Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443

Deteksi URL Phishing Menggunakan Natural Language Processing Dan Support Vector Machine Berbasis Machine Learning

Nabila Nabila, Emilia Hesti*, Aryanti Aryanti

Teknik Elektro, Sarjana Terapan Teknik Telekomunikasi, Politeknik Negeri Sriwijaya, Palembang, Indonesia
Email: [1]nabilabila2934@gmail.com, [2,*]emiliahesti@ymail.com, [3]aryanti@polsri.ac.id
Email Penulis Korespondensi: emiliahesti@ymail.com
Submitted: **27/05/2025**; Accepted: **22/06/2025**; Published**: 22/06/2025**

Abstrak−Phishing merupakan bahaya yang signifikan dalam keamanan siber, menggunakan URL jahat untuk menyesatkan pengguna agar mengungkapkan informasi penting. Penelitian ini berupaya untuk membuat model deteksi URL phishing menggunakan pembelajaran mesin melalui integrasi ekstraksi fitur URL struktural, metodologi Natural Language Processing (NLP), dan algoritma klasifikasi Support Vector Machine (SVM). Indikator tren phishing diperoleh dari fitur-fitur seperti panjang URL, jumlah titik, dan garis miring, sementara konten URL dikuantifikasi sebagai vektor numerik menggunakan Term Frequency-Inverse Document Frequency (TF-IDF). Semua karakteristik selanjutnya diintegrasikan sebagai input ke dalam model support vector machine dengan kernel linier untuk klasifikasi. Hasil evaluasi dari laporan klasifikasi menunjukkan bahwa integrasi TF-IDF dan SVM kernel linier mencapai kinerja optimal, dengan akurasi 90%, presisi 92%, recall 89%, dan skor F1 90%. Sebaliknya, matriks kebingungan menunjukkan akurasi 90,29%, presisi 91,66%, ingatan 88,62%, dan skor F1 90,12%. Studi ini terutama berkontribusi dengan mengintegrasikan NLP dan SVM ke dalam model deteksi phishing adaptif terpadu melalui penggabungan aspek struktural dan tekstual URL. Strategi ini memfasilitasi deteksi phishing yang lebih baik dibandingkan dengan teknik yang hanya bergantung pada karakteristik manual. Model ini, tidak seperti penelitian lain yang difokuskan pada contoh tertentu atau NLP yang dikecualikan, dirancang untuk mengidentifikasi banyak kategori URL phishing secara luas, sehingga meningkatkan relevansinya dalam menangani serangan siber yang terus berkembang.

Kata Kunci: Phishing; Natural Language Processing; Support Vector Machine; Term Frequency-Inverse Document Frequency

Abstract−Phishing represents a significant danger in cybersecurity, using malicious URLs to mislead users into revealing critical information. This research seeks to create a phishing URL detection model using machine learning via the integration of structural URL feature extraction, Natural Language Processing (NLP) methodologies, and the Support Vector Machine (SVM) classification algorithm. Indicators of phishing trends are derived from features such as URL length, the quantity of dots, and slashes, while URL content is quantified as numerical vectors using Term Frequency-Inverse Document Frequency (TF-IDF). All characteristics are subsequently integrated as input into a support vector machine model with a linear kernel for classification. The evaluation results from the classification report indicate that the integration of TF-IDF and linear kernel SVM achieves optimal performance, with 90% accuracy, 92% precision, 89% recall, and 90% F1-score. Conversely, the confusion matrix reveals 90.29% accuracy, 91.66% precision, 88.62% recall, and 90.12% F1-score. This study primarily contributes by integrating NLP and SVM into a unified adaptive phishing detection model via the amalgamation of structural and textual aspects of URLs. This strategy facilitates enhanced phishing detection relative to techniques reliant only on manual characteristics. This model, unlike other research that concentrated on particular instances or excluded NLP, is engineered to identify many categories of phishing URLs broadly, hence enhancing its relevance in tackling the dynamic nature of assaults.

Keywords: Phishing; Natural Language Processing; Support Vector Machine; Term Frequency-Inverse Document Frequency

1. PENDAHULUAN

Asosiasi Penyelenggara Jasa Internet Indonesia (APJII) memproyeksikan jumlah pengguna internet akan mencapai 221.563.479 jiwa pada 2024 dari total populasi penduduk sebesar 278.696.200 jiwa pada 2023. Survei Penetrasi Internet Indonesia 2024 yang dirilis APJII menunjukkan bahwa tingkat penetrasi internet Indonesia telah mencapai 79,5% atau meningkat 1,4% dibandingkan kuartal sebelumnya[1]. Meskipun demikian, maraknya penggunaan internet diiringi dengan meningkatnya kekhawatiran keamanan siber, salah satunya termasuk upaya *phishing*[2].

Menurut laporan kuartal pertama tahun 2023, Indonesia Anti-*Phishing* Data *Exchange* (IDADX) menerima 26.675 laporan *phishing*, mengalami lonjakan signifikan dibandingkan 6.106 laporan pada kuartal sebelumnya. Peningkatan ini mencapai 20.569 laporan hanya dalam satu kuartal. Sektor media sosial menjadi sasaran utama, mencakup 45% dari total serangan *phishing* yang dilaporkan, disusul oleh lembaga keuangan dengan 31%. Indonesia juga menduduki peringkat teratas sebagai negara dengan situs *phishing* yang menggunakan domain .id, diikuti oleh Amerika Serikat[3]. Berdasarkan data dari IDADX kuartal keempat tahun 2024, kategori *phishing* mendominasi dengan total laporan mencapai 85.414 kasus[4]. Walaupun teknologi keamanan terus mengalami kemajuan, serangan *phishing* masih efektif karena mengeksploitasi kelemahan manusia, seperti rendahnya tingkat kewaspadaan dan literasi digital[5].

Phishing dapat menyerang siapa pun, mulai dari individu hingga perusahaan besar, dengan konsekuensi yang berpotensi sangat merugikan, terutama bagi siswa sekolah dan pihak sekolah[6], Pelajar dan pihak sekolah memiliki peran penting karena mereka kerap menjadi sasaran *phishing* dan serangan siber lainnya Oleh karena itu, penerapan sistem pendeteksi URL *phishing* sangat penting untuk melindungi mereka dari ancaman siber yang terus berkembang.

Strategi proaktif sangat penting untuk mengidentifikasi URL *phishing* secara cepat dan tepat guna untuk mencegah dampak yang lebih luas. Seiring dengan kemajuan teknologi, gagasan *Machine Learning* (ML) telah muncul sebagai metode dalam Kecerdasan Buatan (AI) yang dirancang untuk meniru peran manusia dalam tugas

Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443

pemecahan masalah. Machine learning telah menunjukkan kemanjurannya di beberapa domain, khususnya dalam mengidentifikasi serangan siber[7]. Dengan menggunakan metode *machine learning*, model klasifikasi dapat dibangun untuk mengidentifikasi pola dan atribut situs web *phishing*, yang memungkinkan pembedaan yang tepat antara situs legal dan situs yang mungkin melakukan penipuan[7].

Dalam beberapa penelitian terdahulu yang dilakukan oleh Nurhaliza, Rice, Mustakim, dan Nesdi pada tahun 2024 yang berjudul "Implementasi TF-IDF dan *Word*2Vec pada Analisis Sentimen Vaksin Booster Menggunakan Algoritma *Support Vector Machine*," sebanyak 13.297 titik data Twitter dianalisis menggunakan algoritma *Support Vector Machine* beserta dua teknik ekstraksi fitur: *Term Frequency-Inverse Document Frequency* dan *Word*2Vec. Hasil optimal dicapai dengan integrasi SVM dan TF-IDF dalam skenario distribusi data 80:20, menghasilkan akurasi sebesar 85%. Sebaliknya, penggabungan *Support Vector Machine* dan *Word*2Vec menghasilkan akurasi sebesar 81%[8].

Pada tahun 2022, Muhammad Zaid Naeem, Furqan Rustam, Arif Mehmood, Mui-zzud-din, Imran Ashraf, dan Gyu Sang Choi melakukan penelitian klasifikasi sentimen menggunakan algoritma SVM dengan kernel linear untuk menganalisis ulasan film di IMDb. Hasilnya menunjukkan bahwa SVM dengan fitur TF-IDF mencapai akurasi tertinggi sebesar 89,55%, lebih baik dibandingkan metode ekstraksi fitur lain seperti BoW. Penelitian ini mendukung penggunaan TF-IDF yang dipadukan dengan SVM linear sebagai metode efektif untuk klasifikasi sentimen teks[9].

