JURNAL KOMPUTER TEKNOLOGI INFORMASI SISTEM KOMPUTER

p-ISSN: 2962-3022 | e-ISSN: 2963-7104
Volume 5 No 1 Juni 2026
Hal: 320-326
http://ejurnal.lkpkaryaprima.id/index.php/juktisi
DOI: https://doi.org/10.62712/juktisi.v5i1.964

Analisis Kinerja Deep Learning Berbasis *Convolutional Neural Network* (CNN) untuk deteksi Dini *SQl Injection*

Khairul[1], Muhammad Azuan[2], Toni Prabowo[3*], Aradi Sebayang[4], Tengku Didi Ferdillah[5]

[1,2,3,4,5]Magister Teknologi Informasi, Universitas Pembangunan Panca Budi, Medan, Indonesia
Email: [1]khairul@dosen.pancabudi.ac.id, [2]mhd.azuan99@gmail.com, [3]toniprabowo@gmail.com,
[4]aradisebayang23@gmail.com, [5]tengkudidiferdillah23@gmail.com
(*Email Corresponding Author: toniprabowohsb@gmail.com)
Received: 8 April 2026 | Revision: 14 April 2026 | Accepted: 17 April 2026

Abstrak

Serangan *SQL Injection* (SQLi) merupakan ancaman kritis bagi keamanan layanan publik berbasis web di lingkungan pemerintahan. Metode deteksi tradisional sering kali mengalami keterbatasan dalam menangani *payload* yang disamarkan (*obfuscated*) dan memerlukan rekayasa fitur manual yang kompleks. Penelitian ini bertujuan untuk menganalisis kinerja arsitektur *Deep Learning* berbasis *1D-Convolutional Neural Network* (1D-CNN) untuk deteksi dini serangan SQLi, dengan studi kasus pada Datacenter Diskominfo Kota Binjai. Metodologi penelitian mencakup pemrosesan dataset sebanyak 19.078 baris log akses server web riil yang dibagi menggunakan metode *Hold-out Validation* dengan proporsi 80% data latih dan 20% data uji. Arsitektur 1D-CNN dirancang untuk melakukan pemrosesan teks sekuensial guna mengekstrak fitur leksikal lokal secara otomatis langsung dari log mentah. Hasil evaluasi menunjukkan performa klasifikasi yang sangat superior dengan tingkat Akurasi 100%, Presisi 99%, dan *Recall* 97% pada identifikasi serangan. Penelitian ini menyimpulkan bahwa model 1D-CNN sangat handal dan efisien untuk diimplementasikan sebagai sistem peringatan dini (*early warning system*) tanpa mengganggu kinerja operasional layanan publik di lingkungan Pemerintah Kota Binjai.

Kata Kunci: *Deep Learning*, 1D-CNN, *SQL Injection*, Keamanan Siber, Datacenter Diskominfo Binjai.

Abstract

SQL Injection (SQLi) attacks pose a critical threat to the security of web-based public services within government environments. Traditional detection methods often face limitations in handling obfuscated payloads and require complex manual feature engineering. This study aims to analyze the performance of a Deep Learning architecture based on a 1D-Convolutional Neural Network (1D-CNN) for early detection of SQLi attacks, using a case study at the Diskominfo Datacenter of Binjai City. The research methodology involves processing a dataset consisting of 19,078 real-world web server access log entries, which are divided using the Hold-out Validation method with a proportion of 80% training data and 20% testing data. The 1D-CNN architecture is designed to process sequential text data in order to automatically extract local lexical features directly from raw logs. The evaluation results demonstrate highly superior classification performance, achieving an Accuracy of 100%, Precision of 99%, and Recall of 97% in attack identification. This study concludes that the 1D-CNN model is highly reliable and efficient for implementation as an early warning system, without disrupting the operational performance of public services within the Binjai City Government environment.

