Terjemahan artikel 10.22146/jnteti.v14i4.20931

Optimasi Klasifikasi *Email* Spam Menggunakan *Naïve Bayes* Berbasis NLP pada TF-IDF dan SMOTE

Andi Maslan[1], Azan Rahman[1], Umar Faruq[2], Rabei Raad Ali Al-Jawry[3]

[1] Program Studi Teknik Informatika, Fakultas Teknik dan Komputer, Universitas Putera Batam, Batam, Kepulauan Riau, 2943, Indonesia
[2] School of Information Technology, UNITAR International University Petaling Jaya, Selangor, Malaysia
[3] Department of Computer Engineering Technology, Northern Technical University, Mosul, Iraq

[Diserahkan: 10 Juli 2025, Direvisi: 9 Oktober 2025, Diterima: 27 Oktober 2025]
Penulis Korespondensi: Andi Maslan (email: lanmasco@gmail.com

INTISARI — Kemajuan pesat teknologi informasi dan komunikasi telah mengubah cara manusia berinteraksi dan bertukar informasi. Salah satu media komunikasi digital yang paling banyak digunakan adalah *email* (surel). Namun, *email* sering kali disalahgunakan untuk mengirim pesan spam yang berisi tautan *phishing*, *malware*, atau iklan yang tidak diminta, sehingga berpotensi menimbulkan risiko yang serius bagi individu maupun organisasi. Oleh karena itu, pengembangan metode deteksi spam yang akurat dan efisien menjadi makin penting. Studi ini mengusulkan pendekatan klasifikasi *email* spam yang ringan dan efisien menggunakan algoritma *naïve Bayes* yang dikombinasikan dengan ekstraksi fitur *term frequency-inverse document frequency* (TF-IDF) dan *synthetic minority oversampling technique* (SMOTE) untuk mengatasi ketidakseimbangan kelas. Serangkaian langkah prapemrosesan—tokenisasi, *lemmatization*, penghapusan *stopword*, dan TF-IDF—diterapkan untuk melakukan normalisasi dan vektorisasi data teks *email*. Teknik SMOTE diterapkan pada *dataset* pelatihan untuk menyeimbangkan distribusi kelas dan menghindari kebocoran data selama evaluasi. Hasil penelitian menunjukkan bahwa model *naïve Bayes* awalnya mencapai akurasi 88%, *recall* 86%, presisi 100%, dan skor F1 92%. Setelah penerapan SMOTE, model tersebut mencapai akurasi, presisi, *recall*, dan skor F1 100%, yang menunjukkan klasifikasi sempurna *email* spam dan bukan spam. Hasil ini menegaskan bahwa penyeimbangan kelas yang tepat secara signifikan meningkatkan kemampuan model untuk mendeteksi *email* spam. Secara keseluruhan, studi ini menyoroti efektivitas penggabungan TF-IDF, *naïve Bayes*, dan SMOTE sebagai solusi yang tangguh, tetapi efisien secara komputasi untuk deteksi spam modern, khususnya cocok untuk lingkungan waktu nyata dan dengan keterbatasan sumber daya.

KATA KUNCI — Tokenisasi, SMOTE, TF-IDF, Deteksi Spam *Email*, Optimasi Kinerja Model, Peningkatan Akurasi Klasifikasi.

I. PENDAHULUAN

Seiring dengan perkembangan teknologi digital saat ini, *email* menjadi salah satu sarana komunikasi yang umum digunakan, baik untuk urusan pribadi maupun bisnis. Dibandingkan dengan pesan yang dikirimkan menggunakan metode konvensional seperti surat, *email* jauh lebih cepat dan efisien karena dapat menghemat waktu dan uang. Selain mengirim pesan teks, *email* memungkinkan penggunanya mengirim lampiran seperti gambar dan dokumen. Namun, popularitas *email* yang makin meningkat sebagai sarana komunikasi juga menimbulkan permasalahan baru berupa spam. Kemudahan *email* juga dimanfaatkan pihak-pihak yang tidak bertanggung jawab untuk keuntungan pribadi. *Email* spam ini dapat berisi iklan, promosi produk, penipuan, atau konten yang tidak diinginkan. Terkadang *email* spam juga disertai tautan berbahaya yang jika diakses dapat menyebarkan virus atau *malware*. Selain mengganggu korban secara pribadi, *email* spam dapat memenuhi penyimpanan perangkat dan meningkatkan lalu lintas data di jaringan internet.

Berdasarkan laporan dari berbagai sumber, perkembangan satu tahun terakhir menunjukkan bahwa lanskap *phising* terus berkembang sejak State of Phishing Report terakhir, dengan beberapa tren menunjukkan peningkatan yang signifikan dan mengkhawatirkan [1]–[4]. Selain itu, dalam enam bulan terakhir, SlashNext Threat Labs mencatat adanya peningkatan *email phishing* sebesar 1.265%, dengan 68% di antaranya menggunakan taktik *business email compromise* (BEC) berbasis teks, menunjukkan bahwa *chatbot* dan teknik *jailbreak* digunakan untuk membuat serangan yang lebih cepat dan lebih realistis. *Phishing* kredensial juga meningkat sebesar 967%, terutama disebabkan oleh *ransomware* yang berupaya mengakses akun perusahaan. Selain ancaman *email*, serangan pada perangkat seluler juga meningkat, dengan 39% ancaman seluler berupa *smishing*, menunjukkan pergeseran ke arah serangan multisaluran yang makin kompleks [5], [6]. Oleh karena itu, para peneliti terus berinovasi untuk mencegah atau mendeteksi serangan melalui penelitian dengan berbagai pendekatan, seperti pembelajaran mesin atau kecerdasan buatan (*artificial intelligence*, AI).

Dengan menggunakan algoritma dan model komputasi yang dapat mempelajari dan mendeteksi pola data, pembelajaran mesin—sebuah subbidang AI—dapat digunakan untuk menilai dan mengidentifikasi ancaman keamanan serta mengotomatiskan solusi untuk mengatasi kesulitan keamanan dalam jaringan komputer [7], [8]. Tujuan utama pembelajaran mesin adalah mengubah beragam data menjadi keputusan atau tindakan yang efektif dengan sesedikit mungkin intervensi manusia [9]. Pembelajaran mesin bekerja dengan mempelajari pola yang ditemukan dalam *email* spam berdasarkan fitur-fiturnya, seperti frekuensi kalimat *email*, subjek *email*, lampiran, dan teks pesan yang diduga sebagai spam [10].

Terdapat beberapa jenis algoritma [11], [12] yang digunakan dalam pembelajaran mesin, salah satunya adalah algoritma *naïve Bayes* [13], yang menjadi fokus penelitian ini. *Naïve Bayes* memiliki keunggulan dalam mengklasifikasikan teks dan telah banyak diterapkan, termasuk dalam mendeteksi

email spam [14]. Algoritma ini menunjukkan kinerja yang baik, ringan dan cepat. Algoritma ini bekerja dengan menganalisis karakteristik tertentu dari sebuah *email*, seperti keberadaan iklan, promosi produk, penipuan, atau konten berbahaya, untuk menentukan *email* tersebut aman atau berisiko [15].

Berdasarkan landasan teoretis dan penelitian sebelumnya, studi ini menawarkan kebaruan dalam penerapan algoritma *naïve Bayes* untuk mendeteksi spam *email*, dengan mempertimbangkan perkembangan terbaru dalam ancaman siber. Salah satu perkembangan tersebut adalah penggunaan teknologi AI generatif, seperti ChatGPT, yang mulai digunakan dalam menciptakan konten *email* berbahaya yang lebih meyakinkan dan kompleks. Berbeda dengan pendekatan sebelumnya yang hanya mengandalkan *dataset* lama dan pola statis, studi ini berfokus pada karakteristik spam modern yang lebih canggih, seperti penggunaan bahasa alami, pengaburan konten promosi, dan penggunaan teknologi AI untuk mengelabui filter tradisional. Dari segi kontribusi ilmiah, studi ini menyajikan pengembangan dan evaluasi model klasifikasi berbasis *naïve Bayes* yang tidak hanya akurat, tetapi juga ringan dan cepat, sehingga cocok untuk digunakan dalam sistem waktu nyata atau pada perangkat dengan sumber daya terbatas. Studi ini juga membandingkan efektivitas deteksi terhadap data *email* berbahaya terbaru, sehingga memberikan wawasan penting untuk pengembangan sistem keamanan adaptif. Manfaat nyata dari penelitian ini dirasakan langsung oleh masyarakat luas, terutama dalam meningkatkan keamanan digital bagi pengguna *email* individu maupun institusi. Sistem yang dikembangkan dapat meminimalkan risiko penipuan, *malware*, dan konten yang tidak diinginkan, serta menjaga efisiensi komunikasi digital. Selain itu, model ini dapat dengan mudah diadopsi oleh usaha kecil, lembaga pendidikan, dan instansi pemerintah yang belum memiliki infrastruktur keamanan siber yang kompleks, sehingga meningkatkan perlindungan tanpa memerlukan investasi besar dalam teknologi.