Penelitian oleh Bandar Alshawi menunjukkan bahwa kernel Linear unggul dalam deteksi kecurangan kartu kredit, dengan akurasi tertinggi mencapai 95%. Meskipun kernel Linear dan Polynomial memiliki skor ROC yang sama (91%), kernel Linear lebih unggul dalam hal akurasi dan keseimbangan antara presisi dan *recall*, dengan skor F1 tertinggi 92%. Sementara itu, kernel RBF menunjukkan performa yang lebih rendah, dengan akurasi 83%, meskipun memiliki recall yang cukup baik (85%). Hal ini menjadikan kernel Linear pilihan terbaik untuk klasifikasi transaksi secara keseluruhan, mengindikasikan kemampuan superior dalam membedakan antara transaksi sah dan curang dibandingkan dengan kernel lainnya[10].

Pada penelitian tahun 2024 oleh Fitra, Sriyanto, dan Zarnelly yang menganalisis penerapan algoritma *Decision Tree* dalam keamanan siber untuk klasifikasi situs web *phishing*, percobaan menunjukkan bahwa model *Decision Tree* yang dibangun mencapai akurasi sebesar 87,04% dan menunjukkan kinerja yang baik dalam membedakan situs *phishing* dan situs yang sah[11]. Penelitian tersebut menerapkan ekstraksi fitur dari struktur URL. Namun, fitur yang digunakan masih terbatas pada aspek manual, seperti panjang URL dan karakter simbol, tanpa mengimplementasikan pendekatan berbasis teks melalui *Natural Language Processing* (NLP). Selain itu, tidak dijelaskan secara detail mengenai keseimbangan distribusi data yang digunakan dalam proses klasifikasi.

Pada tahun 2024, Arfian, Anindya, Nabila, Keisha, dan Elsya melakukan penelitian tentang penerapan metode K-*Nearest Neighbours* (KNN) untuk mendeteksi situs web *phishing*. Analisis data menunjukkan bahwa metode KNN efektif mengidentifikasi situs web *phishing* dengan akurasi sebesar 88%[12]. Namun, penelitian tersebut belum menerapkan pendekatan *Natural Language Processing* (NLP) maupun ekstraksi fitur struktur URL secara eksplisit, sehingga fitur yang digunakan masih terbatas. Selain itu, algoritma K-Nearest Neighbors (KNN) memiliki kekurangan yang signifikan, termasuk kerentanan terhadap outlier dan tuntutan pemrosesan yang substansial untuk kumpulan data yang besar, karena memerlukan perhitungan jarak antara setiap titik data. Pilihan nilai k secara signifikan memengaruhi hasil, karena nilai yang terlalu kecil dapat menyebabkan overfitting, sedangkan nilai yang terlalu tinggi dapat mengakibatkan underfitting. Penelitian ini menggarisbawahi pentingnya distribusi data yang adil dalam kategorisasi URL *phishing*.

Penelitian tahun 2024 oleh Azzam dan Setia menguji deteksi *phishing* situs web dengan metode klasifikasi *machine learning*. Penelitian ini membandingkan tiga algoritma kategorisasi: *Decision Tree*, *Random Forest*, dan K-*Nearest Neighbours* (K-NN). Model pertama menggunakan pendekatan *Decision Tree* mencapai akurasi 83,3%, model berikutnya yang memanfaatkan strategi *Random Forest* mencapai akurasi 83,4%, dan model ketiga yang menerapkan algoritma K-Nearest Neighbours menunjukkan kinerja yang lebih buruk dengan akurasi 48,2%[7]. Penelitian ini hanya mengandalkan teknik ekstraksi fitur manual dari struktur URL tanpa menggunakan pendekatan *Natural Language Processing* (NLP) untuk mengolah elemen teks dari URL. Tidak adanya pemanfaatan informasi tekstual dalam struktur URL dapat menjadi salah satu alasan mengapa akurasi model, khususnya pada KNN, cukup rendah.

Penelitian tentang klasifikasi deteksi tautan *phishing* oleh DANA Kaget dengan Metode Website-*Based Support Vector Machine*, seperti yang dilakukan oleh Mutiara dan Wiyli (2024), menunjukkan kinerja yang konsisten dengan peningkatan akurasi baik pada data pelatihan maupun pengujian seiring dengan bertambahnya jumlah lipatan dalam validasi silang. Akurasi maksimum yang dicapai adalah 90% untuk pelatihan dan 88% untuk pengujian[13]. Penelitian tersebut menunjukkan hasil akurasi yang cukup tinggi dengan penggunaan algoritma *Support Vector Machine* (SVM) kernel linear, penelitian tersebut memiliki beberapa keterbatasan yang perlu dicermati. Fokus penelitian hanya terbatas pada kasus *phishing* yang meniru situs DANA Kaget, sehingga cakupannya sempit dan kurang dapat digeneralisasi ke bentuk serangan *phishing* lainnya. Selain itu, teknik vektorisasi teks yang digunakan adalah *CountVectorizer*, yang hanya menghitung frekuensi kemunculan kata tanpa mempertimbangkan relevansi atau bobot kata secara keseluruhan dalam korpus data. Hal ini berpotensi menurunkan kualitas representasi fitur, terutama ketika menghadapi URL *phishing* yang memiliki struktur teks kompleks.

Meskipun berbagai penelitian telah menggunakan algoritma machine learning seperti Decision Tree, K-Nearest Neighbors (KNN), *Random Forest* dan Support Vector Machine (SVM) dalam deteksi *phishing*, sebagian

Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443

besar masih mengandalkan fitur manual yang terbatas pada struktur URL tanpa mengintegrasikan teknik *Natural Language Processing* (NLP) untuk ekstraksi fitur tekstual. Selain itu, beberapa penelitian kurang memperhatikan keseimbangan distribusi data, yang dapat mempengaruhi kestabilan dan akurasi model. Penelitian oleh Mutiara dan Wiyli (2024) menggunakan SVM dengan teknik *CountVectorizer* pada kasus spesifik situs DANA Kaget, sehingga kurang dapat digeneralisasi untuk berbagai jenis *phishing*. Sebaliknya, penelitian ini mengisi kesenjangan tersebut dengan mengembangkan model deteksi URL *phishing* berbasis *machine learning* yang melalui beberapa tahapan. Dimulai dengan tahap *preprocessing* untuk membersihkan dan menyiapkan data, diikuti dengan ekstraksi 21 fitur struktural url, seperti panjang URL, jumlah titik, dan fitur lainnya yang relevan dengan *phishing*. Setelah itu, fitur-fitur tersebut dianalisis lebih lanjut menggunakan *Term Frequency-Inverse Document Frequency* (TF-IDF) dalam pendekatan NLP untuk menangkap pola tekstual dalam URL yang sering digunakan pada situs *phishing*. Akhirnya, data yang telah diproses dan dianalisis diterapkan pada model *Support Vector Machine* (SVM) dengan kernel linear, yang telah diuji sebelumnya, serta menggunakan dataset seimbang antara URL *phishing* dan URL legal untuk meningkatkan kestabilan dan akurasi model. Dengan demikian, penelitian ini menawarkan pendekatan yang lebih komprehensif, adaptif, dan relevan untuk mendeteksi *phishing* dalam berbagai skenario, sekaligus menjawab keterbatasan dan gap yang ada pada penelitian sebelumnya.

Support Vector Machine (SVM) dalam pembelajaran mesin digunakan untuk tugas klasifikasi dan regresi. Metode ini bekerja dengan mengidentifikasi *hyperplane* terbaik untuk membedakan antara dua set data atau kelas yang berbeda. *Support Vector Machine* (SVM) unggul dalam mengelola set data yang rumit dan secara efisien mengatasi tantangan klasifikasi, terutama ketika data tidak dapat dipisahkan secara linier[13]. Dalam *Natural Language Processing* (NLP), berkonsentrasi pada pemrosesan teks memerlukan metode untuk mengubah teks menjadi format numerik untuk analisis komputer[14] prosedur ini disebut sebagai vektorisasi. Salah satu metode vektorisasi adalah *Term Frequency-Inverse Document Frequency* (TF-IDF), yang memberikan skor untuk setiap kata dalam dokumen berdasarkan frekuensinya dalam dokumen tersebut (TF - *Term Frequency*) dan kelangkaannya di seluruh koleksi dokumen (IDF - *Inverse Document Frequency*)[15].

