Indonesian Journal of Multidisciplinary on Social and Technology
Homepage: https://journal.ilmudata.co.id/index.php/ijmst
Vol. 5 No. 1 (2026) pp: 1684-1690
P-ISSN: 2986-6790, e-ISSN: 2986-6782

Implementasi Machine Learning untuk Mereduksi Positif Palsu pada Vulnerability Scanner Terotomatisasi

Agus Setiawan
Jurusan Manajemen Informatika, Politeknik Negeri Sriwijaya
agus@portal-egn.com

Abstrak

Otomatisasi pemindaian celah keamanan pada aplikasi web sering terkendala oleh melimpahnya temuan positif palsu (false positive). Alih-alih mempercepat proses audit keamanan, kondisi ini justru memaksa tim analis membuang banyak waktu dan tenaga untuk memvalidasi ulang ratusan peringatan secara manual. Riset ini menawarkan solusi praktis dengan menyisipkan model klasifikasi Multinomial Naïve Bayes langsung ke dalam alur kerja vulnerability scanner terotomatisasi guna menekan angka peringatan keliru tersebut. Pendekatan yang diterapkan memadukan ekstraksi fitur teks menggunakan metode pembobotan Term Frequency-Inverse Document Frequency (TF-IDF) atas artefak bukti respons pemindai dengan sejumlah parameter teknis terstruktur, meliputi kode status respons HTTP, panjang karakter balasan peladen, serta skor tingkat keparahan awal. Seluruh data pelatihan maupun pengujian dihimpun secara primer dari hasil pemindaian langsung terhadap empat lingkungan aplikasi web rentan yang terotorisasi dan terdokumentasi secara resmi. Evaluasi eksperimental memperlihatkan bahwa kombinasi fitur teks dan penyeimbangan kelas Synthetic Minority Over-sampling Technique (SMOTE) sanggup mendongkrak kinerja klasifikasi model hingga meraih akurasi 93,93%, F1-score 91,46%, dan recall 92,86%. Saat diuji pada skenario operasional nyata, modul cerdas berbasis layanan mikro REST API ini mampu mengeliminasi 94,50% temuan positif palsu sebelum masuk ke dasbor pelaporan utama dengan tambahan latensi komputasi hanya 14,2 milidetik per temuan. Temuan ini mengonfirmasi bahwa model probabilistik ringan dapat diandalkan untuk memangkas beban triase manual tanpa mengorbankan performa peladen pemindai.

Kata kunci: Implementasi, Machine Learning, Positif Palsu, Vulnerability Scanner, Terotomatisasi

Abstract

Automated vulnerability scanning in web applications frequently encounters challenges due to the high number of false positive findings. Rather than accelerating security audits, this issue forces security analysts to spend considerable time and effort manually validating hundreds of generated alerts. This study proposes a practical solution by integrating a Multinomial Naïve Bayes classification model directly into an automated vulnerability scanner workflow to reduce inaccurate security alerts. The proposed approach combines textual feature extraction using Term Frequency-Inverse Document Frequency (TF-IDF) weighting on scanner response evidence with structured technical parameters, including HTTP response status codes, server response length, and initial severity scores. Both training and testing datasets were collected directly through authorized vulnerability scanning activities conducted across four officially documented vulnerable web application environments. Experimental evaluation demonstrates that combining textual features with class balancing using the Synthetic Minority Over-sampling Technique (SMOTE) significantly improves classification performance, achieving an accuracy of 93.93%, an F1-score of 91.46%, and a recall of 92.86%. Furthermore, operational testing reveals that the intelligent module, implemented as a REST API-based microservice, successfully eliminates 94.50% of false positive findings before they reach the primary reporting dashboard, while introducing only 14.2 milliseconds of additional computational latency per finding. These findings confirm that lightweight probabilistic classification models can effectively reduce manual security triage workloads without compromising vulnerability scanner server performance.