II. PENELITIAN TERKAIT

Penelitian terkait klasifikasi *email* spam telah dilakukan secara ekstensif. Penelitian sebelumnya menjelaskan bahwa metode *grey wolf optimization-bidirectional encoder representations from transformers* (GWO-BERT) dengan *convolutional neural network* (CNN*), bidirectional long short-term memory* (BiLSTM), dan model LSTM berhasil mempelajari representasi teks *email* yang bermakna dan mengklasifikasikannya ke dalam kategori spam dengan tingkat akurasi 99,14% [16]. Kemudian, sebuah penelitian mengidentifikasi spam menggunakan algoritma *Harris Hawks optimization* (HHO) dengan pendekatan pembelajaran mesin, sehingga dapat mendeteksi fitur-fitur penting yang membedakan spam dari *email* palsu dengan hasil klasifikasi untuk algoritma *decision tree* sebesar 99,75%, AdaBoost 99,67%, dan *naïve Bayes* 96,30% [17]. Hasil penelitian lainnya menunjukkan efektivitas CNN, yang signifikan dalam mengklasifikasikan *email*, dengan tingkat akurasi tinggi sebesar 99,67% untuk data pengujian 20%, 99,64% untuk data pengujian 30%, dan 99,63% untuk data pengujian 40% [11].

Selain itu, penelitian sebelumnya berfokus pada penggunaan *term frequency-inverse document frequency* (TF-IDF) berbasis kata henti dan algoritma *stemming* dengan pengklasifikasi *naïve Bayes* dari *graphics processing unit* (GPU) *multicore* untuk mengidentifikasi *email* spam [18]. Hasilnya menunjukkan bahwa GPU NVIDIA P100 dapat mempercepat proses pelatihan dan pengujian sekaligus mencapai akurasi yang lebih tinggi dibandingkan dengan algoritma *naïve Bayes* konvensional. Penelitian tersebut melaporkan akurasi pelatihan sebesar 99,67% dan akurasi pengujian sebesar 99,03%. Waktu pelatihan pada GPU adalah 1.361 s, sedangkan pada CPU adalah 2.029 s. Sementara itu, hasil pengujian pada GPU adalah 1.978 s dan pada CPU adalah 2.280 s. Penelitian lain mengevaluasi model pengklasifikasi *naïve Bayes* untuk deteksi *email* spam, yang hasilnya menunjukkan bahwa algoritma ini dapat menjadi pilihan yang baik untuk deteksi *email* spam [19]. Akan tetapi, penelitian lebih lanjut diperlukan untuk mengatasi tantangan, seperti evolusi teknik *spammer* dan ketidakseimbangan data [13].

Selain itu, penelitian tentang deteksi *email* spam juga telah dilakukan. Penelitian sebelumnya melaporkan bahwa *naïve Bayes* efektif dalam mengklasifikasikan *email* spam, dengan kinerja terbaik menggunakan *k-fold cross-validation* dengan $k = 9$ [20]. Hasil penelitian ini menunjukkan bahwa *naïve Bayes* menghasilkan akurasi sebesar 84,8%, dengan 3.903 *email* diklasifikasikan benar dan 698 salah. Sementara itu, presisi dan *recall* masing-masing sebesar 0,86 dan 0,85. Temuan terbaru menunjukkan bahwa kinerja algoritma *naïve Bayes* dengan *chi-square* untuk akurasi, presisi, *recall*, dan skor F1 berada pada angka 98,83%. Hasil ini mengindikasikan bahwa *naïve Bayes* yang dikombinasikan dengan *chi-square* efektif untuk klasifikasi *email* spam, dengan presisi tinggi yang menghindari kesalahan positif [21]–[23]. Namun demikian, nilai *recall* masih perlu ditingkatkan agar dapat lebih banyak mendeteksi spam. Studi ini berfokus pada klasifikasi *email* spam menggunakan algoritma *naïve Bayes* berbasis *natural language processing* (NLP) dengan TF-IDF dan SMOTE, serta mempertimbangkan isu-isu dan penelitian lain yang telah disebutkan.

III. METODOLOGI

Penelitian ini dimulai dengan tahap identifikasi masalah utama, yaitu klasifikasi *email* yang sah (*ham*) dan spam menggunakan algoritma *naïve Bayes* [15]. Tahap selanjutnya adalah pengumpulan data melalui studi literatur terkait *email* spam, *machine learning*, dan metode *naïve Bayes*, dengan sumber dari buku dan jurnal nasional maupun internasional. Data yang telah dikumpulkan kemudian diproses untuk memastikan bahwa model dapat menggunakan formatnya. Setelah pemrosesan data, data pelatihan digunakan untuk melatih model *naïve Bayes*. Berdasarkan pola yang ditemukan dalam data pelatihan, model tersebut belajar untuk membedakan antara spam dan *ham*.

Setelah model dilatih, pengujian dilakukan menggunakan data pengujian yang tidak terlihat oleh model selama pelatihan. Hasil prediksi dari model kemudian dibandingkan dengan label asli untuk melihat kemampuan model dalam mengklasifikasikan *email* sebagai spam atau bukan spam (*ham*). Hasil pengujian dievaluasi menggunakan metrik seperti akurasi, presisi, *recall*, dan skor F1 untuk menilai kinerja model dalam mengklasifikasikan data dengan benar. Berdasarkan evaluasi yang telah dilakukan, kinerja model dianalisis untuk menentukan kemampuannya dalam membedakan spam dan *ham*. Jika kinerja model belum memuaskan, perbaikan dapat dilakukan pada proses pelatihan atau pemrosesan data.

Dalam penelitian ini, data sekunder digunakan dan diambil dari *dataset* publik. Data ini diperoleh dari sumber yang dapat diandalkan yang menyediakan *dataset* untuk penelitian dan tidak dihasilkan melalui pengumpulan primer, seperti

wawancara atau observasi. *Dataset* yang digunakan dalam penelitian ini diunduh dari Harvard Dataverse [24], sebuah platform penyimpanan data terbuka yang menyediakan berbagai macam *dataset* berkualitas untuk analisis dan pengembangan lebih lanjut.

Dataset ini terdiri atas 5.728 baris data. Setiap baris mewakili sebuah *email* yang dibagi menjadi dua kategori utama, yaitu spam dan *ham*, dengan 1.369 data spam dan 4.329 data *ham*. Data ini disajikan dalam format tabel dengan dua kolom utama: kolom "text" yang berisi teks *email*, dan kolom "spam" yang berisi label kategori, dalam bentuk "spam" atau "ham."

Dataset ini digunakan untuk melatih dan menguji model deteksi spam berbasis algoritma *naïve Bayes*. Sebelum digunakan, data mentah ini melalui tahap prapemrosesan, seperti penghapusan *stopword*, *lemmatization*, dan konversi teks ke dalam representasi numerik menggunakan metode TF-IDF, sehingga dapat digunakan dalam proses analisis lebih lanjut.

Proses diawali dengan pengumpulan *dataset email* berlabel yang mencakup pesan spam modern, seperti *phishing* dan konten yang dihasilkan AI. Tahap selanjutnya adalah tokenisasi, yang merupakan proses awal prapemrosesan teks yang bertujuan untuk membagi teks atau kalimat menjadi bagian-bagian yang lebih kecil, yaitu token [10], [16], [25]. Tokenisasi memastikan bahwa teks sudah dipisahkan menjadi unit-unit kecil (kata atau token). Token ini umumnya berupa kata atau simbol yang dapat digunakan dalam analisis lebih lanjut [26]. Proses tokenisasi sangat penting karena memungkinkan teks untuk dipecah menjadi unit-unit yang lebih mudah dianalisis, terutama dalam algoritma klasifikasi seperti *naïve Bayes*. Langkah-langkah *coking* adalah mengambil teks mentah dan memecah kalimat menjadi token.

