *robust*. Untuk memvalidasi efektivitas *framework*, evaluasi komprehensif tidak hanya mengukur performa melalui akurasi dan F1-*score*, tetapi juga menganalisis konsistensi kinerja pada masing-masing kelas guna memastikan kemampuan *framework* dalam mengatasi bias klasifikasi.

Berdasarkan rancangan tersebut, penelitian ini bertujuan mendemonstrasikan kapabilitas kerangka kerja terintegrasi yang memadukan Linear SVM dengan strategi *hybrid sampling* bertingkat tiga. Validasi eksperimen difokuskan pada pencapaian kinerja yang seimbang dan tangguh pada seluruh kelas sebagai bukti efektivitas

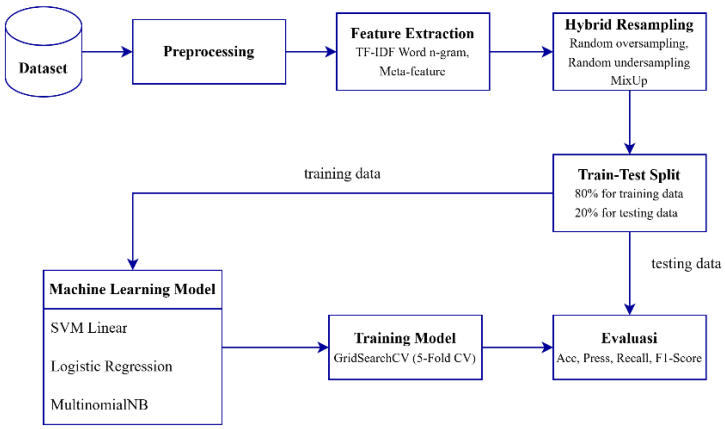
Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 2, Desember 2025
e-ISSN 2685-998X
DOI 10.30865/json.v7i2.9299

pendekatan yang diusulkan. Diharapkan dihasilkan sebuah solusi deteksi yang aplikatif dan siap diimplementasikan dalam skenario dunia nyata guna mendukung sistem deteksi *smishing* yang lebih adaptif dan responsif.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Tahapan penelitian ini disusun secara sistematis sebagaimana ditunjukkan pada Gambar 1. Proses dimulai dari pengumpulan dataset yang selanjutnya melalui tahap prapemrosesan sebelum dilakukan ekstraksi fitur menggunakan TF-IDF *word n-gram* dan meta-fitur. Data yang telah direpresentasikan kemudian ditangani dengan skema *hybrid sampling* untuk mengatasi ketidakseimbangan kelas. Setelah itu, dataset dibagi menjadi data latih dan data uji menggunakan *stratified train–test split*. Data yang sudah diproses kemudian digunakan dalam tahap permodelan, di mana tiga algoritma *machine learning* yaitu *Linear Support Vector Machine* (SVM), *Logistic Regression*, dan *Multinomial Naive Bayes* diterapkan dan dioptimasi melalui *GridSearchCV* dengan validasi *5-fold cross validation*. Seluruh proses diakhiri dengan evaluasi model menggunakan metrik *accuracy*, *precision*, *recall*, dan F1-*score* untuk menilai efektivitas pendekatan yang diusulkan.



Gambar 1. Alur Penelitian

2.2 Dataset

Dataset yang digunakan dalam penelitian ini diperoleh dari repositori Mendeley Data dengan judul *SMS Phishing Dataset for Machine Learning and Pattern Recognition*. Dataset ini berisi sebanyak 5.971 data SMS yang telah dilabeli ke dalam tiga kategori, yaitu *smishing*, *ham*, dan spam. Label tersebut mencakup 489 pesan spam, 638 pesan *smishing*, dan 4844 pesan *ham*. Komposisi dataset yang tidak seimbang ini merepresentasikan kondisi nyata di mana pesan normal (*ham*) cenderung mendominasi komunikasi SMS sehari-hari. Selain atribut utama berupa teks pesan, setiap data juga dilengkapi dengan tiga fitur biner tambahan berupa indikator keberadaan URL, email, dan nomor telepon, yang dapat memberikan informasi relevan dalam mendeteksi potensi *phishing*. Karakteristik fitur biner yang tersedia juga memperkaya representasi data sehingga dapat meningkatkan kemampuan model dalam mengidentifikasi pola-pola mencurigakan.

2.3 Prapemrosesan Data

Prapemrosesan data merupakan tahap fundamental untuk membersihkan, menormalkan, dan mengubah data mentah menjadi format yang siap digunakan oleh model *machine learning*. Proses ini dapat meningkatkan kualitas data, menjaga konsistensi, dan mengurangi noise pada data. Tahap prapemrosesan data atau *preprocessing data* diawali dengan normalisasi label untuk menyamakan variasi penulisan kategori seperti "*smshing*" menjadi "*smishing*" dan memastikan konsistensi label "spam" serta "*ham*". Proses ini dilanjutkan dengan pembersihan teks melalui konversi huruf kecil, penghapusan pola URL, email, dan nomor telepon menggunakan ekspresi reguler, serta tokenisasi dan penyaringan kata. Informasi yang dihapus dari teks utama tetap dipertahankan sebagai *meta-features* untuk memperkaya proses klasifikasi. Penanganan nilai hilang juga dilakukan secara sistematis dengan mengganti nilai kosong menjadi string kosong agar format input tetap seragam dan integritas data terjaga selama tahap *feature extraction*.

2.4 Ekstraksi Fitur

Ekstraksi fitur bertujuan untuk mengubah data teks mentah yang telah diproses menjadi representasi numerik yang dapat dipahami oleh algoritma *machine learning*. Tahap ini penting karena kualitas representasi fitur sangat