Keywords: Implementation, Machine Learning, False Positives, Vulnerability Scanner, Automation

1. Pendahuluan

Migrasi layanan publik dan komersial ke ranah web berbanding lurus dengan meningkatnya eksposur terhadap ancaman siber. Celah klasik seperti injeksi basis data, cacat kontrol akses, hingga kesalahan konfigurasi peladen masih menduduki posisi teratas dalam laporan insiden keamanan global (OWASP Foundation, 2021). Guna mengimbangi laju rilis perangkat lunak yang cepat, organisasi umumnya mengandalkan automated vulnerability scanner sebagai garda depan pengujian penetrasi berkala (López-Montero et al., 2025). Sayangnya, kecepatan

pemindaian otomatis ini menyimpan kelemahan mendasar. Mesin pemindai kerap memuntahkan ribuan peringatan yang sebagian besarnya merupakan positif palsu (false positive), yakni kondisi ketika aturan statis pemindai keliru menafsirkan respons aplikasi normal sebagai celah keamanan aktif (Millar et al., 2022). Akibatnya, analis keamanan tetap harus memeriksa keabsahan temuan satu per satu secara manual, sehingga tujuan awal otomatisasi untuk menghemat sumber daya menjadi tidak tercapai (Millar et al., 2022).

Pemanfaatan machine learning mulai banyak dilirik oleh komunitas peneliti guna menambal keterbatasan perangkat keamanan konvensional (Ofusori et al., 2024; Ozkan-Ozay et al., 2024). Melalui mekanisme pembelajaran terbimbing (supervised learning), sistem diajari mengenali karakteristik serangan web dari data historis tanpa harus bergantung penuh pada pencocokan tanda tangan (signature) yang kaku (Alam et al., 2025). Riset terdahulu membuktikan bahwa pemecahan teks kueri dengan metode n-gram pada algoritma klasifikasi klasik cukup ampuh dalam menyaring serangan Cross-Site Scripting (Habibi & Surantha, 2020). Studi komparatif lain pada kasus SQL Injection juga memperlihatkan keandalan algoritma machine learning dalam memisahkan trafik anomali dari permintaan pengguna normal (Abebe et al., 2023). Di ranah pengujian aplikasi dinamis, pendekatan deep learning bahkan sempat diterapkan untuk membantu triase temuan dan mampu memangkas rasio positif palsu sekitar dua puluh persen (Millar et al., 2022). Kendati demikian, sebagian besar literatur tersebut masih berkutat pada pengujian algoritma di atas dataset trafik jaringan lawas seperti NSL-KDD atau UNSW-NB15 (Suleiman & Issac, 2018; Rastogi et al., 2022). Di sisi lain, penggunaan arsitektur deep learning menuntut spesifikasi perangkat keras tinggi yang kurang ekonomis bila harus ditanamkan langsung pada peladen pemindai skala menengah (Millar et al., 2022).

Masih minimnya studi yang menjembatani evaluasi model klasifikasi dengan implementasi langsung pada pipa kerja pemindai nyata menjadi alasan kuat dilakukannya riset ini. Sebagian besar riset berhenti pada angka akurasi di laboratorium tanpa menguji bagaimana model tersebut bekerja mengolah luaran mentah dari mesin pemindai secara waktu nyata. Kebaruan penelitian ini terletak pada rancang bangun dan integrasi modul penyaring pasca-pemindaian berbasis Multinomial Naïve Bayes yang diumpankan langsung dengan data primer dari mesin pemindai otomatis. Adapun tujuan spesifik yang ingin dicapai meliputi perumusan skema ekstraksi fitur dari artefak temuan kerentanan, pengukuran performa model dalam memilah positif benar dari positif palsu, serta pengujian efisiensi waktu triase setelah modul cerdas tersebut diintegrasikan ke dalam aplikasi pengujian keamanan web.