Pada tahap pengambilan teks mentah, data yang digunakan terdiri atas teks *email* yang dapat berisi kalimat, simbol, dan tanda baca. Sebelum proses tokenisasi, teks-teks ini berbentuk kalimat utuh tanpa pemisah kata yang jelas. Selanjutnya, kalimat-kalimat ini dipecah menjadi kata-kata atau token, misalnya kalimat "*Congratulations! You have won a free vacation*" diubah menjadi ["*Congratulations*," "*You*," "*have*," "*won*," "*a*," "*free*," "*vacation*"]. Proses tokenisasi dilakukan menggunakan pustaka natural language toolkit (NLTK) di Python, yang menyediakan berbagai fungsi untuk tokenisasi teks. Fungsi tokenisasi kata di NLTK digunakan untuk memecah kalimat menjadi kata-kata individual.

Selanjutnya, teknik *lemmatization* digunakan untuk mengubah kata menjadi bentuk dasarnya [10], [16], [27]. Tidak seperti *stemming* yang hanya memotong bagian akhir kata, *lemmatization* mempertimbangkan konteks dan tata bahasa, yang penting untuk mengurangi variasi kata yang tidak perlu. Oleh karena itu, analisis teks lebih fokus pada makna kata-kata yang relevan. Sebelum proses *lemmatization*, teks tokenisasi yang diperoleh dari langkah sebelumnya digunakan [27]. *Lemmatization* diterapkan pada setiap token menggunakan alat atau pustaka tertentu [28]. Dalam penelitian ini, WordNetLemmatizer dari pustaka NLTK digunakan untuk melakukan *lemmatization* [29], [30]. Contoh hasil pada tahap ini adalah, "*running*" diubah menjadi "*run*," "*better*" diubah menjadi "*good*," dan kata "*coating*" diubah menjadi "*paint*."

Tahap selanjutnya adalah penghapusan *stopword*. Tahap ini merupakan proses penghapusan kata-kata yang dianggap tidak penting dalam analisis teks karena kata-kata ini umumnya tidak memberikan informasi yang relevan untuk tujuan klasifikasi [19], [31], [32], [23]. Kata-kata tersebut meliputi konjungsi, preposisi, atau kata kerja bantu yang sering muncul dalam kalimat tetapi tidak berkontribusi untuk membedakan kategori teks. Kata-kata seperti "*and*," "*or*," "*the*," "*for*," dan "*is*" termasuk dalam kategori *stopword*. Penghapusan *stopword* dilakukan dengan menggunakan daftar *stopword* yang sudah ada, seperti yang disediakan oleh pustaka NLTK di Python. Daftar ini sudah mencakup kata-kata umum yang ada dalam bahasa Inggris (dan bahasa lain) yang umumnya dianggap tidak relevan dalam analisis teks.

TF-IDF memberikan bobot pada setiap kata pada dokumen sesuai dengan frekuensi kemunculannya (juga dikenal sebagai *term frequency* atau TF) dan frekuensi kemunculannya yang jarang (juga dikenal sebagai *inverse document frequency* atau IDF) [13], [15], [23], [33], [34]. Contoh implementasinya disajikan pada Tabel I.

Tabel I menunjukkan contoh teks *email* yang dikategorikan sebagai spam dan *ham*. *Email* dalam kategori spam sering berisi konten promosi atau ajakan untuk mengklik tautan yang mencurigakan. Misalnya, baris subjek "*You've won $5,000!*" mendorong pembaca untuk mengklik tautan untuk mengklaim hadiah, yang merupakan ciri umum spam. Demikian pula, pesan "*Free Gift Card Offer*" berisi insentif dan urgensi untuk menyelesaikan survei, yang juga merupakan ciri khas pola spam. Sementara itu, *email* dalam kategori *ham* memberikan konten informatif dan transaksional. Misalnya, "*Meeting Agenda for Monday*" berisi detail yang relevan dengan kegiatan kerja yang dijadwalkan, dan "*Your Receipt from ABC Store*" memberikan konfirmasi pembelian, yang keduanya mencerminkan komunikasi yang sah.

Perhitungan jumlah kata dalam setiap isi *email* adalah sebagai berikut.

- Teks 1 (SPAM): "*Congratulations! You are the lucky winner of our monthly sweepstakes. Click the link below to claim now!*" berisi 17 kata.
- Teks 2 (SPAM): "*Complete a short survey and receive a $100 gift card instantly. Limited time only!*" terdiri atas 14 kata.
- Teks 3 (NOT SPAM): "*Hi team, please find attached the agenda for our project meeting scheduled on Monday at 10 AM*" terdiri atas 19 kata.
- Teks 4 (NOT SPAM): "*Thank you for your purchase. Attached is your receipt for the items bought on July 5*th" terdiri atas 17 kata.

Berdasarkan data teks tersebut, formula berikut digunakan untuk menghitung nilai TF.

$$TF(t) \quad \frac{jumlah\ kata\ (t)\ dalam\ dokumen}{jumlah\ kata\ dalam\ dokumen} \tag{1}$$

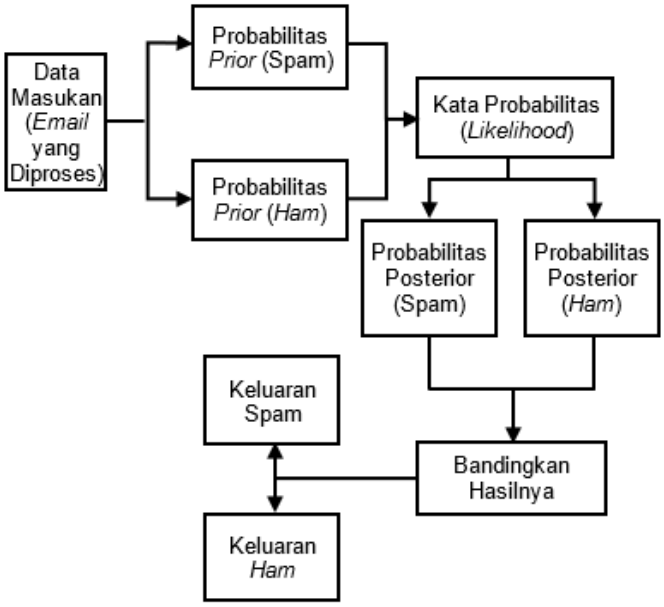
$$IDF(t) = Log(\frac{Total\ Dokumen}{Dokumen\ berisi\ kata\ t}) \tag{2}$$

$$TF - IDF(t,d) = TF(t,d) \times IDF(t). \tag{3}$$

Setelah memperoleh nilai TF, model klasifikasi *email* spam dibangun dan diuji menggunakan prosedur pelatihan dan pengujian. Untuk memastikan bahwa model mampu mempelajari data yang ada serta diuji menggunakan data yang belum pernah digunakan sebelumnya, *dataset* dipisahkan menjadi dua bagian utama: data pelatihan dan data pengujian. *Dataset* didistribusikan menggunakan metode validasi *hold-out*, dengan rasio data pelatihan 80% dan rasio data pengujian 20%. Rasio ini dipilih agar model memperoleh data yang memadai untuk proses pembelajaran sekaligus menyediakan banyak data

TABEL I
CONTOH TEKS *EMAIL*

No.	Kategori	Subjek	Badan *Email*
1	Spam	*You've won $5,000!*	*Congratulations! You are the lucky winner of our monthly sweepstakes. Click the link below to claim now!*
2	Spam	*Free Gift Card Offer*	*Complete a short survey and receive a $100 gift card instantly. Limited time only!*
6	*Ham*	*Meeting Agenda for Monday*	*Hi team, please find attached the agenda for our project meeting scheduled on Monday at 10 AM.*
7	*Ham*	*Your Receipt from ABC Store*	*Thank you for your purchase. Attached is your receipt for the items bought on July 5th.*



Gambar 1. Tahapan algoritma *naïve Bayes*.

untuk menguji akurasinya. Pengujian akurasi dilakukan menggunakan algoritma *naïve Bayes*, yang melakukan klasifikasi berbasis probabilitas. Rumus algoritma *naïve Bayes* dalam penelitian ini dapat dilihat pada (4).

$$P(C|X) = \frac{P(X|C)\times P(C)}{PX} \quad (4)$$

dengan *P(C|X)* merupakan probabilitas *C* (spam/*ham*) pada *X* (teks *email*), *P(X|C)* merupakan probabilitas memperoleh fitur (*X*) pada *C*, *P(C)* merupakan probabilitas *prior C*, dan *P(X)* merupakan probabilitas fitur *X*. Berdasarkan formula ini, algoritma *naïve Bayes* dapat dijabarkan pada Gambar 1.