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 2, Desember 2025
e-ISSN 2685-998X
DOI 10.30865/json.v7i2.9299

mempengaruhi performa model klasifikasi. Dalam penelitian ini, ekstraksi fitur dilakukan dengan pendekatan multi-modal yang menggabungkan informasi berbasis konten tekstual dengan aspek struktural dan non-linguistik pesan melalui pendekatan multi-modal menggunakan *Column Transformer* [17]. Fitur utama diekstrak dari kolom teks melalui metode *Term Frequency-Inverse Document Frequency* (TF-IDF) [18], yang bertujuan untuk mengukur tingkat kepentingan setiap *term* dalam sebuah pesan relatif terhadap seluruh korpus. Untuk meningkatkan generalisasi, *vectorizer* disesuaikan dengan mengimplementasikan fungsi *tokenizer* yang memproses teks tanpa menggunakan *stemming*, sehingga setiap token dipertahankan dalam bentuk aslinya sebelum perhitungan frekuensi. Untuk menangkap keterkaitan kata, fitur TF-IDF diesktrak dalam bentuk *n-gram* dari *uni-gram* hingga *tri-gram*.
Selain fitur konten, delapan meta-fitur direkayasa untuk menangkap karakteristik struktural unik pesan *smishing*. Meta-fitur ini mencakup pengukuran seperti panjang pesan (*length*), jumlah digit , dan jumlah kemunculan pola spesifik. Selain itu, fitur biner direkayasa dari kolom terpisah untuk mempertahankan informasi keberadaan URL, email, atau nomor telepon yang sebelumnya dihapus dari teks utama. Seluruh meta-fitur yang bersifat numerik kemudian distandarisasi menggunakan *Standard Scaler* [19] untuk menormalkan rentang nilainya, mencegah fitur dengan skala yang lebih besar mendominasi proses pelatihan model klasifikasi *Support Vector Machine* (SVM).

2.5 Strategi Penyeimbangan Data

Strategi penyeimbangan data diterapkan untuk mengatasi ketidakseimbangan kelas pada dataset. Arsitektur model yang diusulkan dalam penelitian ini merupakan alur kerja terintegrasi yang dirancang untuk secara efektif menangani masalah ketidakseimbangan kelas yang inheren dalam data *smishing*, di mana rasio kelas mayoritas '*ham*' mencapai sekitar 81% dari total data [20]. Alur kerja ini dirancang untuk mengintegrasikan teknik ekstraksi fitur *multi-modal* dengan strategi penyeimbangan data *multi-level*, serta model klasifikasi dalam satu rangkaian yang kohesif [21]. Untuk menyeimbangkan distribusi data pelatihan, digunakan kombinasi tiga pendekatan, yaitu *EmbeddingMixUp, Random Oversampling* (ROS), *dan Random UnderSampling* (RUS).
Embedding MixUp dipilih karena mampu menghasilkan variasi data sintetis yang lebih kontekstual dibandingkan *oversampling* tradisional [22], dengan cara menggabungkan dua representasi fitur TF-IDF yang berbeda untuk membentuk sampel baru tanpa sekadar menduplikasi data. Strategi ini membantu memperkaya representasi semantik pada kelas minoritas dan mengurangi risiko *overfitting* yang sering muncul pada metode ROS murni. Sementara itu, ROS tetap digunakan untuk meningkatkan proporsi kelas *smishing* dan *spam*, sedangkan RUS berfungsi mengurangi dominasi kelas *ham* agar distribusi akhir menjadi lebih proporsional [23]. Kombinasi ketiga metode ini menghasilkan dataset yang seimbang dan informatif, memungkinkan model mempelajari karakteristik setiap kelas secara lebih representatif serta meminimalkan bias terhadap kelas mayoritas.

2.6 Pembagian Data

Pembagian data (*train-test split*) merupakan prosedur penting untuk mengukur kemampuan generalisasi model, yaitu kinerjanya pada data yang belum pernah dilihat selama pelatihan. Prosedur ini mencegah model mengalami *overfitting* terhadap data latih. Dalam penelitian ini, dataset dibagi dengan rasio 80:20 untuk *training set* dan *testing set*. Pemecahan data ini dilakukan secara *stratified* untuk mempertahankan proporsi distribusi kelas asli pada kedua subset, mengingat karakteristik dataset yang tidak seimbang [24]. Pendekatan ini memastikan bahwa representasi setiap kelas pada data latih dan uji tetap proporsional. Pendekatan ini memastikan estimasi kinerja model bersifat objektif dan tidak bias.

2.7 Arsitektur Model dan Optimasi

Arsitektur model dan optimasi dalam penelitian *machine learning* menentukan kerangka kerja dan parameter terbaik untuk membangun sistem prediktif yang akurat dan general. Sebagai model klasifikasi utama, dipilih algoritma *Support Vector Machine* (SVM) yang telah teruji efektivitasnya dalam membangun *decision boundary* yang *robust* dalam ruang fitur berdimensi tinggi [9]. SVM merupakan pilihan yang menonjol untuk tugas klasifikasi teks karena kemampuannya dalam memproses data *sparse* dengan kompleksitas yang tinggi [25]. Berdasarkan pertimbangan efisiensi komputasi dan kesesuaiannya dengan data bersifat *high dimensional* dan *sparse* seperti representasi TF-IDF, *Linear Support Vector Machine* (SVM) dipilih sebagai model klasifikasi utama dalam penelitian ini. Algoritma ini merupakan varian SVM yang secara khusus dioptimasi untuk permasalahan klasifikasi linear.
Pemilihan model SVM linear didukung oleh literatur yang menunjukkan bahwa teknik klasifikasi linear sangat menjanjikan dan efisien untuk menangani fitur *sparse* berdimensi besar [26], seperti yang dihasilkan oleh ekstraksi fitur TF-IDF *Word n-gram*, terutama dalam tugas klasifikasi teks berskala besar. SVM dengan kernel linear secara khusus dipilih karena efisiensinya dalam pemrosesan dan kemampuannya untuk beroperasi lebih cepat dibandingkan non-linear SVM, namun dengan akurasi yang sebanding pada data teks [27]. Pemilihan model Linear SVM dalam deteksi smishing didukung oleh pertimbangan interpretabilitas. Model yang *inherently interpretable* seperti *linear models* memungkinkan analisis yang lebih transparan terhadap fitur-fitur yang berkontribusi dalam klasifikasi pesan mencurigakan [28]. Optimasi model dilakukan menggunakan *Grid Search Cross-Validation* (GridSearchCV) dengan 5 *fold Cross-Validation* dan menggunakan *Accuracy* sebagai metrik utama.

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 2, Desember 2025
e-ISSN 2685-998X
DOI 10.30865/json.v7i2.9299

2.8 Model Pembanding

Untuk memvalidasi kinerja SVM dengan kernel linear, dua model klasifikasi populer lainnya digunakan sebagai pembanding yaitu *Logistic Regression* (LR) dan *Naive Bayes* (NB), diimplementasikan sebagai pembanding untuk mengevaluasi konsistensi dan keunggulan relatif pendekatan Linear *Support Vector Machine* (SVM) dalam konteks deteksi *smishing*. *Logistic Regression* berfungsi sebagai model *baseline* linear yang umum digunakan pada klasifikasi teks [29], sedangkan Naive Bayes mewakili pendekatan probabilistik yang dikenal efisien pada data dengan representasi TF-IDF [30]. Kedua model pembanding ini dilatih menggunakan rangkaian pemrosesan dan strategi penyeimbangan data yang sama dengan SVM linear, sehingga perbandingan performa dapat dilakukan secara objektif dalam kondisi eksperimental yang setara. Perbandingan ini dilakukan untuk mengetahui sejauh mana SVM linear mampu mengungguli model-model klasik yang telah mapan dalam menangani tugas deteksi smishing. Hasil evaluasi yang komprehensif ini diharapkan dapat memberikan rekomendasi model yang paling robust dan efektif untuk diaplikasikan pada sistem deteksi yang sesungguhnya.