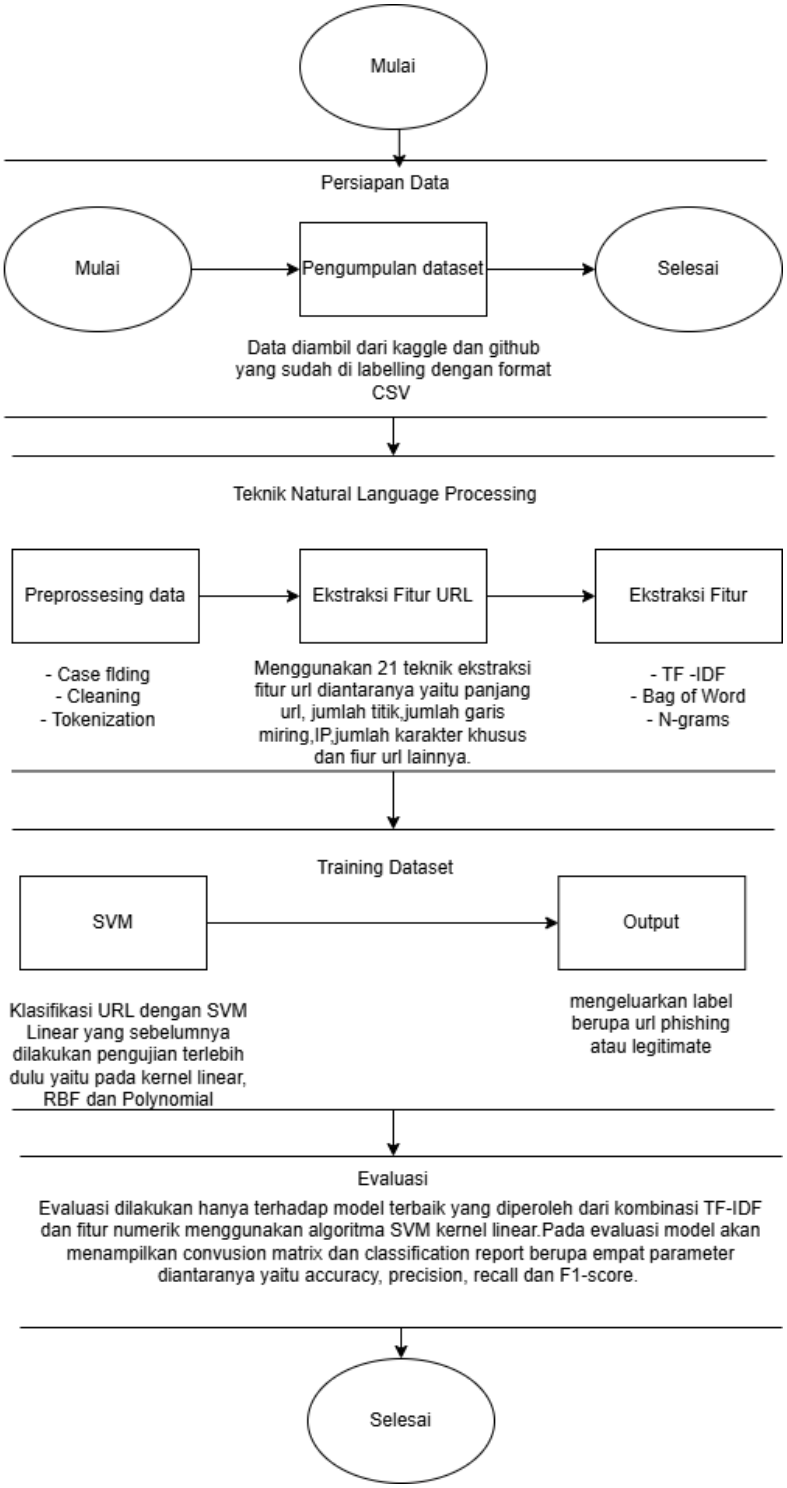
Dengan mengintegrasikan SVM dan TF-IDF, penelitian ini memungkinkan identifikasi pola tekstual dan struktur URL yang sering digunakan dalam situs *phishing*, sehingga meningkatkan akurasi dan efektivitas deteksi. Tujuan utama penelitian ini adalah mengembangkan dan mengevaluasi model deteksi URL *phishing* yang lebih akurat dan relevan, guna meningkatkan keamanan siber dan kesadaran digital, khususnya di kalangan individu dan organisasi yang rentan terhadap serangan *phishing*. *Phishing* tidak hanya mencuri data pribadi seperti nama, alamat, dan informasi login, tetapi juga data keuangan dan akses sistem yang dapat disalahgunakan untuk pencurian identitas, penipuan finansial, dan serangan siber lebih lanjut. Penelitian ini bertujuan untuk melindungi data pribadi dan informasi sensitif yang tersimpan pada perangkat sekolah, organisasi, atau individu, dari potensi ancaman *phishing*, dengan menciptakan solusi yang efektif untuk mengurangi risiko serangan siber di berbagai sektor.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian ini disusun melalui beberapa tahapan yang terstruktur dan saling berkaitan, dimulai dari pengumpulan dan persiapan dataset, dilanjutkan dengan proses ekstraksi fitur menggunakan teknik *Natural Language Processing* (NLP), pelatihan model menggunakan algoritma *Support Vector Machine* (SVM), dan diakhiri dengan evaluasi performa model. Pada tahap ekstraksi fitur berbasis teks, dilakukan perbandingan antara tiga metode representasi teks yang umum digunakan, yaitu TF-IDF, *Bag of Words* (BoW), dan n-gram. Berdasarkan hasil uji coba terhadap kombinasi metode NLP dengan berbagai kernel SVM (Linear, RBF, Polynomial), diperoleh bahwa kombinasi TF-IDF dan fitur numerik dengan SVM kernel linear menghasilkan performa terbaik dalam mendeteksi URL *phishing*. Pemilihan TF-IDF didasarkan tidak hanya pada hasil uji coba internal, tetapi juga sejalan dengan penelitian [16] yang menunjukkan bahwa kombinasi TF-IDF dan SVM dengan kernel linear menghasilkan akurasi yang tinggi. Evaluasi model dilakukan dengan menggunakan *classification report* dan *confusion matrix* pada model terbaik, yaitu kombinasi TF-IDF dengan SVM kernel linear, yang mencakup metrik seperti akurasi, *precision*, *recall*, dan F1-*score*. Berikut ini adalah diagram tahapan yang menggambarkan seluruh proses pengembangan sistem deteksi URL phishing secara terstruktur yang disajikan dalam Gambar 1. Diagram ini juga berfungsi sebagai alat bantu visual yang mempertegas logika alur kerja penelitian, sehingga memudahkan pembaca dalam memahami kompleksitas setiap langkah yang dilakukan. Tidak hanya menggambarkan urutan tahapan, tetapi juga mencerminkan pentingnya konsistensi dan kesinambungan antara proses satu dengan lainnya. Misalnya, keberhasilan pada tahap pelatihan model sangat dipengaruhi oleh ketepatan dalam ekstraksi fitur sebelumnya, sementara hasil evaluasi akhir menjadi indikator keberhasilan seluruh rangkaian proses. Oleh karena itu, representasi visual ini tidak hanya bersifat informatif, tetapi juga strategis dalam memberikan gambaran menyeluruh terhadap struktur penelitian yang telah dirancang secara sistematis dan berbasis pada pendekatan ilmiah yang valid.

Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443

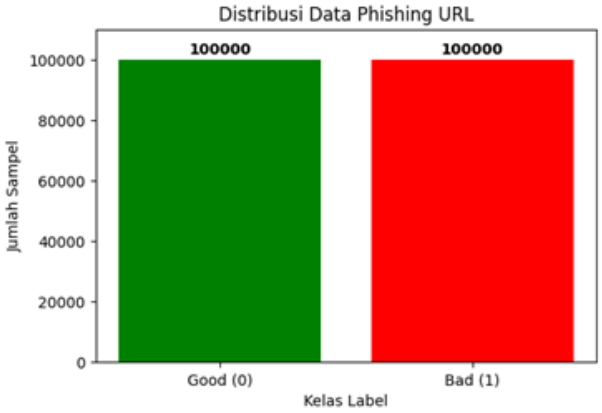


Gambar 1. Tahapan Penelitian

2.2 Persiapan Data

Pada tahap awal, dataset diperoleh dari dua sumber utama yang bersifat terbuka dan dapat diakses publik, yakni Kaggle dan GitHub, yang menyediakan himpunan data dalam format CSV. Setiap entri dalam file tersebut telah dilabeli secara eksplisit sebagai *phishing* atau *legitimate* berdasarkan hasil kurasi dari kontributor dataset sebelumnya. Dataset yang diunduh berasal dari repositori yang telah digunakan secara luas dalam penelitian-penelitian terdahulu terkait deteksi URL *phishing*, sehingga memiliki tingkat reliabilitas yang tinggi. Setelah proses integrasi dan validasi awal, dataset yang digunakan dalam penelitian ini terdiri dari 200.000 entri, dengan proporsi seimbang antara dua kelas: 100.000 URL *phishing* dan 100.000 URL aman. Data ini kemudian dibagi menjadi dua subset menggunakan teknik stratified sampling untuk menjaga distribusi kelas yang setara. Sebanyak 80% dari total data digunakan sebagai data pelatihan (*training set*), sementara sisanya sebanyak 20% digunakan sebagai data pengujian (*testing set*), guna memastikan evaluasi performa model dilakukan secara obyektif dan representatif.

Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443



Gambar 2. Distribusi Data

Pada Gambar 2, terdapat dua kategori utama, yaitu URL yang tergolong aman atau "*good*" dan URL yang termasuk *phishing* atau "*bad*". Jumlah data untuk kedua kategori ini sama banyaknya, yaitu masing-masing sebanyak 100.000 sampel. Berikut hasil dataset yang sudah diberikan label yang disajikan pada Tabel 1.

Tabel 1. Sample Url

No	Url	Label
1	https://www.google.co.id	*Good*
2	https://youtube.com	*Good*
3	https://bantuan12.titviews.my.id	*Bad*

2.3 *Natural Language Processing*

Pemrosesan Bahasa alami (*Natural Language Processing*) merupakan salah satu bidang dalam *Artificial Intelligence* yang berfokus pada pengembangan sistem yang mampu memahami dan memproses bahasa manusia secara alami[17]. Langkah awal dari *Natural Language Processing* yaitu *preprocessing* data yang berfungsi untuk merapikan serta menyiapkan data agar memiliki struktur yang sesuai sebelum digunakan dalam tahap pemrosesan selanjutnya[18]. Tahapan-tahapan dalam proses ini antara lain sebagai berikut:

a. *Case Folding*
   mengubah semua huruf dalam teks menjadi huruf kecil (*lowercase*).
b. *Cleaning*
   Menghapus karakter atau simbol yang tidak dibutuhkan seperti tanda baca, angka, atau karakter khusus. Tujuannya untuk membersihkan teks agar hanya mengandung informasi yang relevan.
c. *Tokenization*
   Pada tahap ini, kalimat diuraikan menjadi potongan-potongan kata yang lebih sederhana melalui proses tokenisasi. Pemisahan ini dilakukan dengan mengenali spasi sebagai batas antar kata, sehingga setiap kata dapat dipisahkan secara efisien.

Tabel 2. Hasil *Preprocessing* Data

URL	*Case Folding*	*Cleaning*	*Tokenization*
https://www.google.co.id	https://www.google.co.id	www google co id	‘www’, ‘google’, ‘co’, ‘id’
https://bantuan12.titviews.my.id	https://bantuan12.titviews.my.id	bantuan12 titviews my id	‘bantuan12’, ‘titviews’, ‘my’, ‘id’

Selanjutnya yaitu ekstraksi fitur url merupakan proses mengambil elemen-elemen penting dari struktur URL untuk dianalisis lebih lanjut. Pada penelitian ini, peneliti menggunakan 21 fitur url beberapa diantaranya Panjang url, jumlah titik dan jumlah slash serta pola-pola teks tertentu yang muncul dalam URL. Menurut penelitian [19] *phishing* website memiliki karakteristik atau ciri-ciri yang dapat dibedakan dengan website yang asli atau *legitimate* website. Salah satu ciri yang relevan adalah panjang URL (URL-*Length*), yang sering kali digunakan untuk membedakan situs *phishing* dari situs sah. Situs *phishing* cenderung memiliki URL yang lebih panjang, rumit, atau mengandung karakter yang mencurigakan. Berikut hasil ekstraksi fitur url yang disajikan pada Tabel 3.

Tabel 3. Hasil Ekstraksi Fitur Url

Url	Panjang url	Jumlah titik	Jumlah slash
https://www.google.co.id	24	3	2
https://bantuan12.titviews.my.id	32	3	2

Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443

Langkah terakhir yaitu ekstraksi fitur menggunakan *Term Frequency-Inverse Document Frequency*, Ekstraksi fitur bertujuan mengubah informasi penting dari data menjadi bentuk numerik agar dapat diproses oleh algoritma *machine learning*. Metode *Term Frequency-Inverse Document Frequency* digunakan untuk menilai pentingnya sebuah kata berdasarkan frekuensinya dalam satu dokumen dan kelangkaannya di seluruh dokumen[20]. Berikut merupakan rumus dari *Term Frequency-Inverse Document Frequency*.

$$\text{TF} = \frac{\text{jumlah kemunculan kata t dalam dokumen d}}{\text{jumlah total kata dalam dokumen d}} \quad (1)$$

$$\text{IDF} = \log\left(\frac{1+\text{N}}{1+\text{df (t)}}\right) + 1 \quad (2)$$

$$\text{TF-IDF} = \text{TF x IDF} \quad (3)$$

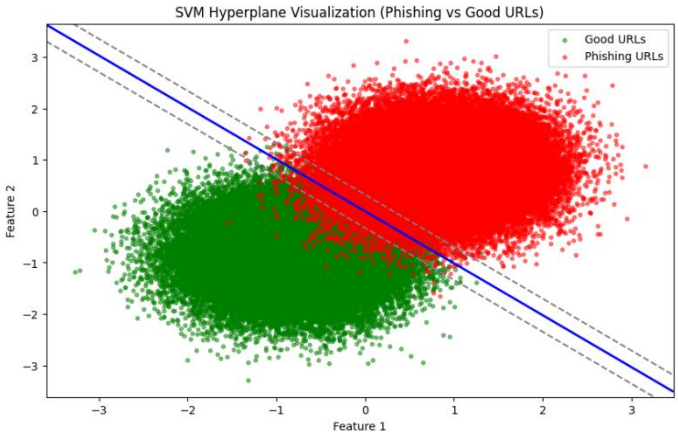
Berdasarkan persamaan 1, 2 dan 3 *Term Frequency* (TF) merepresentasikan frekuensi kemunculan suatu kata (*term*) dalam sebuah dokumen, sedangkan *Inverse Document Frequency* (IDF) menggambarkan tingkat kekhasan kata tersebut dengan mengukur seberapa jarang kata muncul dalam keseluruhan dokumen. Notasi N menyatakan jumlah total dokumen dalam korpus, sementara Df(t) menunjukkan banyaknya dokumen yang memuat *term t* tertentu.

Tabel 4. Hasil *Term Frequency-Inverse Document Frequency*

Url	Bantuan12	id	my	titiviews	google	co	www
https://www.google.co.id	0	0,25	0	0	0,35	0,35	0,35
https://bantuan12.titviews.my.id	0,35	0,25	0,35	0,35	0	0	0

2.4 *Support Vector Machine*

Dataset yang telah direpresentasikan secara numerik digunakan untuk melatih model klasifikasi menggunakan algoritma *Support Vector Machine* (SVM), yang efektif untuk permasalahan klasifikasi dan regresi.Menurut penelitian [21] perbandingan kernel pada algoritma Support Vector Machine (SVM) menunjukkan bahwa kernel linier memberikan akurasi tertinggi sebesar 95,43%. Hasil ini sejalan dengan temuan dalam penelitian ini, di mana kernel linier juga terbukti lebih efektif dalam mendeteksi URL *phishing*. Tujuan utama dari SVM adalah mencari *hyperplane* yang paling optimal untuk memisahkan data ke dalam berbagai kelas dengan jarak pemisah (margin) yang maksimal[22]. Oleh karena itu, dalam penelitian ini, kernel linier dipilih karena memiliki akurasi yang lebih baik dibandingkan dengan kernel lainnya, sehingga model klasifikasi yang dihasilkan dapat lebih efektif dalam membedakan antara URL *phishing* dan URL yang aman berdasarkan fitur-fitur yang telah diekstraksi. Pada gambar 3 menampikan ilustrasi *hyperlane support vector machine*.