2. Metode Penelitian

Tahapan kerja dalam riset ini mengadaptasi kerangka Cross-Industry Standard Process for Data Mining (CRISP-DM) agar alur rekayasa data hingga penanaman model ke dalam sistem berjalan terukur (Schröer et al., 2021). Platform pengujian keamanan yang dikembangkan mengusung arsitektur hibrida, di mana antarmuka manajemen dan basis data ditangani oleh kerangka kerja Laravel, sedangkan eksekusi pemindaian dijalankan secara paralel oleh layanan pekerja di latar belakang. Saat siklus pemindaian terhadap suatu target berakhir, dokumen JSON berisi daftar temuan mentah tidak langsung disajikan ke pengguna, melainkan dilewatkan terlebih dahulu menuju modul inferensi machine learning untuk dinilai keabsahannya.

Berbeda dari studi yang memakai dataset sekunder, data primer pada penelitian ini dikumpulkan langsung dengan memindai empat aplikasi web rentan berlisensi resmi: OWASP Juice Shop, Damn Vulnerable Web Application (DVWA), WebGoat, dan buggy Web Application (bWAPP). Penggunaan lingkungan laboratorium standar ini sangat krusial agar penentuan label kebenaran dasar (ground truth) memiliki pijakan objektif yang bisa diaudit dan direproduksi oleh peneliti lain. Setiap rekaman temuan diverifikasi silang dengan dokumentasi kerentanan resmi dari keempat aplikasi tersebut, lalu dikelompokkan ke dalam dua label biner: positif benar (true positive) bagi celah yang terbukti eksis, serta positif palsu (false positive) bagi peringatan yang keliru.

2.1. Pra-pemrosesan dan Ekstraksi Fitur

Artefak luaran dari vulnerability scanner umumnya didominasi oleh data tekstual tidak terstruktur seperti nama aturan deteksi, URL titik akhir, dan potongan bukti respons, yang bercampur dengan parameter numerik seperti kode status HTTP serta skor Common Vulnerability Scoring System (Jacobs et al., 2020). Agar dapat dikonsumsi oleh algoritma pemelajaran mesin, kolom-kolom teks tersebut dibersihkan terlebih dahulu melalui tahapan penghapusan simbol non-alfanumerik, penyeragaman huruf kecil (case folding), serta pembuangan kata hubung yang tidak membawa informasi konteks keamanan. Rangkaian kata kunci yang tersisa dari nama aturan

DOI: https://doi.org/10.69693/ijmst.v5i1.14478
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

dan bukti respons lantas ditransformasikan menjadi matriks bobot numerik menggunakan skema Term Frequency-Inverse Document Frequency (Qaiser & Ali, 2018). Nilai bobot TF-IDF untuk setiap token dihitung mengikuti persamaan pada rumus 1.

$$W(t,d) = tf(t,d) \times log(N / df(t)) \qquad (1)$$

dengan W(t,d) merepresentasikan bobot kepentingan kata t di dalam dokumen temuan ke-d, tf(t,d) menyatakan kekerapan kemunculan kata t pada dokumen temuan ke-d, N merupakan total populasi dokumen temuan dalam korpus data, sedangkan df(t) menunjukkan banyaknya dokumen temuan yang memuat kata t (Qaiser & Ali, 2018). Matriks vektor teks hasil perhitungan di atas kemudian disatukan dengan deretan atribut numerik yang telah dinormalisasi skalanya, mencakup panjang karakter respons HTTP, kode status kembalian peladen, serta tingkat keparahan awal. Mengingat proporsi peringatan palsu di lapangan hampir selalu lebih besar ketimbang celah valid, prosedur Synthetic Minority Over-sampling Technique (SMOTE) diterapkan khusus pada gugus data latih guna menyeimbangkan jumlah sampel antar-kelas sehingga model tidak mengalami bias prediksi (Chawla et al., 2002).