2.9 Evaluasi

Evaluasi kinerja model klasifikasi merupakan tahap krusial dalam memastikan keandalan sistem deteksi *smishing* yang dikembangkan. Mengingat distribusi data yang sangat tidak seimbang, penilaian tidak hanya difokuskan pada metrik akurasi, tetapi juga pada *precision*, *recall*, dan *F1-score* untuk menilai kemampuan model dalam mengenali kelas minoritas seperti *smishing* dan *spam*. Dalam konteks klasifikasi multi-kelas (*ham*, *spam*, *smishing*), perhitungan metrik dilakukan berdasarkan nilai *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)*, dan *False Negative (FN)* pada setiap kelas, kemudian digabungkan menggunakan pendekatan *macro-average* agar setiap kelas berkontribusi secara seimbang terhadap skor akhir, terlepas dari perbedaan jumlah datanya. Pendekatan ini memungkinkan evaluasi yang lebih representatif terhadap performa model pada seluruh kategori pesan.

3. HASIL DAN PEMBAHASAN

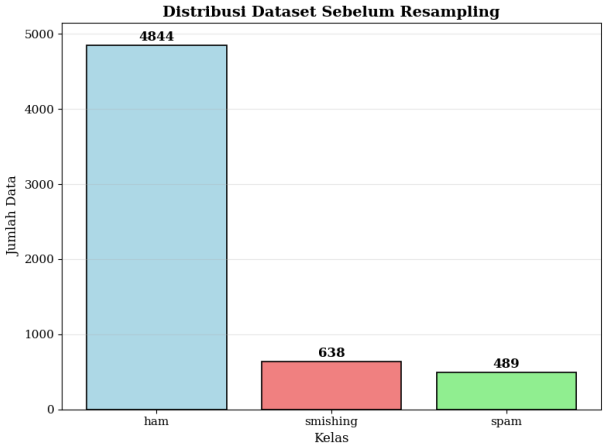
3.1 Statistik Deskripsi Dataset

Analisis statistik deskriptif dataset menunjukkan distribusi kelas yang tidak seimbang seperti terlihat pada Gambar 2. Kelas *ham* mendominasi dengan 4.844 sampel, sementara kelas *smishing* dan spam masing-masing hanya terdiri dari 638 dan 489 sampel.

Tabel 1. Distribusi Dataset Sebelum *Resampling*

Kelas	Jumlah Data
Ham	4.844
Smishing	638
Spam	489

Disproporsi yang signifikan dalam representasi kelas ini berpotensi menimbulkan bias pada model selama fase pelatihan, di mana model cenderung lebih mempelajari karakteristik kelas mayoritas. Akibatnya, kinerja model dalam mengidentifikasi instance dari kelas minoritas dapat menurun secara signifikan. Oleh karena itu, teknik penyeimbangan data dengan pendekatan *hybrid sampling* diimplementasikan guna menciptakan distribusi data yang lebih merata sebelum tahap pemodelan.



Gambar 2. Distribusi Dataset

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 2, Desember 2025
e-ISSN 2685-998X
DOI 10.30865/json.v7i2.9299

Tabel 2. Data Mentah Berdasarkan Label

Text	Label
Velly good, yes please!	*Ham*
This is the 2nd time we have tried 2 contact u. U have won the 750 Pound prize. 2 claim is easy, call 08712101359 NOW! Only 10p per min. BT-national-rate	*Smishing*
I don't know u and u don't know me. Send CHAT to 86688 now and let's find each other! Only 150p/Msg rcvd. HG/Suite342/2Lands/Row/W1J6HL LDN. 18 years or over	*Spam*

3.2 Prapemrosesan Data

Tahap prapemrosesan data teks dilakukan untuk membersihkan dan standarisasi data sebelum proses ekstraksi fitur. Proses ini melibatkan serangkaian transformasi teks yang bertujuan menghasilkan data terstruktur, seperti yang ditunjukkan pada Tabel 3. Proses prapemrosesan teks melibatkan penghapusan tanda baca, konversi seluruh karakter menjadi huruf kecil, serta normalisasi kata-kata informal menjadi bentuk bakunya. Hasil transformasi tersebut menghasilkan teks yang telah terbebas dari *noise* dan inkonsistensi linguistik, sehingga siap untuk diproses lebih lanjut dalam tahap ekstraksi fitur.

Tabel 3. Hasil Prapemrosesan Data

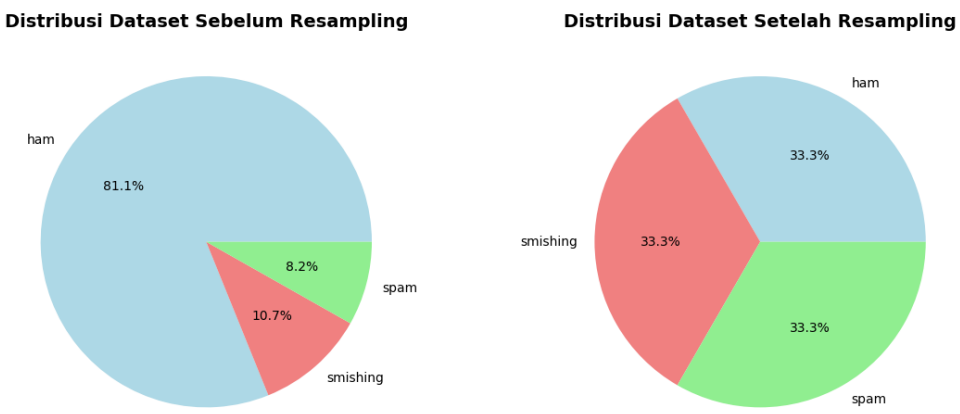
Text	Pembersihan Data
What's up? Do you want me to come online? If you are free we can talk sometime?	what 's up do you want me to come online if you are free we can talk sometime
So u workin overtime nigpun?	so you workin overtime nigpun
Also sir, i sent you an email about how to log into the usc payment portal. I.ll send you another message that should explain how things are back home. Have a great weekend.	also sir i sent you an email about how to log into the usc payment portal i ll send you another message that should explain how things are back home have a great weekend