Gambar 3. *Hyperlane* SVM

2.4 Evaluasi

Tahap analisa hasil dilakukan untuk mengevaluasi performa sistem dalam mendeteksi URL *phishing*. Metode evaluasi yang digunakan antara lain akurasi, presisi, *recall*, dan F1-*score*, yang dihitung berdasarkan *confusion matrix* dan *classification report* yang diambil dari kombinasi terbaik. Hasil evaluasi ini digunakan untuk menilai sejauh mana model *Support Vector Machine* yang dikembangkan mampu mengklasifikasikan URL secara tepat, serta mengidentifikasi kelebihan dan kekurangan sistem yang diimplementasikan.

a. *Classification Report*
   Classification report merupakan rangkuman dari berbagai metrik yang digunakan untuk mengevaluasi kinerja model klasifikasi dalam *machine learning*[23]. Laporan ini menyajikan informasi secara rinci mengenai kemampuan model dalam memprediksi setiap kelas yang terdapat dalam dataset.
b. *Confusion Matrix*

Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443

Confusion matrix merupakan sebuah tabel yang menggambarkan jumlah prediksi yang benar dan salah yang dilakukan oleh model terhadap data uji[24].

Tabel 5. Confusion Matrix

	Predicted Positive (Bad)	*Predicted Negative (Good)*
Actual Positive (Bad)	*True Positive (TP)*	*False Negative (FN)*
Actual Negative (Good)	*False Positive (FP)*	*True Negative (TN)*

Pada confusion matrix terdapa empat parameter diantaranya yaitu:

1. *Accuracy*
   Akurasi adalah perbandingan antara jumlah prediksi yang benar yang dilakukan oleh sistem dengan total jumlah data yang diuji[25], yang secara matematis dituliskan pada persamaan 4.

$$\text{akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (4)$$

2. *Precision*
   Presisi menunjukkan seberapa besar proporsi URL yang diprediksi sebagai *phishing* oleh sistem, yang ternyata memang benar-benar merupakan URL *phishing*. Berikut merupakan rumus dari presisi.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (5)$$

3. *Recall*
   Recall untuk mengukur berapa banyak dari total *phishing* URL yang berhasil dikenali oleh model. Berikut merupakan rumus untuk menghitung *recall*.

$$\textit{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (6)$$

4. F1-*score*
   F1-*score* merupakan rata-rata harmonik dari nilai presisi dan *recall*. Metrik ini digunakan untuk memberikan keseimbangan antara kedua ukuran tersebut, khususnya dalam situasi di mana presisi dan recall memiliki tingkat kepentingan yang setara.

$$\text{F1-}\textit{Score} = 2 \text{ x } \frac{\textit{Precision x Recall}}{\textit{Precision + Recall}} \quad (7)$$

3. HASIL DAN PEMBAHASAN

3.1 Hasil eksperimen

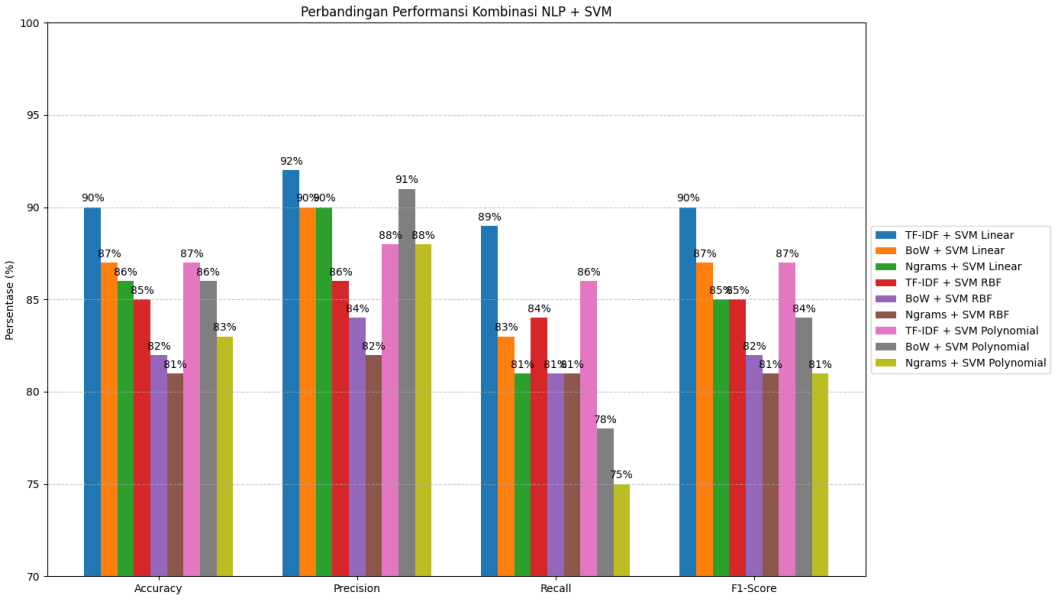
Penelitian ini mengimplementasikan sistem deteksi URL *phishing* dengan memanfaatkan teknik *Natural Language Processing* (NLP) untuk ekstraksi fitur dan algoritma *Support Vector Machine* (SVM) sebagai klasifikasi. Tujuan eksperimen ini untuk menemukan model terbaik dalam mendeteksi URL *phishing*. Ekstraksi fitur dilakukan dengan tiga metode, yaitu *Term Frequency-Inverse Document Frequency*, *Bag of Words*, dan N-grams, yang kemudian diuji dengan beberapa variasi kernel *Support Vector Machine*, yakni linear, *Radial Basis Function* (RBF), dan polynomial, untuk mengevaluasi performa dan menentukan konfigurasi terbaik.

Tabel 6. Hasil eksperimen

Parameter	*Accuracy*	*Precision*	*Recall*	F1-*Score*
TF-IDF + SVM Linear	90%	92%	89%	90%
Bag of Words + SVM Linear	87%	90%	83%	87%
Ngrams + SVM Linear	86%	90%	81%	85%
TF-IDF + SVM RBF	85%	86%	84%	85%
Bag of Words + SVM RBF	82%	84%	81%	82%
Ngrams + SVM RBF	81%	82%	81%	81%
TF-IDF + SVM Polynomial	87%	88%	86%	87%
Bag of Words + SVM Polynomial	85%	91%	78%	84%
Ngrams + SVM Polynomial	82%	88%	75%	81%

Berdasarkan hasil pengujian yang ditampilkan pada Tabel 6, berikut disajikan grafik visualisasi yang menggambarkan perbandingan performa dari berbagai kombinasi teknik *Natural Language Processing* (NLP) dan algoritma *Support Vector Machine* (SVM) yang digunakan dalam penelitian ini.

Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443



Gambar 4. Grafik perbandingan performansi kombinasi NLP dan SVM

Gambar 4 menunjukkan bahwa kombinasi *Term Frequency-Inverse Document Frequency* dengan kernel Linear secara konsisten menghasilkan performa terbaik pada keempat metrik dengan akurasi 90%, *precision* 92%, *recall* 89%, dan F1-*score* 90%, menjadikannya sebagai konfigurasi paling unggul dalam klasifikasi URL *phishing* pada dataset yang digunakan. Sementara itu, teknik *Bag of Words* dan N-grams menunjukkan fluktuasi performa yang lebih besar tergantung pada jenis kernel yang digunakan. Sebagai contoh, kombinasi BoW dengan kernel Polynomial menunjukkan *precision* yang tinggi 91%, namun *recall*-nya menurun cukup signifikan 78%, yang berdampak pula pada penurunan F1-*score*. Hal serupa juga terlihat pada kombinasi N-grams dengan kernel Polynomial, di mana nilai recall dan F1-*score* lebih rendah dibandingkan kombinasi lainnya, visualisasi ini mengidentifikasi pola performa, menggarisbawahi pentingnya pemilihan teknik representasi fitur dan kernel yang tepat, dengan TF-IDF dan kernel Linear memberikan kontribusi signifikan pada kinerja model deteksi *phishing*.