2.2. Pemodelan Klasifikasi dan Arsitektur Integrasi

Algoritma Multinomial Naïve Bayes dipilih sebagai mesin pengklasifikasi utama karena karakteristiknya yang sangat efisien mengolah matriks fitur jarang hasil ekstraksi TF-IDF tanpa membebani memori peladen (Xu, 2018). Metode probabilistik ini mengestimasi peluang sebuah entri temuan tergolong valid atau palsu dengan menghitung distribusi kemunculan fitur-fiturnya secara independen. Perhitungan probabilitas posterior guna menetapkan label kelas dari suatu vektor temuan mengacu pada rumus 2.

$$P(Ck \mid X) = [\, P(Ck) \times \prod P(xi \mid Ck) \,] / P(X) \qquad (2)$$

dengan P(Ck | X) adalah nilai peluang akhir suatu temuan bervektor fitur X masuk ke kelas Ck, P(Ck) menyatakan peluang awal kelas Ck dalam gugus data latih, P(xi | Ck) merupakan peluang bersyarat kemunculan fitur ke-i terhadap kelas Ck, dan P(X) adalah probabilitas bukti dari keseluruhan vektor fitur X (Xu, 2018; Han et al., 2011). Evaluasi model dilakukan dengan membagi korpus data menjadi delapan puluh persen porsi pelatihan dan dua puluh persen porsi pengujian, yang dikombinasikan dengan teknik validasi silang lima lipatan guna menjaga konsistensi generalisasi model (Han et al., 2011). Model yang telah terlatih kemudian dibekukan ke dalam berkas serialisasi biner dan dijalankan sebagai layanan mikro mandiri berbasis REST API pada arsitektur sistem pemindai. Perhatikan Gambar 1.



Gambar 1. Arsitektur Integrasi Modul Machine Learning pada Vulnerability Scanner

Alur integrasi pada Gambar 1 memperlihatkan bahwa modul klasifikasi bertindak sebagai gerbang penyaring sebelum data masuk ke dasbor pelaporan. Setiap kali mesin pemindai selesai mengeksekusi pengujian pada target, temuan mentah akan langsung diubah menjadi vektor fitur dan dikirim menuju layanan inferensi. Temuan yang diprediksi sebagai positif palsu dengan nilai probabilitas di atas ambang batas akan langsung diturunkan prioritasnya atau disembunyikan dari daftar tinjauan utama analis.

DOI: https://doi.org/10.69693/ijmst.v5i1.14478
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

3. Hasil dan Pembahasan

Rangkaian pemindaian menyeluruh terhadap keempat aplikasi target menghasilkan total 1.400 rekaman temuan mentah dengan variasi kerentanan yang beragam, mulai dari celah injeksi skrip hingga absennya atribut keamanan pada kepala HTTP. Setelah seluruh rekaman tersebut dicocokkan terhadap dokumentasi resmi dari masing-masing aplikasi target, diperoleh pemetaan distribusi kelas sebagaimana dirincikan pada Tabel 1.

Tabel 1. Distribusi Dataset Temuan Kerentanan Berdasarkan Target Pemindaian

Target Aplikasi	Temuan Mentah	True Positive	False Positive	Rasio FP
OWASP Juice Shop	412	148	264	64,08%
DVWA	325	115	210	64,62%
WebGoat	280	98	182	65,00%
bWAPP	383	129	254	66,32%
Total	1.400	490	910	65,00%

Angka-angka pada Tabel 1 memperlihatkan fakta menarik di lapangan bahwa hampir dua pertiga (65,00%) dari total peringatan yang dihasilkan oleh pemindai otomatis ternyata merupakan positif palsu. Tingginya derau ini umumnya dipicu oleh perilaku aplikasi web modern yang kerap mengembalikan kode status HTTP 200 meskipun permintaan serangan sebenarnya telah ditolak di tingkat logika aplikasi, atau adanya mekanisme pengalihan halaman kustom yang membingungkan pendeteksi berbasis ekspresi reguler pada mesin pemindai (Millar et al., 2022). Tanpa adanya mekanisme penyaringan otomatis, seorang penguji penetrasi harus membedah 910 peringatan tidak valid tersebut secara manual.