3.3 Hasil Penanganan Ketidakseimbangan Kelas

Strategi penyeimbangan dataset berhasil mengatasi ketidakseimbangan kelas sebagaimana divisualisasikan pada Gambar 3. Distribusi dataset yang awalnya didominasi oleh kelas *ham* sebesar 81,1% berhasil ditransformasi menjadi distribusi yang seimbang dengan komposisi merata 33,3% untuk setiap kelas. Seperti terlihat pada Tabel 4, transformasi ini dicapai melalui kombinasi *random oversampling* yang meningkatkan sampel kelas minor seperti *smishing* dan spam, menjadi 2.712 sampel masing-masing, dan *random undersampling* yang mengurangi kelas mayoritas ham menjadi 2.712 sampel. Implementasi *embedding-level MixUp* dengan parameter alpha 0,4 dan ratio 0,4 efektif dalam memperluas variasi data, yang berkontribusi terhadap peningkatan *robustness* model. Hasil evaluasi menunjukkan bahwa strategi penyeimbangan dataset ini berhasil mempertahankan keragaman karakteristik pesan dan mengatasi bias klasifikasi yang disebabkan oleh ketidakseimbangan distribusi awal.

Tabel 4. Distribusi Data Setelah *Resampling*

Kelas	Jumlah Data
Ham	2.712
Smishing	2.712
Spam	2.712



Gambar 3. Distribusi Dataset Sebelum dan Sesudah Proses *Resampling*

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 2, Desember 2025
e-ISSN 2685-998X
DOI 10.30865/json.v7i2.9299

3.4 Optimasi *Hyperparameter*

Proses optimasi *hyperparameter* menggunakan *GridSearchCV* dengan validasi silang 5-*fold* berhasil mengidentifikasi konfigurasi optimal untuk ketiga model. Berdasarkan hasil yang ditunjukkan pada Tabel 5, model SVM linear mencapai skor F1-*macro* tertinggi sebesar 87,6%, diikuti oleh *Logistic Regression* 86,9% dan *Multinomial Naive Bayes* 86,8%. Perbedaan ini menunjukkan bahwa pendekatan berbasis margin seperti SVM mampu memberikan keseimbangan terbaik antara *precision* dan *recall* di seluruh kelas.

Tabel 5. Konfigurasi Optimal Hasil *GridSearchCV*

Model	Parameter Kunci	Nilai Optimal	F1-Macro (%)
Linear SVM	C=0.5, loss='squared_hinge', dual=False	Regulasi kuat	87,6
Logistic Regression	C=2.0, class_weight=None	Regulasi ringan	86,9
Multinomial NB	Alpha=0.1, fit_prior=True, min_df=3	Smoothing moderat	86,8

Sebagaimana terlihat pada Tabel 5, model *Linear* SVM menunjukkan konfigurasi terbaik dengan parameter regularisasi C = 0.5, fungsi *loss squared_hinge*, dan formulasi primal (*dual = False*). Kombinasi ini menghasilkan performa paling optimal karena regularisasi yang lebih kuat mampu menyeimbangkan antara kompleksitas model dan kemampuan generalisasi. Hasil ini mengonfirmasi bahwa SVM dengan kernel linear efektif dalam menangani data teks berdimensi tingi dan jarang *sparse* seperti representasi TF-IDF. Model *Logistic Regression* menempati posisi kedua dengan konfigurasi optimal pada C = 2.0 dan tanpa pembobotan kelas tambahan (*class_weight = None*). Kondisi ini menunjukkan bahwa mekanisme *hybrid resampling* yang diterapkan pada tahap prapemrosesan sudah cukup untuk menyeimbangkan distribusi antar kelas, sehingga tidak diperlukan lagi kompensasi melalui *class weighting*.
Sementara itu, model *Multinomial Naive Bayes* mencapai konfigurasi terbaik dengan α = 0.1, *fit_prior = True*, serta *min_df* = 3. Nilai *smoothing* moderat (α = 0.1) membantu menstabilkan estimasi probabilitas untuk token yang jarang muncul, sedangkan *fit_prior = True* memungkinkan model mempertimbangkan distribusi kelas asli. Nilai *min_df* yang relatif tinggi mengindikasikan sensitivitas model terhadap fitur berfrekuensi rendah, sehingga hanya mempertahankan token yang muncul dalam minimal tiga dokumen. Dari sudut pandang arsitektur algoritma, Linear SVM terbukti paling unggul dalam menangani karakteristik data teks berbasis TF-IDF *word n-gram* (1–3). Prinsip *maximum margin* yang mendasari SVM memungkinkan model membentuk *decision boundary* yang optimal untuk memisahkan ketiga kelas (*ham*, *smishing*, dan spam), sekaligus mempertahankan *balanced performance* di seluruh kategori.

3.5 Performa Klasifikasi Model

a. Analisis Efektivitas Hybrid Sampling pada Linear SVM

Untuk membuktikan efektivitas strategi *hybrid sampling* tiga tingkat, kinerja model Linear SVM dianalisis pada dua skenario, yaitu sebelum dan sesudah penerapan *framework hybrid sampling*. Hasil yang ditunjukkan pada Tabel 6 mengindikasikan adanya perubahan performa yang signifikan dan menggambarkan pengaruh metodologis yang substansial dari pendekatan tersebut terhadap peningkatan kemampuan model dalam menangani ketidakseimbangan data.

Tabel 6. Perbandingan Performa Klasifikasi Linear SVM Sebelum dan Sesudah Penerapan *Hybrid Sampling*

Model	*Acc* (%)	*Ham*			*Smishing*			Spam		
		Prec (%)	*Rec* (%)	*F1* (%)	*Prec* (%)	*Rec* (%)	*F1* (%)	*Prec* (%)	*Rec* (%)	*F1-Score* (%)
Linear SVM tanpa *resampling*	96,65	99,0	100	99,0	89,0	86,0	88,0	82,0	80,0	81,0
Linear SVM + ROS + RUS + MixUp	96,73	99,0	99,0	99,0	91,0	84,0	87,0	78,0	87,0	82,0