Keywords*: Deep Learning, 1D-CNN, SQL Injection, Cyber Security, Diskominfo Binjai Datacenter.*

1. PENDAHULUAN

Implementasi Sistem Pemerintahan Berbasis Elektronik (SPBE) sesuai Peraturan Presiden No. 95 Tahun 2018 telah mendorong transformasi digital yang masif di lingkungan Pemerintah Kota Binjai. Dinas Komunikasi dan Informatika (Diskominfo) Kota Binjai memegang peranan sentral dalam mengelola infrastruktur jaringan dan berbagai aplikasi pelayanan publik strategis, seperti JDIH, layanan perizinan, hingga aplikasi e-Masyarakat. Sebagai pusat data (datacenter) daerah, Diskominfo menyimpan aset informasi vital yang mencakup data kependudukan, kepegawaian, hingga perencanaan anggaran, yang menjadikannya target potensial bagi berbagai ancaman siber. Krusialnya pengamanan aset ini sejalan dengan tantangan implementasi SPBE di Indonesia secara nasional, di mana kelemahan infrastruktur teknis dan kerentanan keamanan siber telah memicu insiden berskala besar seperti gangguan pada Pusat Data Nasional (PDN) [1]. Peningkatan jumlah aplikasi berbasis web secara linier berbanding lurus dengan kompleksitas ancaman keamanan. Salah satu serangan yang paling persisten dan berbahaya adalah *SQL Injection* (SQLi), yang mengeksploitasi kerentanan pada lapisan database untuk mencuri, memanipulasi, atau menghapus data sensitif. Berbagai studi menunjukkan bahwa SQL Injection tetap menjadi ancaman utama dalam keamanan aplikasi web karena tekniknya terus berevolusi melalui metode *obfuscation* (penyamaran kode) yang sulit dideteksi oleh mekanisme pertahanan standar [2].

Mekanisme keamanan yang saat ini diterapkan di Datacenter Diskominfo umumnya masih mengandalkan pendekatan konvensional, seperti *Firewall* berbasis aturan (*rule-based*) dan *Intrusion Detection System* (IDS) berbasis *signature*. Namun, metode ini memiliki keterbatasan signifikan dalam menghadapi serangan baru atau varian serangan yang belum terdaftar dalam *database signature*. Selain itu, sistem tradisional cenderung menghasilkan tingkat *false positive* yang tinggi dan kurang adaptif terhadap pola serangan dinamis, sehingga diperlukan pendekatan yang lebih cerdas dan proaktif dalam mendeteksi ancaman [3].

Sebagai solusi atas keterbatasan tersebut, pemanfaatan *Deep Learning*, khususnya arsitektur *Convolutional Neural Network* (CNN), Arsitektur yang mengandalkan algoritma berbasis *decision tree* (seperti Random Forest) dan CNN lebih

JURNAL KOMPUTER TEKNOLOGI INFORMASI SISTEM KOMPUTER
p-ISSN: 2962-3022 | e-ISSN: 2963-7104
Volume 5 No 1 Juni 2026
Hal: 320-326
http://ejurnal.lkpkaryaprima.id/index.php/juktisi
DOI: https://doi.org/10.62712/juktisi.v5i1.964

hemat waktu pemrosesan (hanya 20 milidetik), dibandingkan dengan model LSTM yang terbukti sangat lambat dengan waktu deteksi mencapai rata-rata 2,4 detik per siklus [4]. mulai berkembang pesat dalam domain keamanan siber. CNN memiliki keunggulan dalam ekstraksi fitur secara otomatis dari data mentah tanpa memerlukan rekayasa fitur manual yang rumit. Penelitian sebelumnya membuktikan bahwa CNN mampu mencapai tingkat akurasi deteksi SQL Injection hingga di atas 97% karena kemampuannya dalam mengenali pola spasial dan tekstual pada *payload* serangan dengan latensi yang rendah [5].

Sejumlah literatur telah mengkaji efektivitas *Deep Learning* dalam mendeteksi serangan siber. Penggunaan model berbasis CNN dilaporkan mampu mengungguli metode pembelajaran mesin tradisional, baik dalam hal akurasi maupun kemampuan menekan laju *false positive* [6]. Lebih lanjut, integrasi teknik optimasi pada model *Deep Learning* terbukti dapat meningkatkan akurasi deteksi hingga 98,35% sekaligus mempercepat proses pengolahan data berbasis teks [7]. Secara umum, pendekatan berbasis kecerdasan buatan menunjukkan performa yang lebih tangguh dalam mengidentifikasi pola serangan baru (*zero-day attacks*) dibandingkan metode berbasis aturan statis..