Gambar 1 mengilustrasikan langkah-langkah algoritma *naïve Bayes*. Data masukan adalah data *email* yang telah diproses. Probabilitas *prior* adalah probabilitas data masuk ke kelas tertentu sebelum mempertimbangkan fitur (kata) di dalamnya, seperti spam dan *ham*. Rumus untuk menghitung probabilitas *prior* ditunjukkan pada (5).

$$P(kelas) = \frac{jumlah\ data\ pada\ kelas\ tertentu}{total\ kata}. \quad (5)$$

Probabilitas *likelihood* adalah peluang suatu kata muncul dalam kelas tertentu. *Likelihood* membantu menentukan suatu kata tertentu lebih mungkin muncul dalam *email* spam atau *email ham*. Rumusnya disajikan dalam (6).

$$P(kata/kelas) = \frac{jumlah\ kemunculan\ suatu\ kata}{jumlah\ total\ kata}. \quad (6)$$

Naïve Bayes kemudian menghitung probabilitas posterior untuk setiap kelas dengan mengalikan probabilitas *prior* dan *likelihood*. Rumus yang digunakan disajikan dalam (7).

$$P(kelas|kata) = \frac{P(kelas)\times P(kata\ n|kelas)}{P(kata)}. \quad (7)$$

Dalam mengidentifikasi kelas berdasarkan probabilitas posterior tertinggi, metode *naïve Bayes* mengklasifikasikan data *email* berdasarkan nilai tertinggi antara probabilitas posterior spam dan *ham* setelah menghitung probabilitas posterior untuk kelas spam dan *ham*. Evaluasi model bertujuan untuk menilai kinerja algoritma *naïve Bayes* dalam mengidentifikasi *email* spam dan *ham*. Kriteria evaluasi yang relevan, seperti akurasi, presisi, *recall*, skor F1, dan *confusion matrix*, digunakan dalam proses peninjauan.

IV. HASIL DAN PEMBAHASAN

Bagian ini menjelaskan hasil penelitian yang telah dilakukan berdasarkan tahapan yang telah dirancang sebelumnya. Proses pengumpulan data, implementasi algoritma *naïve Bayes*, dan analisis hasil klasifikasi *email* spam dijelaskan secara sistematis. Gambaran umum keberhasilan metode yang digunakan juga disajikan. Tabel II menyajikan hasil prapemrosesan pada *dataset email* yang digunakan dalam penelitian ini, termasuk hasil tokenisasi, *lemmatization*, dan penghapusan *stopword*.

Proses tokenisasi berperan penting dalam menyiapkan data tekstual untuk dianalisis dengan memecah kalimat lengkap menjadi komponen yang lebih kecil dan lebih bermakna. Misalnya, kalimat "*You've won $1,000! Click here to claim your prize now*" diubah menjadi serangkaian token individual seperti ["*You*," "*'ve*," "*won*," "$," "1,000," "!," "*Click*," "*here*," "*to*," "*claim*," "*your*," "*prize*," "*now*," "."]. Dekomposisi ini memungkinkan model pembelajaran mesin untuk memproses dan menganalisis pola tekstual secara lebih efektif, terutama dalam tugas-tugas seperti klasifikasi *email* spam. Selanjutnya, proses *lemmatization*, yang dilakukan setelah proses tokenisasi, diterapkan untuk mengubah kata-kata ke bentuk dasarnya. Misalnya, token seperti "*selected*," "*gifts*," "*claiming*," dan "*won*" diproses menjadi masing-masing "*select*," "*gift*," "*claim*," and "*win*." Normalisasi ini memastikan bahwa variasi suatu kata diperlakukan sebagai satu istilah, sehingga meningkatkan akurasi dan efisiensi proses klasifikasi teks.

Hasil *lemmatization* yang diterapkan pada *dataset email* disajikan dalam Tabel III. Kata "*won*" diproses menjadi "*win*" dan kontraksi "*you've*" dipisahkan menjadi "*you*" dan "*have*." Pada contoh kedua, kata "*selected*" diubah menjadi "*select*" dan "*gifts*" dinormalisasi menjadi "*gift*." Sementara itu, "*suspended*" diubah menjadi "*suspend*," dan kata kerja bantu "*has*" dan "*be*" masing-masing diubah menjadi "*have*" dan "*be*." Pada contoh keempat, "*working*" diubah menjadi "*work*," dan "*needed*" menjadi "*need*." Pada contoh kelima, kata "*confirmed*" diproses menjadi "*confirm*." Perubahan ini memastikan bahwa setiap kata diproses dalam bentuk dasarnya, sehingga dapat mengurangi variabilitas dan menyederhanakan teks untuk pemrosesan lebih lanjut seperti klasifikasi *email* spam.

Proses penghapusan *stopword* menghasilkan *dataset* yang lebih bersih dan fokus pada kata-kata yang berkontribusi pada proses klasifikasi. Penghapusan *stopword* ini bertujuan untuk mengurangi derau yang dapat memengaruhi kinerja model

TABEL II
HASIL TOKENISASI

Sebelum Tokenisasi	Setelah Tokenisasi
Subject: You've won $1,000! Click here to claim your prize now.	[“*Subject*,” “:,” “*You*,” “‘,” “*ve*,” “*won*,” “$,” “1,000,” “!,” “*Click*, “*here*,” “*to*,” “*claim*,” “*your*,” “*prize*,” “*now*,” “.”]
Congratulations! You've been selected for a free gift card. Just complete this short survey.	[“*Congratulations*,” “!,” “*You*,” “‘,” “*ve*,” “*been*,” “*selected*,” “*for*,” “*a*,” “*free*,” “*gift*,” “*card*,” “.,” “*Just*,” “*complete*,” “*this*,” “*short*,” “*survey*,” “.”]
Reminder: Your account has been suspended. Login to verify your information.	[“*Reminder*,” “:,” “*Your*,” “*account*,” “*has*,” “*been*,” “*suspended*,” “.,” “*Login*,” “*to*,” “*verify*,” “*your*,” “*information*,” “.”]
Earn money fast working from home. No experience needed.	[“*Earn*,” “*money*,” “*fast*,” “*working*,” “*from*,” “*home*,” “.,” “*No*,” “*experience*,” “*needed*,” “.,”]
Your payment of $500 is confirmed.	[“*Your*,” “*payment*,” “*of*,” “$,” “500,” “*is*,” “*confirmed*,” “.”]

TABEL III
HASIL *LEMMATIZATION*

Sebelum *Lemmatization*	Setelah *Lemmatization*
Subject: You've won $1,000! Click here to claim your prize now.	[“*subject*,” “:,” “*you*,” “*have*,” “*win*,” “$,” “1,000,” “!,” “*click*,” “*here*,” “*to*,” “*claim*,” “*your*,” “*prize*,” “*now*,” “.”]
Congratulations! You"ve been selected for a free gift card. Just complete this short survey.	[“*congratulation*,” “!,” “*you*,” “*have*,” “*be*,” “*select*,” “*for*,” “*a*,” “*free*,” “*gift*,” “*card*,” “.,” “*just*,” “*complete*,” “*this*,” “*short*,” “*survey*,” “.”]
Reminder: Your account has been suspended. Login to verify your information.	[“*reminder*,” “:,” “*your*,” “*account*,” “*have*,” “*be*,” “*suspend*,” “.,” “*login*,” “*to*,” “*verify*,” “*your*,” “*information*,” “.”]
Earn money fast working from home. No experience needed.	[“*earn*,” “*money*,” “*fast*,” “*work*,” “*from*,” “*home*,” “.,” “*no*,” “*experience*,” “*need*,” “.”]
Your payment of $500 is confirmed.	[“*your*,” “*payment*,” “*of*,” “$,” “500,” “*be*,” “*confirm*,” “.”]

dalam membedakan *email* spam dan *email ham*. Tabel IV menyajikan contoh teks yang telah melalui proses penghapusan *stopword*.

Pada contoh pertama, kata-kata “*in*,” “*to*,” dan “*my*” dihilangkan karena tidak memberikan kontribusi signifikan terhadap klasifikasi. Selain itu, pada contoh tersebut, dua kata lain yang dihilangkan adalah “*the*” dan “*in*.” Setelah tahap ini, *dataset* menjadi lebih bersih dan siap untuk diproses lebih lanjut.