Seperti yang ditunjukkan pada Tabel 6, penerapan strategi *Hybrid Sampling*, khususnya melalui komponen *Embedding MixUp*, memberikan kontribusi penting dalam meningkatkan kemampuan model Linear SVM dalam mengenali *smishing* dan spam sebagai kelas minoritas. *Embedding MixUp* memperluas variasi representasi sintetis pada ruang *embedding* sehingga distribusi fitur kelas *smishing* yang semula terbatas menjadi lebih kaya dan informatif. Mekanisme ini membantu model mempelajari pola linguistik *smishing* yang halus dan sulit dibedakan

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 2, Desember 2025
e-ISSN 2685-998X
DOI 10.30865/json.v7i2.9299

dari pesan normal. Selain menambah jumlah sampel efektif, teknik ini menghasilkan transisi fitur yang lebih halus antar contoh sehingga model tidak terfokus pada pola kelas mayoritas dan sensitivitas terhadap karakteristik *smishing* dapat meningkat. Peningkatan *precison smishing* sebesar 2% menunjukkan bahwa model menjadi lebih tepat dalam mengidentifikasi pesan *smishing*, sementara penurunan *recall* sebesar 2% masih berada pada batas *trade-off* yang dapat diterima. Di sisi lain, peningkatan *recall* spam dari 80% menjadi 87% menegaskan bahwa *Hybrid Sampling* juga memperkuat kemampuan model dalam mengenali kelas lain yang sebelumnya lebih sulit terdeteksi. Sementara itu, performa pada kelas ham tetap stabil dengan nilai *precision*, *recall*, dan *F1-score* yang konsisten di angka 99%. Secara keseluruhan, *Hybrid Sampling* membantu model untuk tidak terlalu bergantung pada pola kelas mayoritas dan memberikan distribusi pembelajaran yang lebih seimbang, sehingga menghasilkan peningkatan performa pada kategori yang sebelumnya lebih sulit terdeteksi tanpa mengurangi akurasi keseluruhan.

b. Perbandingan Efektivitas Algoritma Klasifikasi Teks

Kinerja yang lebih seimbang pada Tabel 6 tersebut selaras dengan hasil optimasi *hyperparameter* yang telah dijabarkan sebelumnya. Parameter optimal yang diperoleh, seperti pengaturan regularisasi yang lebih kuat serta pemilihan formulasi primal, memungkinkan Linear SVM memanfaatkan representasi fitur hasil *Hybrid Sampling* dengan lebih efektif. Kombinasi antara struktur parameter yang telah dioptimalkan dan distribusi data yang lebih seimbang menghasilkan model yang mampu mempertahankan akurasi tinggi sekaligus meningkatkan sensitivitas terhadap kelas minoritas. Konsistensi ini menunjukkan bahwa konfigurasi Linear SVM yang dihasilkan dari proses tuning tidak hanya sesuai dengan karakteristik data teks yang berdimensi tinggi dan *sparse*, tetapi juga mendukung pencapaian performa klasifikasi akhir yang stabil sebagaimana terlihat pada Tabel 7.

Tabel 7. Perbandingan Performa Model

Algoritma	*Acc* (%)	Macro Avg.			Weighted Avg.		
		Prec (%)	*Rec* (%)	*F1* (%)	*Prec* (%)	*Rec* (%)	*F1* (%)
Linear SVM	96,7	90,0	90,0	90,0	97,0	97,0	97,0
Logistic Regression	96,0	88,0	89,0	88,0	96,0	96,0	96,0
Multinomial NB	96,3	89,0	89,0	89,0	97,0	96,0	96,0

Sebagaimana disajikan pada Tabel 7, model Linear SVM menunjukkan performa paling unggul dengan akurasi sebesar 96,7%, melampaui *Multinomial Naive Bayes* dengan akurasi 96,3% dan *Logistic Regression* dengan akurasi 96%. Keunggulan Linear SVM menjadi semakin jelas dan substansial ketika dianalisis menggunakan metrik F1-*score macro*, di mana model ini mencapai nilai 90%. Pencapaian ini mengonfirmasi bahwa keunggulan model ini tidak hanya pada akurasi secara umum, tetapi terletak pada kemampuannya yang unggul dalam menjaga keseimbangan antara *precision* dan *recall* secara merata di semua kelas, sehingga menjadikannya pilihan paling optimal. Nilai *weighted average* seluruh model berada di kisaran 96-97%, menandakan ketiganya andal menangani kelas mayoritas. Namun, SVM linear tetap paling konsisten dengan *weighted* F1 97%, mengindikasikan stabilitas yang lebih baik pada kelas minoritas. Secara praktis, selisih akurasi 0,4% terhadap *Multinomial* NB dan 0,7% terhadap *Logistic Regression* mungkin tampak kecil, tetapi pada sistem deteksi *smishing* berskala besar, perbedaan ini dapat bermakna dalam menekan *false negative* dan meningkatkan keamanan pengguna. Keunggulan SVM linear selaras dengan sifat algoritmanya yang mengoptimalkan *maximum margin* [26], sehingga lebih *robust* untuk data teks berdimensi tinggi dan *sparse* seperti representasi TF-IDF.

c. Analisis Detail Performa per Kelas

Tabel 8. Evaluasi *Precision*, *Recall*, dan *F1-Score* per Kelas

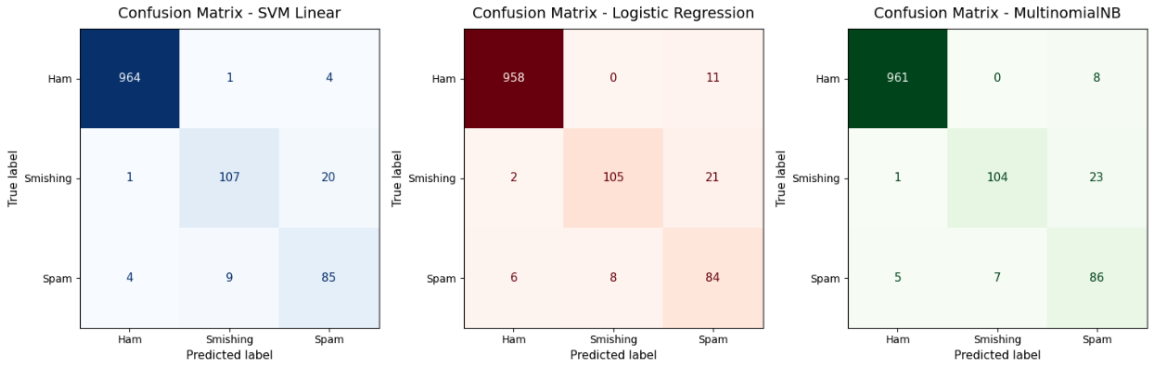
Model	Kelas	*Precision* (%)	*Recall* (%)	*F1-Score* (%)
SVM Linear	Ham	99,0	99,0	99,0
	Smishing	91,0	84,0	87,0
	Spam	78,0	87,0	82,0
Logistic Regression	Ham	99,0	99,0	99,0
	Smishing	93,0	82,0	87,0
	Spam	72,0	86,0	79,0
Multinomial NB	Ham	99,0	99,0	99,0
	Smishing	94,0	81,0	87,0
	Spam	74,0	88,0	80,0