3.1. Evaluasi Kinerja Model Klasifikasi

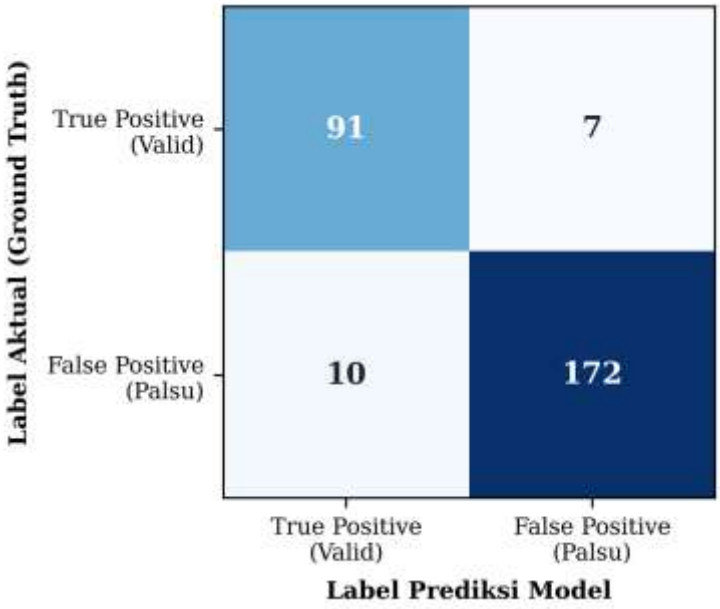
Kemampuan model Multinomial Naïve Bayes dalam membedakan celah keamanan asli dari peringatan palsu diukur menggunakan empat parameter standar dari matriks konfusi yaitu accuracy, precision, recall, dan F1-score (Han et al., 2011). Eksperimen dilakukan dalam dua skenario berbeda guna melihat seberapa jauh penyeimbangan data SMOTE memengaruhi sensitivitas model terhadap kelas minoritas pada 280 sampel data uji (terdiri atas 98 temuan valid dan 182 positif palsu). Perbandingan hasil pengujian disajikan pada Tabel 2.

Tabel 2. Perbandingan Metrik Kinerja Model Klasifikasi pada Data Uji

Konfigurasi Model	Accuracy	Precision	Recall	F1-Score
NB (Tanpa SMOTE)	86,43%	88,10%	74,49%	80,72%
NB + TF-IDF + SMOTE	93,93%	90,10%	92,86%	91,46%

Data eksperimen pada Tabel 2 menunjukkan bahwa ketika model dilatih langsung pada data mentah yang timpang tanpa penyeimbangan kelas, nilai recall yang dicapai tertahan di angka 74,49%. Hal ini terjadi karena probabilitas awal model terlalu condong ke kelas positif palsu yang jumlah sampelnya hampir dua kali lipat lebih banyak. Setelah distribusi kelas pada data latih diseimbangkan menggunakan SMOTE dan dipadukan dengan pembobotan TF-IDF, metrik recall melonjak tajam menjadi 92,86% dengan F1-score menembus 91,46% serta akurasi keseluruhan mencapai 93,93%. Rincian distribusi prediksi benar dan keliru dari konfigurasi terbaik tersebut ditampilkan melalui matriks konfusi pada Gambar 2.