Hasil perhitungan TF-IDF, yang menggambarkan bobot setiap kata dalam dokumen, ditunjukkan pada Tabel V. Perhitungan dilakukan menggunakan pustaka scikit-learn yang terdapat di Python, yang secara otomatis menghitung nilai TF-IDF berdasarkan kumpulan dokumen yang dianalisis. Sementara itu, hasil perhitungan TF-IDF untuk ketiga dokumen teks *email* tersebut ditunjukkan pada Tabel VI.

Pada Tabel VI, dokumen 1, 2, dan 3 merujuk pada contoh *email* yang dikategorikan sebagai spam. Pada dokumen 1, kata “*software*” menghasilkan nilai TF-IDF sebesar 0,3192, yang menunjukkan bahwa kata ini cukup berpengaruh dalam

TABEL IV
HASIL PENGHAPUSAN *STOPWORD*

Sebelum	Setelah
help television in 1919 by seat to my knoweledge . chrono cross in 1969	[“*help*,” “*television*,” 1919, “*seat*,” “*knowledge*,” “*chrono*,” “*cross*,” 1969]
the most expensive car sold in graand ! cheap cars in graand	[“*most*,” “*expensive*,” “*car*,” “*sold*,” “*graand*,” “*cheap*,” “*car*,” “*graand*”]
save your money by getting an oem software ! need in software for your pc ? just visit our site , we might have what you need.	[“*save*,” “*money*,” “*getting*,” “*oem*,” “*software*,” “*need*,” “*software*,” “*pc*,” “*visit*,” “*site*,” “*might*,” “*have*,” “*need*”]

TABEL V
CONTOH *EMAIL*

Dokumen	Teks *Email*
Dokumen 1	*Subject: do not have money, get software cds from here ! software compatibility . . . . ain ‘ t it great ? grow old along with me the best is yet to be . all tragedies are finish ‘ d by death . all comedies are ended by marriage.*
Dokumen 2	*security alert Confirm national credit union information*
Dokumen 3	*Subject: claim your free $ 1000 homedepot gift card . claim your homedepot gift card - a $ 1000 value . were sure you can find a use for this gift card in your area ( ) . by exclusive rewards udexhoyp*

dokumen tersebut. Kata ini sering muncul dalam *email* spam, terutama pada promosi perangkat lunak bajakan atau penawaran diskon yang mencurigakan. Pada dokumen 2, kata “*alert*” menghasilkan skor TF-IDF sebesar 0,4529, yang menunjukkan bahwa kata ini cukup penting dalam dokumen tersebut. Kata ini sering digunakan dalam *email* yang bertujuan membuat penerima merasa panik atau terburu-buru, seperti dalam *email* penipuan yang meminta konfirmasi informasi keuangan segera. Pada dokumen 3, kata “*claim*” memperoleh nilai TF-IDF sebesar 0,2959, yang berarti bahwa kata ini juga cukup relevan dalam dokumen tersebut. Kata “*claim*” sering ditemukan pada *email* spam yang berisi klaim hadiah palsu atau kompensasi menggiurkan untuk menarik perhatian penerima. Secara keseluruhan, skor TF-IDF yang lebih tinggi menunjukkan lebih banyak kata yang relevan dan memiliki peran penting dalam proses klasifikasi spam dan *ham*.

Untuk melatih dan mengevaluasi model klasifikasi, data dengan berbagai rasio disiapkan selama tahap eksperimen. Distribusi data yang digunakan dalam penelitian ini adalah 80%:20%, 70%:30%, dan 60%:40%, yang sebagian besar dialokasikan untuk pelatihan dan sisanya untuk pengujian. Tujuan eksperimen ini adalah untuk menentukan pengaruh perbedaan pembagian data terhadap kinerja model, khususnya akurasi dan deteksi spam.

Tabel VII menunjukkan pembagian data yang mengalami penurunan akurasi seiring dengan penurunan proporsi data pelatihan. Penurunan ini relatif kecil dan tidak signifikan. Akurasi tertinggi diperoleh pada pembagian data 80%:20%, yang menyediakan porsi data pelatihan lebih besar pada model. Pembagian ini umumnya menghasilkan hasil yang lebih stabil dan lebih baik karena model dilatih dengan jumlah data yang lebih besar, sehingga memungkinkan model untuk menangkap pola dengan lebih baik.

TABEL VI
HASIL TF-IDF LEBIH DARI 1 DOKUMEN

No	Kata	Dokumen 1	Dokumen 2	Dokumen 3
1	*Software*	0,3192	0,0000	0,0000
2	*Compatibility*	0,3189	0,0000	0,0000
3	*Finish*	0,2158	0,0000	0,0000
4	*Death*	0,2659	0,0000	0,0000
5	*Old*	0,2039	0,0000	0,0000
6	*Union*	0,0000	0,5342	0,0000
7	*Alert*	0,0000	0,4529	0,0000
8	*National*	0,0000	0,4010	0,0000
9	*Confirm*	0,0000	0,3310	0,0000
10	*Security*	0,0000	0,3264	0,0000
11	*Credit*	0,0000	0,3023	0,0000
12	*claim*	0,0000	0,0000	0,2959
13	*Gift*	0,0000	0,0000	0,5026
14	*Card*	0,0000	0,0000	0,4109
15	*Area*	0,0000	0,0000	0,1049
16	*Value*	0,0000	0,0000	0,1039

TABEL VII
PERBANDINGAN ANTARA PELATIHAN DAN PENGUJIAN

Pelatihan (%)	Pengujian (%)	Akurasi (%)
80	20	90
70	30	89
60	40	88

Dalam penelitian ini, algoritma *naïve Bayes* digunakan untuk mengklasifikasikan *email* sebagai spam atau *ham* dengan mempertimbangkan distribusi kata dalam *dataset*. Untuk memahami cara kerja *naïve Bayes*, berikut adalah prinsip dasar dan perhitungan matematisnya. Berdasarkan *dataset* kecil *email* yang dibagi ke dalam dua kategori (spam dan *ham*), model kemudian memperkirakan probabilitas *email* baru termasuk ke dalam salah satu kategori tersebut berdasarkan kemunculan kata-kata dalam data pelatihan.

Tabel VIII menyajikan contoh data *email* yang digunakan untuk mengklasifikasikan teks ke dalam dua kategori utama: spam dan *ham*. *Dataset* terdiri atas 5.728 *email*, dengan 1.369 *email* spam dan 4.329 *email ham*. Setiap baris dalam tabel berisi teks *email* yang menunjukkan karakteristik umum dari setiap kategori. Contoh pertama dan kedua dikategorikan sebagai spam karena mengandung unsur-unsur yang umum ditemukan dalam *email* promosi atau *phishing*, seperti ajakan untuk membeli perangkat lunak murah ("*Save your money by getting an OEM software*") atau permintaan untuk mengonfirmasi informasi sensitif ("*Confirm your national credit union information*"). Jenis konten ini biasanya digunakan untuk memikat pengguna agar memberikan data pribadi atau tertarik pada penawaran yang mencurigakan. Sebaliknya, contoh ketiga dan keempat dikategorikan sebagai *ham* karena berisi pesan yang bersifat pribadi dan tidak mengandung unsur promosi atau permintaan informasi sensitif. Misalnya, sapaan pribadi seperti "*Hello David, another trip in the cards*" dan ucapan selamat seperti "*Vince, congratulations on your promotion*" mencerminkan komunikasi normal antarindividu yang umumnya tidak dianggap sebagai spam. Oleh karena itu, klasifikasi *email* berdasarkan isi teks sangat bergantung pada pemahaman konteks dan pola bahasa yang digunakan. *Email* spam cenderung menggunakan kata-kata promosi yang tidak biasa atau permintaan informasi, sedangkan email *ham* lebih bersifat pribadi dan relevan secara kontekstual. Contoh data semacam ini sangat penting dalam proses pelatihan model klasifikasi agar sistem dapat secara akurat membedakan antara spam dan *ham*.