Hasil pada Tabel 8 menunjukkan bahwa ketiga model memiliki kinerja sempurna pada kelas ham dengan nilai *precision*, *recall*, dan F1-*score* mencapai 99%. Pada kelas smishing, ketiganya memiliki F1-score yang setara di

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 2, Desember 2025
e-ISSN 2685-998X
DOI 10.30865/json.v7i2.9299

87%, namun SVM linear unggul dengan *recall* tertinggi 84% untuk menekan *false negative*, sekalipun memiliki *precision* 91% yang lebih rendah dari model lain. Di kelas spam, SVM linear kembali unggul dengan F1-score tertinggi 82%, sementara Multinomial NB hanya unggul dalam *recall*. Secara keseluruhan, model SVM linear menunjukkan keunggulan F1-*score* pada kelas *spam* dan *recall* tertinggi pada kelas *smishing*. Dengan demikian, hasil evaluasi menunjukkan bahwa model SVM linear lebih unggul dalam menangani kelas minoritas, sehingga penerapannya disarankan untuk sistem deteksi *smishing* dengan karakteristik data yang tidak seimbang.

d. Analisis *Confusion Matrix*



Gambar 4. *Confusion Matrix* hasil prediksi model klasifikasi pada data uji

Berdasarkan analisis *confusion matrix* pada Gambar 4, ketiga model menunjukkan kinerja yang sangat baik dalam mengklasifikasikan pesan *ham* sebagai kelas mayoritas. Linear SVM memperoleh *True Positive* tertinggi yaitu 964 dari 969 sampel, dengan hanya 4 kesalahan klasifikasi menjadi spam. *Logistic Regression* dan *Multinomial Naive Baye*s masing-masing menghasilkan 958 dan 961 prediksi benar, namun *Logistic Regression* melakukan 11 kesalahan klasifikasi *ham* menjadi spam, sedangkan *Naive Bayes* melakukan 8 kesalahan ke pesan spam. Pada kategori *smishing*, Linear SVM juga unggul dengan 107 prediksi benar dari 128 sampel, dengan 1 kesalahan ke *ham* dan terdapat 20 kesalahan ke spam, sedangkan *Logistic Regression* dan *Naive Bayes* mencatat performa sedikit lebih rendah.

Untuk kelas spam, *Multinomial Naive Baye*s menunjukkan *recall* tertinggi dengan 86 prediksi benar dari 98 sampel, diikuti SVM Linear dengan 85 benar dan Logistic Regression dengan 84 benar. Secara keseluruhan, Linear SVM unggul pada kelas *ham* dan *smishing*, sementara *Naive Bayes* lebih baik dalam mendeteksi spam, yang mencerminkan karakteristik algoritma masing-masing. SVM dengan kernel linear lebih efektif dalam memisahkan kelas pada data teks berdimensi tinggi melalui prinsip *maximum margin*, sedangkan *Naive Bayes* lebih sensitif terhadap token berfrekuensi tinggi yang umum pada pesan spam.

3.6 Deskripsi Umum dan Implikasi Hasil

Penelitian ini menunjukkan bahwa model berbasis Linear SVM dengan representasi TF-IDF *word n-gram* (1–3) dan tambahan fitur meta menghasilkan performa paling optimal dalam deteksi *smishing*, dengan akurasi 96,7% dan F1-macro 87,6%. Dibandingkan dengan penelitian-penelitian sebelumnya, capaian ini menunjukkan kinerja yang kompetitif bahkan sedikit lebih unggul. Sebagai contoh, penelitian sebelumnya yang dilakukan oleh Misra dan Soni [10] mengadopsi pendekatan *rule-based flowchart* melalui empat modul terintegrasi, yaitu *URL Filter*, *Source Code Analyzer*, *SMS Content Analyzer*, dan *APK Download Detector*, berhasil mencapai akurasi 96,29%. Sementara itu, penelitian lain oleh Alsufyani dan Alajmani [31] mengevaluasi model CNN, BiGRU, dan GRU untuk deteksi *phishing* SMS berbahasa Arab melaporkan bahwa GRU mencapai akurasi tertinggi sebesar 95,33%, diikuti BiGRU (94,42%) dan CNN (93,59%). Dengan akurasi 96,7%, model yang diusulkan dalam penelitian ini berhasil melampaui capaian model GRU dan setara dengan sistem *rule-bas*ed yang lebih kompleks. Keunggulan metodologis yang mendasari capaian ini mencakup kemampuan pendekatan yang diusulkan untuk menggabungkan representasi *linguistik* dan *non-linguistik* guna mempelajari pola khas pesan berbahaya tanpa ketergantungan pada analisis perilaku atau modul *eksternal* layaknya sistem *rule-based*, sekaligus menawarkan efisiensi dan efektivitas yang kompetitif dengan arsitektur *deep learning* yang lebih kompleks. Keunggulan utama lainnya terletak pada kemampuan model dalam menyelesaikan tugas klasifikasi multi-kelas yang lebih menantang, serta adaptabilitasnya terhadap variasi pola pesan baru tanpa memerlukan pembaruan aturan manual yang berkelanjutan.

Selain keunggulan metodologis, strategi *Hybrid Sampling* membantu menghasilkan distribusi prediksi yang lebih seimbang pada ketiga kelas. Hal ini tercermin dari penurunan *recall ham* dari 100% menjadi 99%, yang diikuti peningkatan *recall* pada kelas *smishing* dan spam. Perubahan ini menunjukkan bahwa model tidak lagi terlalu bias