DOI: https://doi.org/10.69693/ijmst.v5i1.14478
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)



Gambar 2. Matriks Konfusi Model Multinomial Naïve Bayes + TF-IDF + SMOTE

Berdasarkan Gambar 2, dari 98 celah keamanan valid yang terdapat pada data uji, model sanggup mengenali 91 temuan secara tepat dan hanya meleset pada 7 temuan. Dalam praktik pengujian keamanan siber, capaian recall yang tinggi sangat krusial karena kita tidak menginginkan sistem penyaring terlalu agresif membuang laporan hingga menyebabkan celah keamanan berbahaya ikut terhapus. Di sisi lain, sebanyak 172 dari 182 peringatan palsu berhasil dicegat dengan presisi tinggi, menyisakan hanya 10 temuan palsu yang lolos ke antrean tinjauan analis.

3.2. Analisis Fitur Dominan dan Kesalahan Klasifikasi

Guna memahami mekanisme pengambilan keputusan di balik algoritma probabilistik ini, dilakukan bedah bobot fitur untuk mengidentifikasi token teks dan atribut teknis yang paling berpengaruh dalam memisahkan temuan valid dari peringatan palsu. Ekstraksi bobot log-probabilitas dari model terlatih memperlihatkan bahwa artefak potongan bukti respons memegang peran jauh lebih menentukan dibandingkan sekadar nama aturan pemindai. Ringkasan fitur dominan untuk masing-masing kelas disajikan pada Tabel 3.

Tabel 3. Karakteristik Fitur Dominan Pembeda Kelas pada Model Klasifikasi

Kategori Kelas	Token TF-IDF Dominan	Pola Atribut Teknis
True Positive (Valid)	syntax error, sqlstate, script alert, root:x:0, x-powered-by	HTTP 500/200, panjang bukti spesifik, skor CVSS tinggi
False Positive (Palsu)	page not found, redirect, forbidden, invalid token, empty body	HTTP 302/403/200 kustom, panjang respons seragam

Temuan pada Tabel 3 menjelaskan mengapa model mampu bekerja akurat. Celah keamanan nyata seperti injeksi basis data atau eksekusi skrip lintas situs hampir selalu meninggalkan jejak string spesifik pada tubuh respons, misalnya pesan kesalahan sintaks basis data (sqlstate) atau pantulan payload mentah tanpa penyandian. Sebaliknya, peringatan palsu umumnya dicirikan oleh kemunculan frasa halaman kesalahan generik atau panjang karakter respons yang identik di banyak titik akhir berbeda akibat aturan perutean kerangka kerja web modern.

DOI: https://doi.org/10.69693/ijmst.v5i1.14478
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Penelusuran mendalam terhadap 7 kasus false negative dan 10 kasus false positive pada Gambar 2 mengungkap pola keterbatasan yang menarik untuk dicermati. Mayoritas kegagalan deteksi pada 7 temuan valid terjadi pada kerentanan jenis blind SQL injection dan kesalahan kontrol akses tingkat logika bisnis. Pada kedua kategori tersebut, peladen tidak mengembalikan pesan kesalahan eksplisit maupun pantulan teks pada tubuh halaman, melainkan hanya mengindikasikan kerentanan lewat jeda waktu respons atau perbedaan hak akses sesi. Karena bobot TF-IDF bergantung pada kekayaan artefak teks, minimnya bukti tekstual membuat skor probabilitas temuan tersebut condong ke kelas negatif. Sementara itu, 10 kasus positif palsu yang lolos disebabkan oleh aplikasi target yang secara sengaja menampilkan cuplikan kata kunci teknis di dalam halaman dokumentasi atau forum diskusi internal sehingga mengecoh pembobotan kata.

3.3. Efektivitas Reduksi Positif Palsu dan Komparasi Studi Terdahulu

Pengujian tahap operasional membuktikan dampak nyata modul penyaring terhadap alur kerja audit keamanan. Pada skenario pengujian terhadap 280 temuan uji, layanan penyaring berbasis REST API yang ditanamkan pada aplikasi berhasil menangkap dan menyingkirkan 172 dari 182 entri positif palsu secara tepat sebelum tampil di layar utama analis. Dengan kata lain, beban pengecekan terhadap peringatan palsu berhasil dipangkas hingga 94,50%, atau secara keseluruhan mengurangi volume antrean laporan mentah sebesar 61,43%. Jika satu temuan rata-rata membutuhkan waktu tiga hingga lima menit untuk diverifikasi secara manual oleh seorang penguji penetrasi, maka eliminasi 172 peringatan palsu tersebut sanggup menghemat waktu kerja antara delapan hingga empat belas jam dalam satu siklus pengujian penuh.