TABEL VIII
CONTOH *DATASET EMAIL*

No	Teks *Email*	Kategori
1	"*Save your money by getting an OEM software*"	Spam
2	"*Confirm your national credit union information*"	Spam
3	"*Hello David, another trip in the cards*"	*Ham*
4	"*Vince, congratulations on your promotion*"	*Ham*

Probabilitas *prior*, *likelihood*, dan posterior adalah tiga langkah utama dalam perhitungan kategorisasi teks metode *naïve Bayes*. Proporsi dari total jumlah *email* dalam *dataset* digunakan untuk menghitung probabilitas *prior* pada tahap pertama. Masing-masing memiliki peluang awal 0,5 karena jumlah *email* spam dan *ham* sama.

Pada langkah kedua, jumlah kata unik pada masing-masing kategori dihitung. Kategori spam memiliki 10 kata unik, sedangkan kategori *ham* memiliki 7 kata, sehingga total kosakata gabungan dari kedua kategori tersebut adalah 17 kata. Selanjutnya, probabilitas kemunculan kata tertentu (seperti "*money*") di setiap kategori dihitung menggunakan *Laplace smoothing*.

Hasil perhitungan probabilitas posterior menunjukkan nilai *P* (Spam | kata) lebih besar daripada *P* (*Ham* | kata). Hasil ini menunjukkan bahwa frasa "*free software and save money*" lebih cenderung diklasifikasikan sebagai spam menurut model *naïve Bayes*. Teknik ini telah terbukti efisien dalam mengklasifikasikan teks berdasarkan kemunculan kata-kata tertentu dan sangat berguna dalam sistem deteksi spam otomatis.

Untuk menghindari nilai 0, teknik *Laplace smoothing* diterapkan ke dalam perhitungan. Tanpa teknik ini, nilai probabilitas 0 dapat menyebabkan seluruh hasil perhitungan menjadi 0.

Tabel IX menyajikan hasil perhitungan probabilitas *naïve Bayes*, yang menunjukkan bahwa setiap kata berkontribusi dalam menentukan kategori sebuah *email*. Setiap kata dalam *dataset* diberi nilai *likelihood* untuk kedua kategori (spam dan *ham*) berdasarkan frekuensi kemunculannya dalam kelas masing-masing. Misalnya, kata-kata seperti "*save*," "*money*," "*getting*," dan "*promotion*" menunjukkan nilai *likelihood* yang lebih tinggi dalam kategori spam (0,074) dibandingkan dengan kategori *ham* (0,042). Hasil ini menunjukkan bahwa istilah-istilah tersebut sering muncul dalam pesan spam. Sebaliknya, kata-kata seperti "*hello*," "*trip*," dan "*card*" memiliki nilai *likelihood* yang lebih tinggi dalam *email ham* (0,083), yang menunjukkan bahwa kata-kata tersebut lebih sering muncul dalam komunikasi yang sah.

Nilai probabilitas ini kemudian digabungkan sesuai dengan teorema *naïve Bayes* untuk menghitung probabilitas posterior untuk setiap kategori. Kategori probabilitas posterior yang lebih tinggi menentukan hasil klasifikasi akhir. Dalam hal ini, kata-kata yang memiliki *likelihood* lebih tinggi dalam kategori spam meningkatkan kemungkinan bahwa *email* yang berisi istilah-istilah tersebut akan dikategorikan sebagai spam. Dengan demikian, Tabel IX memberikan gambaran umum tentang kontribusi tingkat kata yang digunakan oleh model *naïve Bayes* untuk membedakan antara *email* spam dan *ham*. Temuan ini menunjukkan interpretasi model, khususnya dalam menunjukkan pengaruh fitur tekstual (kata kunci) terhadap hasil klasifikasi.

Model tersebut mencapai akurasi klasifikasi sempurna (100%) setelah teknik SMOTE yang tepat diterapkan pada data

TABEL IX
PERHITUNGAN NAÏVE BAYES

No	Kata	Spam	Bukan Spam	*Likelihood*	
				Spam	*Ham*
1	*save*	1	0	0.074	0.042
2	*money*	1	0	0.074	0.042
3	*getting*	1	0	0.074	0.042
4	OEM	1	0	0.074	0.042
5	*software*	1	0	0.074	0.042
6	*confirm*	1	0	0.074	0.042
7	*national*	1	0	0.074	0.042
8	*credit*	1	0	0.074	0.042
9	*union*	1	0	0.074	0.042
10	*information*	1	0	0.074	0.042
11	*hello*	0	1	0.037	0.083
12	david	0	1	0.037	0.083
13	*trip*	0	1	0.037	0.083
14	*card*	0	1	0.037	0.083
15	vince	0	1	0.037	0.083
16	*congratulation*	0	1	0.037	0.083
17	*promotion*	0	1	0.037	0.083

pelatihan. Hasil ini menunjukkan bahwa *dataset* yang diseimbangkan ulang memungkinkan pengklasifikasi *naïve Bayes* untuk secara efektif mempelajari karakteristik *email* spam tanpa bias terhadap kelas mayoritas (*ham*).

Berdasarkan temuan penelitian, kinerja model dievaluasi sebelum dan setelah penerapan SMOTE. Sebelum SMOTE, *naïve Bayes* mencapai akurasi 88%, *recall* 86%, presisi 100%, dan skor F1 92%. Hasil ini menunjukkan bahwa meskipun model dapat mengklasifikasikan *email* spam dengan presisi tinggi, banyak pesan spam masih salah diklasifikasikan sebagai *ham*, seperti yang ditunjukkan oleh nilai *recall* yang relatif rendah (86%) dan adanya banyak *false negative*.

Setelah penerapan SMOTE hanya pada data pelatihan, kinerja model meningkat secara signifikan. Nilai akurasi, presisi, *recall*, dan skor F1 semuanya mencapai 100%, yang menunjukkan bahwa model mampu mengklasifikasikan semua *email* spam dan *ham* dengan benar dalam data pengujian. Peningkatan ini mengindikasikan bahwa *dataset* yang seimbang memungkinkan pengklasifikasi mempelajari pola yang lebih representatif dari kedua kelas, sehingga menghilangkan bias dan meningkatkan generalisasi.

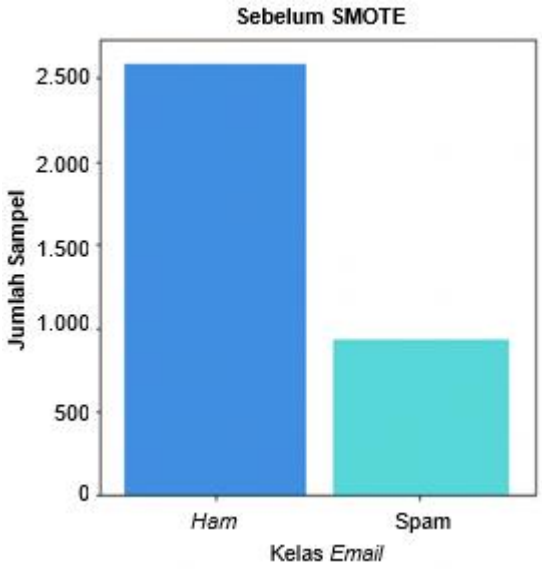
Secara keseluruhan, penerapan SMOTE secara efektif menyelesaikan masalah ketidakseimbangan kelas dan secara signifikan meningkatkan kinerja model deteksi spam. Peningkatan nilai *recall* dari 0,86 menjadi 1,00 menunjukkan bahwa model menjadi lebih sensitif dalam mendeteksi *email* spam yang sebelumnya salah diklasifikasikan. Hasil ini menunjukkan bahwa SMOTE adalah teknik yang efektif untuk meningkatkan keandalan dan ketahanan model klasifikasi teks dalam skenario data yang tidak seimbang. Peningkatan metrik kinerja selaras dengan perubahan distribusi kelas sebelum dan sesudah SMOTE, sebagaimana ditunjukkan pada Tabel X.

Model tersebut mencapai skor sempurna di semua metrik evaluasi setelah SMOTE diterapkan hanya pada data pelatihan, meningkatkan *recall* dari 0,86 menjadi 1,00 dan sepenuhnya menghilangkan *false negative*. Hal ini menegaskan bahwa SMOTE berhasil menyeimbangkan *dataset* dan meningkatkan kemampuan model untuk mendeteksi *email* spam secara akurat.