3.2 Evaluasi Model

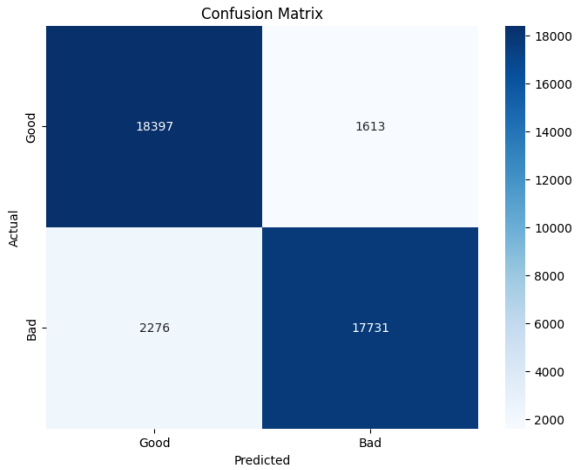
Tabel 7 menampilkan hasil evaluasi model dalam bentuk *classification report*, berdasarkan penerapan pendekatan *Term Frequency-Inverse Document Frequency* untuk ekstraksi fitur serta penggunaan kernel Linear pada algoritma *Support Vector Machine* sebagai metode klasifikasi. Evaluasi dilakukan terhadap dua kelas, yaitu kelas 0 yang merepresentasikan URL *good* dan kelas 1 yang merepresentasikan URL *bad* (*phishing*).

Tabel 7. *Classification Report*

	Precision	Recall	F1-score	Support
0	0.89	0.92	0.90	20010
1	0.92	0.89	0.90	20007
Accuracy			0.90	40017
Macro avg	0.90	0.90	0.90	40017
Weighted avg	0.90	0.90	0.90	40017

Berdasarkan hasil evaluasi, model mencatatkan nilai *precision* sebesar 0.89 untuk kelas 0 dan 0.92 untuk kelas 1. Hal ini menunjukkan bahwa model memiliki tingkat ketepatan yang tinggi dalam mendeteksi URL yang termasuk kategori *phishing*. Sementara itu, recall tercatat 0.92 untuk kelas 0 dan 0.89 untuk kelas 1, yang berarti sistem lebih responsif dalam mengenali URL yang aman, namun tetap menunjukkan performa yang baik dalam mengidentifikasi URL berbahaya.Skor F1 yang diperoleh untuk kedua kelas berada pada angka 0.90, mencerminkan keseimbangan yang proporsional antara ketepatan dan sensitivitas model dalam melakukan klasifikasi. Total jumlah data uji terbagi secara merata pada kedua kelas, masing-masing berjumlah sekitar 20.000 sampel. Hal ini mendukung nilai akurasi keseluruhan sebesar 0.90, yang berarti 90% prediksi yang dilakukan model sesuai dengan label sebenarnya.Selain itu, nilai *macro average* dan *weighted average* untuk seluruh metrik evaluasi (*precision*, *recall*, dan F1-*score*) juga konsisten pada angka 0.90. Keseragaman ini mengindikasikan bahwa model bekerja secara merata pada kedua kelas dan tidak menunjukkan kecenderungan bias terhadap salah satu kategori. Untuk mendukung hasil evaluasi tersebut, pada Gambar 4 menyajikan *Confusion Matrix* yang menggambarkan sebaran prediksi model terhadap data aktual.

Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443



Gambar 5. *Confusion Matrix*

Berdasarkan hasil *confusion matrix* pada Gambar 5, model berhasil mengklasifikasikan 18.397 URL aman secara tepat sebagai *Good*, yang dalam konteks evaluasi disebut sebagai *True Negative* (TN). Sementara itu, terdapat 17.731 URL *phishing* yang juga berhasil dikenali dan diklasifikasikan dengan benar sebagai *Bad*, yang tergolong sebagai *True Positive* (TP). Di sisi lain, sebanyak 1.613 URL aman secara keliru diprediksi sebagai *phishing*, sehingga termasuk dalam kategori *False Positive* (FP). Kesalahan ini berarti model memberikan sinyal palsu terhadap URL yang sebenarnya tidak berbahaya. Selain itu, ditemukan 2.276 URL *phishing* yang tidak terdeteksi dan justru diklasifikasikan sebagai aman. Kasus seperti ini termasuk dalam *False Negative* (FN), yang memiliki implikasi lebih serius karena model gagal mengenali ancaman yang sebenarnya. Berdasarkan *confusion matrix* yang dihasilkan, dilakukan perhitungan manual terhadap metrik evaluasi utama seperti *accuracy*, *precision*, *recall*, dan F1-*score* guna mendukung interpretasi kinerja model secara kuantitatif.

Tabel 8. *Confusion Matrix*

	Predicted Positive (Bad)	*Predicted Negative (Good)*
Actual Positive (Bad)	17.731	2.276
Actual Negative (Good)	1.613	18.397

$$Accuracy = \frac{17731 + 18397}{17731 + 18397 + 1613 + 2276} = 0.9029 = 90.29\ \%$$

$$Precision = \frac{17731}{17731 + 1613} = \frac{17731}{19344} = 0.9166 = 91.66\%$$

$$Recall = \frac{17731}{17731 + 2276} = \frac{17731}{20007} = 0.8862 = 88.62\%$$

$$\text{F1- } score = 2 \text{ x } \frac{0.9166\ x\ 0.8862}{0.9166 + 0.8862} = 0.9012 = 90.12\%$$

Model mencapai akurasi sebesar 90,29%, menggambarkan tingkat ketepatan prediksi yang tinggi dan andal. Nilai *precision* sebesar 91,66% menunjukkan bahwa model mampu meminimalkan kesalahan dalam mengklasifikasikan URL aman. Sementara itu, *recall* sebesar 88,62% mengindikasikan masih terdapat ruang untuk meningkatkan sensitivitas dalam mendeteksi URL *phishing*. Dengan F1-*score* sebesar 90,12%, model berhasil mempertahankan keseimbangan yang optimal antara *precision* dan *recall*, menandakan performa klasifikasi yang konsisten dan efektif.

3.3 Hasil Perbandingan

Pada Bagian ini, akan dibahas perbedaan antara dua pendekatan utama dalam deteksi URL *phishing* menggunakan algoritma *Support Vector Machine* (SVM), yaitu metode yang mengintegrasikan *teknik Natural Language Processing* (NLP) dan metode yang hanya mengandalkan fitur manual tanpa NLP. Perbedaan utama antara metode deteksi URL *phishing* dengan dan tanpa penggunaan *Natural Language Processing* (NLP) terletak pada cara ekstraksi fitur yang digunakan. Metode tanpa NLP mengandalkan fitur manual yang diekstraksi langsung dari struktur dan karakteristik URL, seperti panjang URL, jumlah karakter khusus, dan pola domain mencurigakan. Sedangkan metode dengan NLP memanfaatkan teknik pengolahan bahasa alami, seperti TF-IDF dan n-gram, untuk merepresentasikan URL dalam bentuk fitur teks yang lebih kompleks dan informatif.

Tabel 9. Hasil perbandingan *classification report* dari kombinasi SVM dengan dan tanpa NLP

Parameter	NLP + SVM	Fitur url manual + SVM
Accuracy	90%	71%

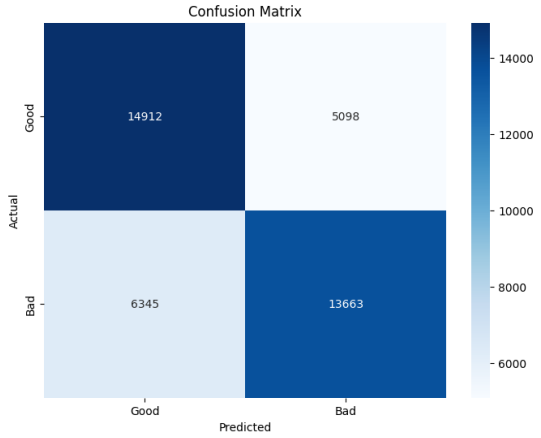
Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443

Parameter	NLP + SVM	Fitur url manual + SVM
Precision	92%	73%
Recall	89%	68%
F1-*score*	90%	70%

Pada Tabel 10 menyajikan hasil dari confusion matrix dengan dan tanpa menggunakan *Natural Language Processsing.*

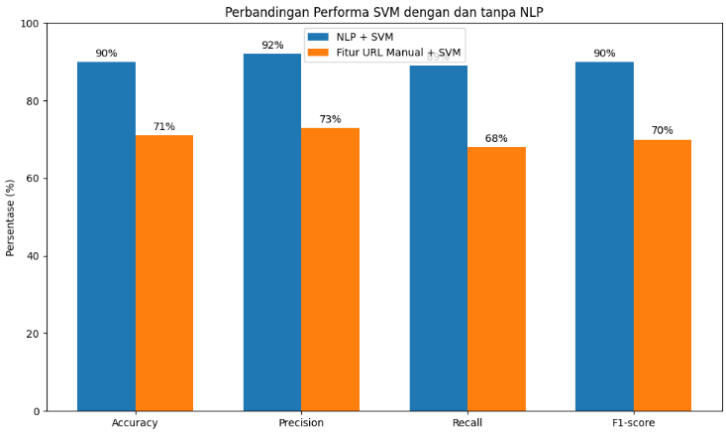
Tabel 10. Hasil perbandingan *confusion matrix* kombinasi SVM dengan dan tanpa NLP