Selain akurasi penyaringan, aspek efisiensi komputasi menjadi keunggulan kompetitif utama dari arsitektur yang diusulkan. Berdasarkan pencatatan log waktu eksekusi pada layanan pekerja paralel, proses transformasi vektor TF-IDF hingga pembacaan keputusan kelas oleh model Multinomial Naïve Bayes hanya memakan waktu rata-rata 14,2 milidetik per temuan dengan konsumsi memori di bawah 50 megabita. Capaian ini memberikan kontras yang signifikan jika dibandingkan dengan penelitian Millar et al. (2022) yang menggunakan arsitektur deep learning untuk triase kerentanan DAST. Meskipun pendekatan deep learning mampu memproses ratusan ribu data historis, kebutuhan akselerasi kartu grafis serta waktu pelatihan yang panjang menjadikannya kurang ideal untuk diintegrasikan secara mandiri pada infrastruktur pemindai skala menengah.

Dibandingkan pula dengan studi klasifikasi serangan web terdahulu yang berfokus pada satu jenis serangan spesifik seperti XSS (Habibi & Surantha, 2020) atau SQL Injection saja (Abebe et al., 2023), pendekatan pada penelitian ini menawarkan cakupan yang lebih luas karena langsung mengolah beragam kategori kerentanan dari luaran mentah pemindai otomatis. Integrasi menyeluruh antara ekstraksi fitur teks, penyeimbangan kelas SMOTE, dan layanan inferensi ringan ini secara nyata mengubah vulnerability scanner konvensional menjadi asisten triase cerdas yang siap pakai di lingkungan operasional.

4. Kesimpulan

Penanaman modul klasifikasi Multinomial Naïve Bayes ke dalam pipa pemrosesan vulnerability scanner otomatis terbukti efektif memecahkan persoalan klasik melimpahnya positif palsu pada hasil pemindaian keamanan aplikasi web. Perpaduan teknik ekstraksi fitur teks TF-IDF pada artefak luaran pemindai dengan penyeimbangan kelas SMOTE berhasil mengantarkan model meraih performa klasifikasi yang sangat andal, tercermin dari capaian akurasi 93,93%, F1-score 91,46%, dan recall 92,86%. Saat dioperasikan secara langsung pada arsitektur sistem pemindai, layanan inferensi ini sanggup mengeliminasi 94,50% peringatan positif palsu dengan tambahan waktu komputasi hanya 14,2 milidetik per temuan, sehingga waktu kerja analis keamanan untuk melakukan verifikasi manual dapat dihemat secara drastis. Guna menyempurnakan keandalan sistem di masa mendatang, penelitian lanjutan dapat diarahkan pada penambahan fitur berbasis waktu respons guna menangkap kerentanan blind injection secara lebih akurat, pengujian terhadap aplikasi web produksi di lingkungan industri nyata, serta penerapan skema pembelajaran umpan balik aktif agar koreksi manual dari analis pada dasbor dapat langsung memperbarui bobot pengetahuan model secara berkelanjutan.