Jurnal Sistem Komputer dan Informatika (JSON)
Volume 7, Nomor 2, Desember 2025
e-ISSN 2685-998X
DOI 10.30865/json.v7i2.9299

terhadap kelas mayoritas dan mampu mengenali lebih banyak sampel dari kelas minoritas tanpa mengurangi akurasi keseluruhan. Integrasi fitur TF-IDF dan *meta-features* seperti panjang teks, jumlah digit, serta keberadaan URL, alamat email, dan nomor telepon turut meningkatkan performa model dengan memberikan konteks struktural tambahan yang relevan terhadap karakteristik pesan *smishing*. Secara keseluruhan, penelitian ini menunjukkan bahwa SVM linear dengan kombinasi fitur linguistik dan *meta-features* yang dioptimasi melalui *hybrid sampling* dapat menjadi alternatif yang efisien dan adaptif dibandingkan pendekatan *rule-based*, karena mampu mempertahankan tingkat akurasi yang tinggi, menangani klasifikasi multi-kelas yang lebih kompleks, dan menawarkan fleksibilitas yang lebih baik dalam menghadapi variasi serangan *smishing* di masa depan.
Meskipun mencapai akurasi tinggi sebesar 96,7%, penelitian ini masih menunjukkan keterbatasan dalam kesenjangan performa antar kelas, dengan F1-score kelas *smishing* dan spam masing-masing hanya 87% dan 82%. Hal ini mengindikasikan bahwa pendekatan berbasis TF-IDF dan fitur meta belum optimal dalam menangkap konteks semantik mendalam serta membedakan *smishing* dan spam yang memiliki karakteristik tumpang tindih. Untuk penelitian selanjutnya, direkomendasikan pengembangan model berbasis *embedding* kontekstual seperti BERT atau Bi-LSTM guna meningkatkan pemahaman semantik dan kemampuan generalisasi pada data *real-world*. Eksplorasi teknik data augmentasi yang lebih canggih juga diperlukan untuk mengatasi ketidakseimbangan dataset tanpa mengorbankan *robustnes* model terhadap variasi ancaman baru.

4. KESIMPULAN

Penelitian ini berhasil mencapai tujuannya untuk membuktikan sinergi antara algoritma Linear SVM dan strategi *hybrid sampling* tiga tingkat dalam mengatasi masalah ketidakseimbangan data pada deteksi *smishing* multi-kelas. Hasil evaluasi menunjukkan pencapaian tertinggi diperoleh dari model Linear SVM teroptimasi (C=0,5) yang berhasil meraih akurasi 96,7% dan F1-*macro* 87,6% pada data uji. Kombinasi ini memungkinkan Linear SVM untuk membentuk *decision boundary* yang lebih optimal berkat distribusi data yang telah diseimbangkan oleh strategi *hybrid*. Aspek paling signifikan terletak pada kemampuan *framework* dalam menyeimbangkan performa deteksi dengan konsisten di semua kategori, termasuk recall 99% pada kelas ham serta peningkatan sensitivitas terhadap smishing dan spam sebagai kelas minoritas. Hasil ini membuktikan keefektifan pendekatan *hybrid sampling* dalam mengatasi bias klasifikasi pada lingkungan data tidak seimbang. Dari perspektif implementasi, penelitian ini berhasil memberikan kontribusi substantif melalui penyediaan solusi komputasional yang efisien untuk masalah deteksi *smishing* multi-kelas. Kerangka kerja yang dikembangkan terbukti tidak hanya mengatasi keterbatasan metode konvensional dalam menangani ketidakseimbangan distribusi kelas, melainkan juga menawarkan arsitektur yang dapat beradaptasi dengan perkembangan teknik *smishing* di masa depan. Temuan ini membuka peluang integrasi sistem dalam *platform* keamanan siber yang sudah ada sekaligus menetapkan standar baru dalam penanganan data tidak seimbang untuk tugas klasifikasi teks dalam konteks keamanan digital. Nilai strategis penelitian ini semakin ditegaskan melalui kemampuannya menyajikan solusi *machine learning* murni yang setara dengan sistem *rule-based* kompleks, sehingga menawarkan alternatif implementasi yang mudah dikembangkan untuk kebutuhan deteksi *smishing* secara *real-time*. Namun demikian, penelitian ini masih memiliki keterbatasan dalam kesenjangan performa antar kelas, dengan F1-score kelas smishing dan spam masing-masing hanya 87% dan 82%. Hal ini mengindikasikan bahwa pendekatan berbasis TF-IDF dan fitur meta belum optimal dalam menangkap konteks semantik mendalam serta membedakan smishing dan spam yang memiliki karakteristik tumpang tindih. Sebagai arah pengembangan lebih lanjut, penelitian ini dapat diperkuat melalui pemanfaatan model berbasis embedding kontekstual seperti BERT atau Bi-LSTM guna meningkatkan pemahaman semantik pada pesan smishing. Selain itu, eksplorasis teknik augmentasi data yang lebih adaptif juga diperlukan untuk meningkatkan kemampuan generalisasi model dalam menghadapi pola serangan yang terus berkembang di skenario nyata.

REFERENCES

[1] A. F. Mahmud and S. Wirawan, “Phishing Website Detection Using Machine Learning Classification Method,” *SISTEMASI*, vol. 13, no. 4, pp. 1368–1380, 2024, doi: 10.32520/stmsi.v13i4.3456.
[2] G. Tanbhir, M. F. Shahriyar, K. Shahed, A. M. R. Chy, and M. Al Adnan, “Hybrid Machine Learning Model for Detecting Bangla Smishing Text Using BERT and Character-Level CNN,” in *13th International Conference on Electrical and Computer Engineering (ICECE)*, 2024, pp. 57–62. doi: 10.1109/ICECE64886.2024.11024872.
[3] Slamet, “Smishing Guard: Strategi Pengembangan Sistem Deteksi dan Respons Ancaman SMS Phishing,” *SPIRIT*, vol. 17, no. 1, pp. 12–23, 2024, doi: 10.53567/spirit.v17i1.380.
[4] S. Hosseinpour and S. Das, “POSTER: A Multi-Signal Model for Detecting Evasive Smishing,” in *Proceedings of the 18th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, 2025, pp. 292–293. doi: 10.1145/3734477.3736147.
[5] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*, 2nd ed. Cambridge: MIT Press, 2018.