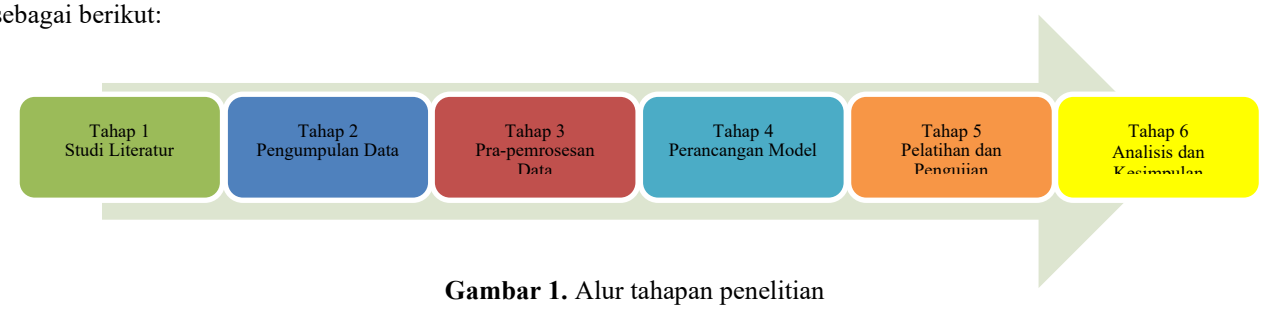
Algoritma *Long Short-Term Memory* (LSTM) memiliki kemampuan yang sangat baik dalam memilah data berdimensi tinggi pada log akses web, dengan akurasi pengenalan mencapai 98% [8]. Meskipun demikian, terdapat tantangan dalam implementasi nyata, di mana sebagian besar penelitian saat ini berfokus pada model *hybrid* kompleks seperti CNN-LSTM. Meskipun akurat, model tersebut membutuhkan sumber daya komputasi yang besar, yang seringkali kurang efisien jika diterapkan pada infrastruktur pemerintahan daerah yang mengutamakan efisiensi operasional. Terdapat kebutuhan mendesak untuk meneliti penggunaan model CNN tunggal (*single* CNN) yang dioptimalkan agar tetap memberikan akurasi tinggi namun dengan beban komputasi yang lebih ringan dan proses deteksi yang lebih cepat.

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk merancang dan mengimplementasikan arsitektur *1D-Convolutional Neural Network* (1D-CNN) untuk deteksi dini serangan *SQL Injection* pada log server Diskominfo Kota Binjai. Fokus utama penelitian ini adalah menganalisis kinerja model melalui evaluasi *Confusion Matrix* yang mencakup parameter akurasi, presisi, *recall*, dan *F1-score*, serta mengukur efisiensi waktu komputasi. Melalui penelitian ini, diharapkan dapat tercipta sistem peringatan dini (*early warning system*) yang akurat dan efisien untuk memperkuat resiliensi infrastruktur digital di lingkungan Pemerintah Kota Binjai.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian ini dilakukan melalui pendekatan eksperimental yang disusun secara sistematis untuk menjamin validitas dan akurasi hasil deteksi. Alur penelitian dimulai dari identifikasi masalah melalui studi literatur hingga penarikan kesimpulan berdasarkan hasil evaluasi kinerja model. Secara visual, tahapan tersebut dijelaskan pada Gambar 1, dengan rincian sebagai berikut:



Gambar 1. Alur tahapan penelitian

Berikan penjelasan Tahapan Alur penelitian.

1. Studi Literatur
   Melakukan kompilasi teori mengenai arsitektur *Deep Learning*, khususnya 1D-CNN, serta karakteristik serangan *SQL Injection* dan teknik *Natural Language Processing* (NLP) untuk pengolahan teks. Pendekatan *Deep Learning* dipilih karena sifatnya yang lebih adaptif dalam mengenali pola serangan yang terus bertransformasi dibandingkan metode konvensional [9].
2. Pengumpulan Data
   Mengakuisisi data sekunder berupa log akses dari server produksi Diskominfo Kota Binjai yang merepresentasikan aktivitas lalu lintas data secara riil.
3. Pra-pemrosesan Data
   Mengonversi data log tidak terstruktur menjadi format numerik menggunakan teknik NLP agar dapat diolah oleh algoritma CNN.
4. Perancangan Model
   Merancang arsitektur model 1D-CNN yang dioptimalkan untuk mengekstraksi fitur lokal pada data teks sekuensial.
5. Pelatihan dan Pengujian

JURNAL KOMPUTER TEKNOLOGI INFORMASI SISTEM KOMPUTER
p-ISSN: 2962-3022 | e-ISSN: 2963-7104
Volume 5 No 1 Juni 2026
Hal: 320-326
http://ejurnal.lkpkaryaprima.id/index.php/juktisi
DOI: https://doi.org/10.62712/juktisi.v5i1.964

Melakukan proses pembelajaran model menggunakan sebagian data dan menguji ketangguhannya menggunakan data yang belum pernah dilihat sebelumnya.