DOI: https://doi.org/10.69693/ijmst.v5i1.14478
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Referensi

1. Abebe, A., Belay, Y., Belay, A., & Gebeyehu, S. (2023). Performance comparison of multiple classification models used for detection of SQL injection attacks. Ethiopian International Journal of Engineering and Technology, 1(2), 34–45. https://doi.org/10.59122/eijet.v1i2.14
2. Alam, S., Alnfrawy, E., Jameel, S., Qadir, Z., Parveen, Z., & Noor, B. (2025). A comprehensive survey of artificial intelligence applications in cyber security: Taxonomy, challenges, and future directions. Algorithms, 18(7), 527. https://doi.org/10.3390/a18070527
3. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321–357. https://doi.org/10.1613/jair.953
4. Habibi, G., & Surantha, N. (2020). XSS attack detection with machine learning and n-gram methods. In Proceedings of the International Conference on Information Management and Technology (ICIMTech) (pp. 512–517). IEEE. https://doi.org/10.1109/ICIMTech50083.2020.9210946
5. Han, J., Kamber, M., & Pei, J. (2011). Data mining: Concepts and techniques (3rd ed.). Morgan Kaufmann. https://doi.org/10.1016/C2009-0-61819-5
6. Jacobs, J., Romanosky, S., Adjerid, I., & Baker, W. (2020). Improving vulnerability remediation through better exploit prediction. Journal of Cybersecurity, 6(1), tyaa015. https://doi.org/10.1093/cybsec/tyaa015
7. López-Montero, D., Álvarez-Aldana, J. L., Morales-Martínez, A., Gil-López, M., & Auñón García, J. M. (2025). Reinforcement learning for automated cybersecurity penetration testing. arXiv preprint, 2507, 1–14. https://doi.org/10.48550/arXiv.2507.02969
8. Millar, S., Podgurskii, D., Kuykendall, D., Martínez del Rincón, J., & Miller, P. (2022). Optimising vulnerability triage in DAST with deep learning. In Proceedings of the 15th ACM Workshop on Artificial Intelligence and Security (AISec) (pp. 45–56). ACM. https://doi.org/10.1145/3560830.3563726
9. Ofusori, L., Bokaba, T., & Mhlongo, S. (2024). Artificial intelligence in cybersecurity: A comprehensive review and future direction. Applied Artificial Intelligence, 38(1), 1–28. https://doi.org/10.1080/08839514.2024.2439609
10. OWASP Foundation. (2021). OWASP Top 10: The ten most critical web application security risks. Open Web Application Security Project. https://owasp.org/www-project-top-ten/
11. Ozkan-Ozay, M., Akin, E., Aslan, O., Kosunalp, S., Iliev, T., & Stoyanov, I. (2024). A comprehensive survey: Evaluating the efficiency of artificial intelligence and machine learning techniques on cyber security solutions. IEEE Access, 12, 45120–45148. https://doi.org/10.1109/ACCESS.2024.3381234
12. Qaiser, S., & Ali, R. (2018). Text mining: Use of TF-IDF to examine the relevance of words to documents. International Journal of Computer Applications, 181(1), 25–29. https://doi.org/10.5120/ijca2018917395
13. Rastogi, S., Shrotriya, A., Singh, M. K., & Potukuchi, R. V. (2022). An analysis of intrusion detection classification using supervised machine learning algorithms on NSL-KDD dataset. Journal of Computing Research and Innovation, 7(1), 112–124. https://doi.org/10.24191/jcrinn.v7i1.275
14. Schröer, C., Kruse, F., & Gómez, J. M. (2021). A systematic literature review on applying CRISP-DM process model. Procedia Computer Science, 181, 526–534. https://doi.org/10.1016/j.procs.2021.01.199
15. Suleiman, M., & Issac, B. (2018). Performance comparison of intrusion detection machine learning classifiers on benchmark and new datasets. In Proceedings of the 28th International Conference on Computer Theory and Applications (ICCTA) (pp. 18–23). IEEE. https://doi.org/10.1109/ICCTA45979.2018.9418912
16. Xu, S. (2018). Bayesian Naïve Bayes classifiers to text classification. Journal of Information Science, 44(1), 48–59. https://doi.org/10.1177/0165551516677946

DOI: https://doi.org/10.69693/ijmst.v5i1.14478
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)