[6] J. Schmidt *et al.*, "Improving Machine-Learning Models in Materials Science through Large Datasets," *Mater. Today Phys.*, vol. 48, no. September, p. 101560, 2024, doi: 10.1016/j.mtphys.2024.101560.
[7] A. H. Salem, S. M. Azzam, O. E. Emam, and A. A. Abohany, "Advancing Cybersecurity: A Comprehensive Review of AI-Driven Detection Technique," *J. Big Data*, vol. 11, no. 105, 2024, doi: 10.1186/s40537-024-00957-y.
[8] S. W. Irianandа, R. W. Budiawan, A. Y. Rahman, and I. Istiadi, "Optimasi Klasifikasi Sentimen Komentar Pengguna Game Bergerak Menggunakan SVM, Grid Search dan Kombinasi N-Gram," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 11, no. 4, pp. 743–752, 2024, doi: 10.25126/jtiik.1148244.
[9] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A Comprehensive Survey on Support Vector Machine Classification: Applications, Challenges and Trends," *Neurocomputing*, vol. 408, pp. 189–215, 2020, doi: 10.1016/j.neucom.2019.10.118.
[10] S. Mishra and D. Soni, "Smishing Detector: A Security Model to Detect Smishing through SMS Content Analysis and URL Behavior Analysis," *Futur. Gener. Comput. Syst.*, vol. 108, pp. 803–815, 2020, doi: 10.1016/j.future.2020.03.021.
[11] M. Alshinwan, O. A. Khashan, Z. Alarnaout, S. S. Shreem, A. Y. Shdefat, and N. A. Karim, "A Novel Smishing Defense Approach Based on Meta-Heuristic Optimization Algorithms," *Cybersecurity*, vol. 8, no. 1, pp. 8–35, 2025, doi: 10.1186/s42400-024-00328-3.
[12] P. Sun, Z. Wang, L. Jia, and Z. Xu, "SMOTE-kTLNN: A Hybrid Re-sampling Method Based on SMOTE and a Two-Layer Nearest Neighbor Classifier," *Expert Syst. Appl.*, vol. 238, p. 121848, 2023, doi: 10.1016/j.eswa.2023.121848.
[13] H. I. Hussein, S. A. Anwar, and M. I. Ahmad, "Imbalanced Data Classification Using SVM Based on Improved Simulated Annealing Featuring Synthetic Data Generation and Reduction," *Comput. Mater. Contin.*, vol. 75, no. 1, pp. 547–564, 2023, doi: 10.32604/cmc.2023.036025.
[14] A. Salehi and M. Khedmati, "Hybrid Clustering Strategies for Effective Oversampling and Undersampling in Multiclass Classification," *Sci. Rep.*, vol. 15, p. 3460, 2025, doi: 10.1038/s41598-024-84786-2.
[15] A. Ahmad, O. Chaudhari, and R. Chandra, "A Review of Ensemble Learning and Data Augmentation Models for Class Imbalanced Problems : Combination , Implementation and Evaluation," *Expert Syst. Appl.*, vol. 244, p. 122778, 2024, doi: 10.1016/j.eswa.2023.122778.
[16] R. Asyrofi, "Synthetic-MixUp : A Simple Framework for Imbalanced Text Classification," *2023 IEEE 12th Glob. Conf. Consum. Electron.*, pp. 927–929, 2023, doi: 10.1109/GCCE59613.2023.10315313.
[17] H. Sun *et al.*, "Reliable Object Tracking by Multimodal Hybrid Feature Extraction and Transformer-Based Fusion," *Neural Networks*, vol. 178, no. February, pp. 1–12, 2024, doi: 10.1016/j.neunet.2024.106493.
[18] M. Liang and T. Niu, "Research on Text Classification Techniques Based on Improved TF-IDF Algorithm and LSTM Inputs," *Procedia Comput. Sci.*, vol. 208, pp. 460–470, 2022, doi: 10.1016/j.procs.2022.10.064.
[19] L. B. V. de Amorim, G. D. C. Cavalcanti, and R. M. O. Cruz, "The Choice of Scaling Technique Matters for Classification Performance," *Appl. Soft Comput.*, vol. 133, p. 109924, 2023, doi: 10.1016/j.asoc.2022.109924.
[20] G. Kou, H. Chen, and M. A. Hefni, "Improved Hybrid Resampling and Ensemble Model for Imbalance Learning and Credit Evaluation," *J. Manag. Sci. Eng.*, vol. 7, no. 4, pp. 511–529, 2022, doi: 10.1016/j.jmse.2022.06.002.
[21] C. N. Mohammed and A. M. Ahmed, "A Semantic-Based Model With a Hybrid Feature Engineering Process for Accurate Spam Detection," *J. Electr. Syst. Inf. Technol.*, vol. 11, p. 26, 2024, doi: 10.1186/s43067-024-00151-3.
[22] B. Li, Y. Hou, and W. Che, "Data Augmentation Approaches in Natural Language Processing: A Survey," *AI Open*, vol. 3, pp. 71–90, 2022, doi: 10.1016/j.aiopen.2022.03.001.
[23] M. C. Untoro and M. A. N. M. Yusuf, "Evaluate of Random Undersampling Method and Majority Weighted Minority Oversampling Technique in Resolve Imbalanced Dataset," *IT J. Res. Dev.*, vol. 8, no. 1, pp. 1–13, 2023, doi: 10.25299/itjrd.2023.12412.
[24] D. S. Cross-validation, "A Comparative Study of the Use of Stratified Cross-Validation and Distribution-Balanced Stratified Cross-Validation in Imbalanced Learning," vol. 23, no. 4, p. 2333, 2023, doi: 10.3390/s23042333.
[25] H. Wang and Y. Shao, "Sparse and Robust SVM Classifier for Large Scale Classification," *Appl. Intell.*, vol. 53, no. 16, pp. 19647–19671, 2023, doi: 10.1007/s10489-023-04511-w.
[26] M. Mujahid *et al.*, "Data Oversampling and Imbalanced Datasets: an Investigation of Performance for Machine Learning and Feature Engineering," *J. Big Data*, vol. 11, p. 87, 2024, doi: 10.1186/s40537-024-00943-4.
[27] S. Al Hasan *et al.*, "Classification of Multi-Labeled Text Articles with Reuters Dataset using SVM," in *International Conference on Science and Technology (ICOSTECH)*, 2022, pp. 1–5. doi: 10.1109/ICOSTECH54296.2022.9829153.
[28] M. Soni, *Artificial Intelligence*. India: Poorav Publications, 2024.
[29] Q. Li, S. Zhao, S. Zhao, and J. Wen, "Logistic Regression Matching Pursuit Algorithm for Text Classification," *Knowledge-Based Syst.*, vol. 277, p. 110761, 2023, doi: 10.1016/j.knosys.2023.110761.
[30] L. Zhang, "Features Extraction Based on Naive Bayes Algorithm and TF-IDF for news classification," *PLoS One*, vol. 20, no. 7, p. e0327347, 2025, doi: 10.1371/journal.pone.0327347.
[31] S. Alsufyani and S. Alajmani, "A Deep Learning for Arabic SMS Phishing Based on URLs Detection," *Int. J. Adv. Comput. Sci. Appl.*, vol. 16, no. 1, pp. 388–396, 2025, doi: 10.14569/IJACSA.2025.0160138.