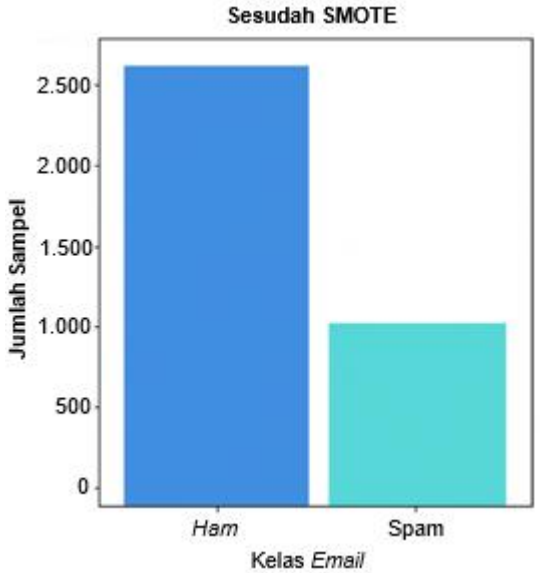
Gambar 2 menunjukkan perbandingan distribusi kelas sebelum dan sesudah penerapan SMOTE pada data pelatihan.

TABEL X
PERHITUNGAN AKURASI TEKNIK SMOTE

Metrik	Sebelum SMOTE	Setelah SMOTE (Hanya Pelatihan)
Akurasi	0,88	1,00
Presisi	1,00	1,00
Recall	0,86	1,00
Skor F1	0,92	1,00
Penunjang (data pengujian)	912	912



(a)

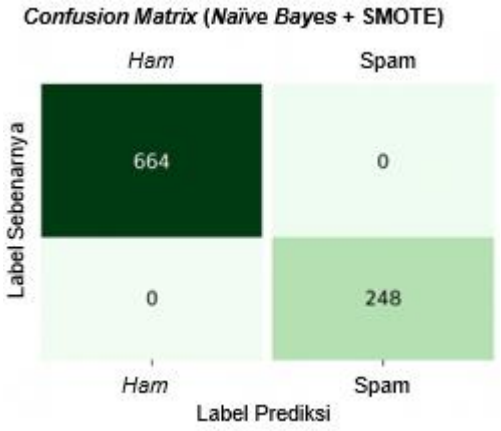


(b)

Gambar 2. Perbedaan distribusi data, (a) sebelum SMOTE, (b) setelah SMOTE.

Sebelum SMOTE, data sangat tidak seimbang, dengan kategori *ham* mendominasi kategori spam, menyebabkan model salah mengklasifikasikan banyak *email* spam. Setelah penerapan SMOTE pada data pelatihan, distribusi menjadi seimbang, memungkinkan model untuk mempelajari pola dari kedua kategori secara proporsional. Penyesuaian ini meningkatkan kemampuan model untuk mengenali *email* spam dan meningkatkan kinerja *recall* tanpa menimbulkan bias.

Gambar 3 menyajikan *confusion matrix* dari model *naïve Bayes* akhir setelah penerapan teknik SMOTE pada data



Gambar 3. *Confusion matrix* (*naïve Bayes* + SMOTE).

pelatihan. Matriks tersebut menunjukkan bahwa model mencapai hasil klasifikasi yang sempurna, tanpa *false positive* (FP = 0) dan tanpa *false negative* (FN = 0). Hasil ini menunjukkan bahwa sebanyak 664 *email ham* diidentifikasi dengan benar sebagai *ham*, dan sebanyak 248 *email* spam diklasifikasikan secara akurat sebagai spam.

Hasil penelitian menunjukkan bahwa model mencapai akurasi, presisi, *recall*, dan skor F1 sebesar 100%, yang menunjukkan bahwa data seimbang yang dihasilkan melalui SMOTE memungkinkan pengklasifikasi untuk secara efektif membedakan antara *email* spam dan *email ham*. Hal ini menegaskan bahwa kombinasi ekstraksi fitur TF-IDF, SMOTE untuk penyeimbangan kelas, dan klasifikasi *naïve Bayes* memberikan pendekatan yang sangat andal untuk deteksi *email* spam.

Kinerja tersebut menunjukkan bahwa model telah berhasil mempelajari pola teks yang mendasari *email* spam dan *email* sah, meminimalkan baik peringatan palsu maupun deteksi yang terlewat. Oleh karena itu, konfigurasi ini dapat diimplementasikan secara efektif dalam sistem penyaringan spam ringan untuk meningkatkan keamanan *email*.

V. KESIMPULAN

Pada tahap awal, model *naïve Bayes* mencapai akurasi 88%, dengan *recall* 86%, presisi 100%, dan skor F1 92%. Namun, model tersebut menunjukkan jumlah *false negative* yang relatif tinggi (137), yang mengindikasikan bahwa banyak *email* spam salah diklasifikasikan sebagai *email ham*. Sebaliknya, nilai *false positive* adalah 0, menunjukkan bahwa model tersebut tidak pernah salah mengklasifikasikan *email* yang sah sebagai spam. Hal ini terjadi karena data sangat tidak seimbang, dengan proporsi *email* spam yang jauh lebih besar, sehingga menyebabkan model bias terhadap kelas mayoritas dan berkinerja buruk pada kelas minoritas.

Setelah penerapan SMOTE, model tersebut mencapai akurasi, presisi, *recall*, dan skor F1 sebesar 100%, sedangkan nilai *false positive* dan *false negative* berkurang menjadi nol. Temuan ini menegaskan bahwa penerapan teknik penyeimbangan kelas yang tepat seperti SMOTE dapat secara signifikan meningkatkan keandalan dan kemampuan generalisasi model deteksi spam.

Terlepas dari hasil yang sangat baik, penelitian ini memiliki beberapa keterbatasan. Model yang diusulkan menggunakan algoritma yang relatif sederhana (*naïve Bayes*) dan kumpulan data berukuran sedang, yang dapat membatasi kemampuan generalisasinya pada kumpulan data yang lebih besar, lebih beragam atau multibahasa. Selain itu, akurasi sempurna yang diperoleh dalam kondisi eksperimental terkontrol mungkin tidak sepenuhnya mencerminkan lingkungan dunia nyata, dengan konten spam yang berkembang secara dinamis dan mungkin mencakup komponen multimedia atau dihasilkan oleh AI.

Penelitian selanjutnya disarankan untuk mengevaluasi model menggunakan *dataset* yang lebih besar dan lebih heterogen, dengan memasukkan sampel spam multibahasa dan berbasis gambar untuk menguji keefektifannya. Studi selanjutnya juga dapat mengeksplorasi integrasi arsitektur *deep learning*, metode *ensemble*, atau teknik representasi teks hibrida untuk meningkatkan akurasi deteksi dan kemampuan adaptasi. Selain itu, implementasi dan pengujian model dalam sistem penyaringan *email* waktu nyata akan menjadi langkah berharga untuk menilai skalabilitas, efisiensi, dan penerapannya di lingkungan operasional.

KONFLIK KEPENTINGAN

Penulis menyatakan tidak ada konflik kepentingan.

KONTRIBUSI PENULIS

Konseptualisasi, Andi Maslan; metodologi, Andi Maslan; perangkat lunak, Azan Rahman; validasi, Andi Maslan; penulisan—penyusunan draf asli, Andi Maslan; penulisan—peninjauan dan penyuntingan, Umar Faruq, Andi Maslan, dan Rabei Raad Ali Al-Jawry; visualisasi, Andi Maslan; pengawasan, Rabei Raad Ali Al-Jawry dan Azan Rahman; pendanaan, Andi Maslan dan Azan Rahman.

UCAPAN TERIMA KASIH

Penulis menyampaikan terima kasih kepada Universitas Putera Batam atas dukungan dan pendanaan yang diberikan dalam pelaksanaan penelitian ini. Tanpa bantuan dan komitmen kuat mereka, penelitian ini tidak akan dapat dilaksanakan. Selain itu, penulis ingin menyampaikan apresiasi kepada seluruh pihak yang telah memberikan kontribusi, baik secara langsung maupun tidak langsung, dalam proses penelitian ini.

REFERENSI

[1] J. Hong, "The state of phishing attacks," *Commun. ACM*, vol. 55, no. 1, hal. 74–81, Jan. 2012, doi: 10.1145/2063176.2063197.

[2] D. Hylender, P. Langlois, A. Pinto, dan S. Widup, "2024 Data Breach Investigations Report," The Verizon DBIR Team, California, CA, AS, 2024.

[3] World Economic Forum, "Global Cybersecurity Outlook 2024, 2024. [Online]. Tersedia: https://www3.weforum.org/docs/WEF_Global_Cybersecurity_Outlook_2024.pdf

[4] R.G. Broadhurst dan H. Trivedi, "Malware in spam email: Risks and trends in the Australian spam intelligence database," *Trends Issues Crime Crim. Justice*, no. 603, hal. 1–18, Sep. 2020, doi: 10.52922/ti04657.