Confusion Matrix	SVM + NLP		Tanpa NLP	
	Predicted Bad	*Predicted Good*	*Predicted Bad*	*Predicted Good*
Actual Bad (Positif)	17.731	2.276	13.663	6.345
Actual Good (Negatif)	1.613	18.379	5.098	14.912



Gambar 6. *Confusion matrix* SVM tanpa NLP

Evaluasi performa model *Support Vector Machine* (SVM) dalam mendeteksi URL *phishing* menunjukkan perbedaan signifikan antara pendekatan yang menggunakan *Natural Language Processing* (NLP) dan yang tidak. Model dengan NLP berhasil mencapai akurasi sebesar 90%, dengan *precision* dan *recall* seimbang di kedua kelas, masing-masing berkisar antara 89% hingga 92%, serta F1-*score* sekitar 0,90. Hal ini didukung oleh *confusion matrix* yang memperlihatkan bahwa dari total URL *phishing* sebenarnya, sebanyak 17.731 berhasil diklasifikasikan dengan benar sebagai *phishing*, sedangkan 2.276 salah diklasifikasikan sebagai bukan *phishing*. Untuk URL bukan *phishing*, 18.397 diprediksi dengan tepat, sementara 1.613 salah diklasifikasikan. Sebaliknya, model tanpa NLP hanya memperoleh akurasi sebesar 71% dengan *precision* dan *recall* yang lebih rendah, yaitu *precision* 73% dan *recall* 68%, serta F1-score sekitar 0,70. *Confusion matrix* pada model ini juga menunjukkan performa yang lebih buruk, di mana hanya 13.663 URL *phishing* yang terdeteksi benar dan 6.345 terlewat, serta 14.912 URL bukan *phishing* yang diprediksi tepat dan 5.098 salah diklasifikasikan sebagai *phishing*. Perbedaan ini mengindikasikan bahwa integrasi NLP dalam ekstraksi fitur memberikan model kemampuan lebih baik dalam mengenali pola kompleks pada URL *phishing* sehingga meningkatkan ketepatan dan sensitivitas deteksi dibandingkan metode berbasis fitur manual tanpa NLP.



Gambar 7. Grafik perbandingan *classification report* SVM dengan dan tanpa NLP

Berdasarkan Gambar 7, dapat disimpulkan bahwa model SVM yang menggabungkan NLP menunjukkan hasil yang jauh lebih baik pada semua metrik evaluasi, yaitu *accuracy*, *precision*, *recall*, dan F1-*score*. Hal ini menandakan

Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443

bahwa penggunaan NLP dalam proses ekstraksi fitur memberikan kontribusi signifikan dalam meningkatkan ketepatan klasifikasi URL *phishing*. Selain itu, *precision* dan *recall* pada model dengan NLP juga lebih tinggi dan seimbang, sehingga model tersebut tidak hanya akurat dalam mendeteksi URL *phishing*, tetapi juga minim kesalahan dalam mengklasifikasikan URL yang aman. F1-*score* yang lebih tinggi pada model dengan NLP memperkuat kesimpulan bahwa keseimbangan antara presisi dan sensitivitas dapat dicapai dengan lebih optimal jika teknik NLP digunakan. Oleh karena itu, penerapan NLP dalam ekstraksi fitur sangat disarankan untuk meningkatkan performa sistem deteksi URL *phishing* menggunakan algoritma SVM kernel linear.

3.4 Perbandingan Hasil dengan Penelitian Sebelumnya

Tabel 11 menyajikan perbandingan antara hasil penelitian ini dengan beberapa penelitian sebelumnya yang juga membahas deteksi *phishing* menggunakan algoritma machine learning. Penelitian ini menggunakan kombinasi metode *Term Frequency-Inverse Document Frequency* (TF-IDF) dan *Support Vector Machine* (SVM) dengan kernel linear, yang terbukti menghasilkan akurasi tertinggi sebesar 90,29%. Hasil ini melampaui penelitian sebelumnya.

Tabel 11. Perbandingan Hasil dengan Penelitian Sebelumnya

Penelitian	Metode yang digunakan	Akurasi
Penelitian ini	TF-IDF + SVM Linear	90,29%
Penelitian oleh Fitra, Sriyanto, dan Zarnelly (2024)	*Decision Tree*	87,04%
Penelitian oleh Arfian, Anindya, Nabila, Keisha, dan Elsya (2024)	KNN	88%
Penelitian oleh Azzam dan Setia (2024)	KNN, *Random Forest*, *Decision Tree*	48,2%,83,4%, 83,3%
Mutiara dan Wiyli (2024)	*CountVectorizer* + SVM Linear	88%

Keunggulan penelitian ini tidak hanya terletak pada akurasi yang lebih tinggi, tetapi juga pada pendekatannya yang lebih komprehensif. Penelitian ini memadukan ekstraksi 21 fitur struktural dari URL, seperti panjang URL dan jumlah titik, dengan pendekatan *Natural Language Processing* (NLP) menggunakan TF-IDF untuk menangkap pola tekstual yang lazim pada URL *phishing*. Selain itu, penggunaan dataset yang seimbang antara URL *phishing* dan legal juga berkontribusi terhadap kestabilan performa model. Sebagian besar penelitian sebelumnya masih bergantung pada metode ekstraksi manual yang terbatas pada struktur URL tanpa melibatkan teknik NLP, sehingga belum mampu menangkap informasi tekstual secara mendalam. Bahkan, beberapa studi hanya berfokus pada kasus spesifik seperti situs DANA Kaget, yang membuat model yang dihasilkan kurang dapat digeneralisasi. Berbeda dengan itu, model dalam penelitian ini dirancang untuk mendeteksi berbagai jenis URL *phishing* secara luas, sehingga lebih adaptif terhadap beragam bentuk serangan siber yang berkembang. Dengan demikian, dapat disimpulkan bahwa integrasi teknik NLP dalam proses ekstraksi fitur sangat efektif dalam meningkatkan performa sistem deteksi *phishing*, khususnya ketika dikombinasikan dengan algoritma SVM kernel linear.

4. KESIMPULAN

Penelitian ini berhasil mengimplementasikan teknik *Natural Language Processing* (NLP) menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF) dan algoritma *Support Vector Machine* (SVM) sebagai model klasifikasi untuk mendeteksi url *Phishing*. Hasil evaluasi dari *confusion matrix* menunjukkan bahwa model yang dibangun mampu mencapai akurasi sebesar 90,29%, *precision* 91,66%, *recall* 88,62%, dan F1-*score* 90,12%, dan pada *classification report* mencapai akurasi 90% *precision* 92% *recall* 89% dan F1-*score* 90%yang menunjukkan kinerja yang andal dan sensitif dalam membedakan antara URL *phishing* dan URL yang aman. Kombinasi teknik NLP dengan SVM kernel linear memberikan performa terbaik dibandingkan kombinasi lain seperti Bag of Words dan N-grams, serta kernel SVM yang berbeda. Hasil ini membuktikan bahwa ekstraksi fitur berbasis NLP memberikan kontribusi signifikan dalam meningkatkan efektivitas deteksi *phishing* dengan mengidentifikasi pola-pola teks yang kompleks dan karakteristik URL *phishing* secara lebih mendalam. Namun demikian, penelitian ini juga menemukan beberapa kelemahan, terutama masih adanya sejumlah *false positive* dan *false negative* yang menunjukkan keterbatasan dalam mendeteksi URL *phishing* yang belum pernah dikenali sebelumnya. Salah satu keterbatasan utama penelitian ini adalah jumlah dan variasi dataset yang digunakan, yang masih terbatas sehingga model belum sepenuhnya adaptif terhadap pola baru dalam URL *phishing*. Untuk penelitian selanjutnya, disarankan untuk memperluas dan memperkaya dataset dengan menambahkan URL *phishing* dari berbagai sumber dan domain yang lebih bervariasi. Selain itu, perlu dieksplorasi algoritma lain atau kombinasi metode yang dapat meningkatkan akurasi dan stabilitas deteksi. Penelitian juga dapat mencakup analisis lebih dalam terhadap fitur-fitur yang relevan serta pendekatan baru dalam pemilihan kernel SVM atau algoritma pembelajaran mesin lainnya. Dengan pengembangan ini, diharapkan sistem dapat menjadi lebih efektif dan adaptif dalam menghadapi ancaman *phishing* yang terus berkembang seiring dengan kemajuan teknologi dan semakin kompleksnya serangan siber.