[5] Slashnext, "The State of Phishing 2023," 2023. [Online]. Tersedia: https://newsletter.radensa.ru/wp-content/uploads/2023/10/SlashNext-The-State-of-Phishing-Report-2023.pdf

[6] B.P.S. Brodjonegoro, "Pembangunan digital Indonesia ke depan," dalam Horizon Pembangunan Digital Indonesia 2025-2030, 1st ed., Nezar Patria, Ed., Jakarta, Indonesia: Kementerian Komunikasi dan Informatika, 2024.

[7] M. Thomas dan B. Meshram, "A brief review of network forensics process models and a proposed systematic model for investigation," dalam *Intell. Cyber Phys. Syst. Internet Things ICoICI 2022,* 2023, hal. 599–627, doi: 10.1007/978-3-031-18497-0_45.

[8] B. Fakiha, "Enhancing cyber forensics with AI and machine learning: A study on automated threat analysis and classification," *Int. J. Saf. Secur. Eng.*, vol. 13, no. 4, hal. 701–707, Sep. 2023, doi: 10.18280/ijsse.130412.

[9] R. Al-Mugern, S.H. Othman, dan A. Al-Dhaqm, "An improved machine learning method by applying cloud forensic meta-model to enhance the data collection process in cloud environments," *Eng. Technol. Appl. Sci.*

Res., vol. 14, no. 1, hal. 13017–13025, Feb. 2024, doi: 10.48084/etasr.6609.

[10] C.S. Eze dan L. Shamir, "Analysis and prevention of AI-based phishing email attacks," *Electronics*, vol. 13, no. 10, hal. 1–13, Mei 2024, doi: 10.3390/electronics13101839.

[11] W.A. Baroto, "Advancing digital forensic through machine learning: An integrated framework for fraud investigation," *Asia Pac. Fraud J.*, vol. 9, no. 1, hal. 1–16, Jan.-Jun. 2024, doi: 10.21532/apfjournal.v9i1.346.

[12] N.N. Nicholas dan V. Nirmalrani, "An enhanced mechanism for detection of spam emails by deep learning technique with bio-inspired algorithm," *e-Prime - Adv. Elect. Eng. Electron. Energy*, vol. 8, hal. 1–9, Jun. 2024, doi: 10.1016/j.prime.2024.100504.

[13] I.S.A. Munawar dan E.T. Arujisaputra, "Comparison of the results of the naïve Bayes method and synthetic minority over sampling technique in sentiment analysis of user reviews," *J. Elektron. Sist. Inf.*, vol. 2, no. 1, hal. 179–188, Jun. 2024, doi: 10.31848/jesii.v2i1.3416.

[14] A. Falasari dan M.A. Muslim, "Optimize naïve Bayes classifier using chi square and term frequency inverse document frequency for amazon review sentiment analysis," *J. Soft Comput. Explor.*, vol. 3, no. 1, hal. 31–36, Mar. 2022, doi: 10.52465/joscex.v3i1.68.

[15] B. Sonare dkk., "E-mail spam detection using machine learning," dalam *2023 4th Int. Conf. Emerg. Technol. (INCET),* 2023, hal. 1–5, doi: 10.1109/INCET57972.2023.10170187.

[16] G. Nasreen dkk., "Email spam detection by deep learning models using novel feature selection technique and BERT," *Egypt. Inform. J.*, vol. 26, hal. 1–11, Jun. 2024, doi: 10.1016/j.eij.2024.100473.

[17] M.M. Abualhaj dkk., "Enhancing spam detection using Harris Hawks optimization algorithm," *TELKOMNIKA Telecommun. Comput. Electron. Control,* vol. 23, no. 2, hal. 447–454, Apr. 2025, doi: 10.12928/TELKOMNIKA.v23i2.26615.

[18] M. Jaiswal, S. Das, dan Khushboo, "Detecting spam e-mails using stop word TF-IDF and stemming algorithm with naïve Bayes classifier on the multicore GPU," *Int. J. Elect. Comput. Eng.*, vol. 11, no. 4, hal. 3168–3175, Agu. 2021, doi: 10.11591/ijece.v11i4.pp3168-3175.

[19] A.C. Sitepu, Wanayumini, dan Z. Situmorang, "Determining bullying text classification using naive Bayes classification on social media," *J. Varian*, vol. 4, no. 2, hal. 133–140, Apr. 2021, doi: 10.30812/varian.v4i2.1086.

[20] M. Thomas dan B.B. Meshram, "An efficient spam mail classification approach using improved moth flame optimization and multi-class support vector machine," *Int. J. Intell. Eng. Syst.*, vol. 16, no. 6, hal. 813–823, 2023, doi: 10.22266/ijies2023.1231.67.

[21] C.N. Mohammed dan A.M. Ahmed, "A semantic-based model with a hybrid feature engineering process for accurate spam detection," *J. Elect. Syst. Inf. Technol.*, vol. 11, hal. 1–16, Jul. 2024, doi: 10.1186/s43067-024-00151-3.

[22] A. Rajesh dan T. Hiwarkar, "Exploring preprocessing techniques for natural languagetext: A comprehensive study using Python code," *Int. J. Eng. Technol. Manag. Sci.*, vol. 7, no. 5, hal. 390–399, Sep./Okt. 2023, doi: 10.46647/ijetms.2023.v07i05.047.

[23] M.R. Ningsih, J. Unjung, H.A. Farih, dan M.A. Muslim, "Classification email spam using naive Bayes algorithm and chi-squared feature selection," *J. Appl. Intell. Syst. (JAIS),* vol. 9, no. 1, hal. 74–87, Apr. 2024, doi: 10.62411/jais.v9i1.9695.

[24] N. Tayefeh, "Email Spam," Harvard Dataverse, doi: 10.7910/DVN/V8BSBP.

[25] E.G. Dada dkk., "Machine learning for email spam filtering: Review, approaches and open research problems," *Heliyon*, vol. 5, no. 6, hal. 1–23, Jun. 2019, doi: 10.1016/j.heliyon.2019.e01802.

[26] O. Ogundairo dan P. Broklyn, "AI-driven phishing detection systems," tidak dipublikasikan.

[27] B.A. Kumari dan C. Nagaraju, "Robust machine learning technique for detection and classification of spam mails," tidak dipublikasikan.

[28] N. Borisova, E. Karashtranova, dan I. Atanasova, "The advances in natural language processing technology and its impact on modern society," *Int. J. Elect. Comput. Eng. (IJECE),* vol. 15, no. 2, hal. 2325–2333, Apr. 2025, doi: 10.11591/ijece.v15i2.pp2325-2333.

[29] Z. Abidin, A. Junaidi, dan Wamiliana, "Text stemming and lemmatization of regional languages in Indonesia: A systematic literature review," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 10, no. 2, hal. 217–231, Jun. 2024, doi: 10.20473/jisebi.10.2.217-231.

[30] A.B.P. Negara, "The influence of applying stopword removal and smote on Indonesian sentiment classification*," Lontar Komput., J. Ilm. Teknol. Inf.*, vol. 14, no. 3, hal. 172–185, Des. 2023, doi: 10.24843/lkjiti.2023.v14.i03.p05.

[31] D. Khurana, A. Koli, K. Khatter, dan S. Singh, "Natural language processing: State of the art, current trends and challenges," *Multimed. Tools Appl.*, vol. 82, no. 3, hal. 3713–3744, Jan. 2023, doi: 10.1007/s11042-022-13428-4.

[32] A. Daud dkk., "Identification of chatbot usage in online store services using natural language processing methods," *Adv. Sustain. Sci. Eng. Technol.*, vol. 6, no. 2, hal. 1–9, Feb.-Apr. 2024, doi: 10.26877/asset.v6i2.18309.

[33] S. Atawneh dan H. Aljehani, "Phishing email detection model using deep learning," *Electronics*, vol. 12, no. 20, hal. 1–26, Okt. 2023, doi: 10.3390/electronics12204261.

[34] F. Millennianita, U. Athiyah, dan A.W. Muhammad, "Comparison of naïve Bayes classifier and support vector machine methods for sentiment classification of responses to bullying cases on Twitter," *J. Mechatron. Artif. Intell.*, vol. 1, no. 1, hal. 11–26, Jun. 2024, doi: 10.17509/jmai.v1i1.69959.