Building of Informatics, Technology and Science (BITS)
Volume 7, No 1, June 2025 Page: 526−537
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i1.7443

REFERENCES

[1] A. Sudiro, M. D. Ilmawan, and N. V. Puspita, "Pendampingan Terpadu Untuk Maksimalkan Pemasaran Digital Umkm Soto Kudus Kedai Taman Cabang Mojokerto," *DedikasiMU: Journal of Community Service*, vol. 6, no. 4, 2024, doi: https://doi.org/10.30587/dedikasimu.v6i4.8556.

[2] V. A. Windarni, A. F. Nugraha, S. T. A. Ramadhani, D. A. Istiqomah, F. M. Puri, and A. Setiawan, "Deteksi Website *Phishing* Menggunakan Teknik Filter Pada Model Machine Learning," *Information System Journal (INFOS)* , vol. 6, no. 1, 2023, doi: https://doi.org/10.24076/infosjournal.2023v6i01.1268.

[3] Indonesia Anti-*Phishing* Data Exchange (IDADX), "Laporan Aktivitas Phising Domain~.ID," 2023. Accessed: May 18, 2025. [Online]. Available: https://surl.li/vfofxr

[4] Indonesia Anti-*Phishing* Data Exchange (IDADX), "Laporan Aktivitas Abuse Domain .Id Indonesia Domain Abuse Data Exchange," 2024. Accessed: May 26, 2025. [Online]. Available: https://surli.cc/oojpve

[5] A. Erikha and Z. Arifin Hoesein, "Strategi Pencegahan Kebocoran Data Pribadi melalui Peran Kominfo dan Gerakan Siberkreasi dalam Edukasi Digital," *Jurnal Retentum*, vol. 7, no. 1, 2025, doi: 10.46930.

[6] Y. Yuliana, "The Importance Of Cybersecurity Awareness For Children," *Lampung Journal of International Law*, vol. 4, no. 1, pp. 41–48, Jun. 2022, doi: 10.25041/lajil.v4i1.2526.

[7] A. F. Mahmud and S. Wirawan, "Deteksi *Phishing* Website menggunakan Machine Learning Metode Klasifikasi," *Sistemasi: Jurnal Sistem Informasi*, vol. 13, no. 4, 2024, doi: https://doi.org/10.32520/stmsi.v13i4.

[8] C. A. Nurhaliza Agustina, R. Novita, Mustakim, and N. E. Rozanda, "The Implementation of TF-IDF and Word2Vec on Booster Vaccine Sentiment Analysis Using Support Vector Machine Algorithm," in *Procedia Computer Science*, Elsevier B.V., 2024, pp. 156–163. doi: 10.1016/j.procs.2024.02.162.

[9] M. Z. Naeem, F. Rustam, A. Mehmood, Mui-zzud-din, I. Ashraf, and G. S. Choi, "Classification of movie reviews using term frequency-inverse document frequency and optimized machine learning algorithms," *PeerJ Comput Sci*, vol. 8, 2022, doi: 10.7717/PEERJ-CS.914.

[10] B. Alshawi, "Comparison of SVM kernels in Credit Card Fraud Detection using GANs," *International Journal of Advanced Computer Science and Application (IJACSA)*, vol. 15, no. 1, 2024, doi: 10.14569/ijacsa.2024.0150131.

[11] F. S. Salam Nagalay, "Analisis Penerapan Algoritma Decision Tree Dalam Keamanan Siber Untuk Kelasifikasi Situs Website *Phishing*," *Jurnal Ilmiah Rekayasa dan Manajemen Sistem Informasi*, vol. 10, no. 1, pp. 1–8, 2024, doi: http://dx.doi.org/10.24014/rmsi.v10i1.28401.

[12] A. Jauhar Himawan, A. Meyla Kartika Sari, N. Agatha Parsa, K. Sabilah Putri Hermansyah, and E. Sabrina Dea Rizki, "Penerapan Metode K-Nearest Neighbors dalam Mendeteksi Website *Phishing*," *COREAI*, vol. 5, no. 2, 2024, doi: 10.33650/coreai.v5i2.10484.

[13] M. Vebriani and W. Yustanti, "Klasifikasi Deteksi Link Phising DANA Kaget Menggunakan Metode Support Vector Machine Berbasis Website," *Journal of Informatics and Computer Science*, vol. 06, no. 2, 2024, doi: https://doi.org/10.26740/jinacs.v6n02.p408-416.

[14] N. H. Shaker and B. N. Dhannoon, "Word embedding for detecting cyberbullying based on recurrent neural networks," *IAES International Journal of Artificial Intelligence*, vol. 13, no. 1, pp. 500–508, Mar. 2024, doi: 10.11591/ijai.v13.i1.pp500-508.

[15] I. P. Ramayasa, I. G. A. Des Saryanti, I. K. Dharmendra, and Edwar, "Perbandingan Metode Vektorisasi Pada Analisa Sentiment, Studi Kasus : Cyberbullying Pada Komentar Instagram," *Jurnal Teknologi Informasi Dan Komputer*, vol. 9, no. 5, 2023, doi: https://doi.org/10.36002/jutik.v9i5.2645.

[16] D. E. Cahyani and I. Patasik, "Performance comparison of TF-IDF and Word2Vec models for emotion text classification," *Bulletin of Electrical Engineering and Informatics* , vol. 10, no. 5, 2021, doi: 10.11591/eei.v10i5.3157.

[17] R. Danar Dana, Mulyawan, A. Bahtiar, and I. Ali, *Dasar Dasar Natural Language Processing (NLP)* . Minhaj Pustaka, 2024. [Online]. Available: https://surl.lu/gpfpax

[18] A. N. Putri, A. Aryanti, and S. Soim, "Implementasi Algoritma SVM Non-Linear Pada Klasifikasi Analisis Sentimen Perkembangan AI di Sektor Pendidikan," *Technology and Science (BITS)*, vol. 6, no. 2, 2024, doi: 10.47065/bits.v6i2.5522.

[19] S. Shabudin, N. S. Sani, K. A. Z. Ariffin, and M. Aliff, "Feature Selection for *Phishing* Website Classification," *Int J Adv Comput Sci Appl*, vol. 11, no. 4, 2020, doi: https://doi.org/10.14569/ijacsa.2020.0110477.

[20] M. R. Sudrajat and M. Zakariyah, "Penerapan Natural Language Processing dan Machine Learning untuk Prediksi Stres Siswa SMA Berdasarkan Analisis Teks," *Building of Informatics, Technology and Science (BITS)*, vol. 6, no. 3, Dec. 2024, doi: 10.47065/bits.v6i3.6180.

[21] J. Anggraini and D. Alita, "Implementasi Metode SVM Pada Sentimen Analisis Terhadap Pemilihan Presiden (Pilpres) 2024 Di Twitter," *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 9, no. 2, pp. 102–111, Aug. 2024, doi: 10.30591/jpit.v9i2.6560.

[22] panji bintoro, ratnasari, edy wihardjo, pratiwi putri indah, and andi asari, *Pengantar Machine Learning*. PT MAFY MEDIA LITERASI INDONESIA, 2024. [Online]. Available: https://surli.cc/qqlrwg

[23] M. L. B. Permadi and R. Gumilang, "Penerapan Algoritma CNN (Convolutional Neural Network) Untuk Deteksi Dan Klasifikasi Target Militer Berdasarkan Citra Satelit," *SOSTECH Jurnal sosial dan teknologi*, vol. 4, no. 2, 2024, doi: https://doi.org/10.59188/jurnalsostech.v4i2.1138.

[24] D. Normawati and S. A. Prayogi, "Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," *J-SAKTI Jurnal Sains Komputer dan Informatika*, vol. 5, no. 2, 2021, doi: http://dx.doi.org/10.30645/j-sakti.v5i2.369.

[25] A. Ramadhan, Lindawati, and M. M. Rose, "Komparasi Algoritma Neural Network dan K-Nearest Neighbor Dalam Mendeteksi Malware Android," *Building of Informatics, Technology and Science (BITS)*, vol. 5, no. 1, Jun. 2023, doi: https://doi.org/10.47065/bits.v5i1.3538.