6. Analisis dan Kesimpulan
   Mengevaluasi kinerja model berdasarkan metrik standar dan merangkum hasil penelitian sebagai rekomendasi keamanan sistem.

2.2 Pengumpulan Data (*Data Collection*)

Dataset yang digunakan dalam penelitian ini merupakan data sekunder berupa *Web Server Access Log* yang diperoleh langsung dari infrastruktur server Diskominfo Kota Binjai. Total data yang dikumpulkan sebanyak 19.078 baris, yang kemudian dikategorikan secara biner ke dalam dua label.

1. Data Normal (Label 0): Merupakan permintaan HTTP sah yang tidak mengandung instruksi SQL berbahaya.
2. Data Serangan (Label 1): Merupakan permintaan yang teridentifikasi mengandung pola *SQL Injection*.

Penggunaan data log riil menjadi krusial dalam penelitian ini karena mampu merepresentasikan anomali dan karakteristik serangan yang terjadi pada lingkungan operasional yang sebenarnya. Studi terdahulu mengonfirmasi bahwa penggunaan dataset berbasis log atau *honeypot* memberikan hasil deteksi yang lebih realistis dan valid dibandingkan penggunaan dataset sintetis [10].

2.3 Pra-pemrosesan Data (*Data Preprocessing*)

Mengingat data log merupakan teks tidak terstruktur, transformasi data dilakukan melalui beberapa tahapan NLP untuk memastikan kualitas input model. Tahapan tersebut meliputi:

1. *Parsing* dan *Cleaning*
   Melakukan ekstraksi pada bagian URI *Request* dan menerapkan *URL decoding* untuk mengembalikan karakter khusus ke format asli.
2. *Case Folding*
   Menyeragamkan seluruh karakter teks menjadi huruf kecil (*lowercase*) untuk mengurangi variansi data.
3. *Tokenization*
   Memecah rangkaian teks menjadi unit-unit token yang lebih kecil. Teknik tokenisasi (pemotongan kata) yang difokuskan hanya pada kata kunci dan simbol SQL efektif membuang informasi yang tidak penting (*noise*) dari kueri, sehingga model komputasi menjadi jauh lebih ringan dan cepat [11].
4. *Padding*
   Menyamakan panjang seluruh urutan (*sequence*) data agar memiliki dimensi input yang konsisten bagi layer CNN.
5. Label *Encoding*
   Mengubah label kategori menjadi representasi numerik biner.

Kualitas pra-pemrosesan data sangat menentukan performa akhir model, karena representasi teks yang bersih mempermudah model CNN dalam mengenali pola fitur serangan secara akurat [11].

2.4 Perancangan Arsitektur Model 1D-CNN

Model yang diusulkan adalah *1D-Convolutional Neural Network* (1D-CNN) yang dirancang khusus untuk menangani data sekuensial berbasis teks. Detail arsitektur model ditunjukkan pada Tabel 1:

Tabel 1. Konfigurasi Arsitektur Model 1D-CNN

Nama Lapisan	Spesifikasi / Konfigurasi	Fungsi Utama
Embedding Layer	*Input Dimension* sesuai jumlah token	Merepresentasikan token dalam bentuk vektor padat.
Convolutional 1D	128 *filters*, *kernel size* 5, aktivasi ReLU	Mengekstraksi pola fitur lokal dari *payload* log.
Global Max Pooling	*Pooling size* standar	Merangkum fitur paling dominan dari hasil konvolusi.
Dropout	*Rate* 0.5	Mencegah terjadinya *overfitting* selama pelatihan.
Dense (Output)	1 *unit*, aktivasi Sigmoid	Melakukan klasifikasi akhir (Normal vs Serangan).

Pemilihan 1D-CNN didasarkan pada efisiensi komputasinya yang lebih tinggi dibandingkan model *hybrid* kompleks seperti CNN-LSTM, namun tetap memiliki performa tinggi dalam pengenalan pola spasial pada teks [9]. Selain itu, arsitektur ini mendukung kebutuhan deteksi secara *real-time* karena waktu inferensi yang relatif singkat [12].

Gambar 1. Arsitektur Model 1D-CNN

JURNAL KOMPUTER TEKNOLOGI INFORMASI SISTEM KOMPUTER
p-ISSN: 2962-3022 | e-ISSN: 2963-7104
Volume 5 No 1 Juni 2026
Hal: 320-326
http://ejurnal.lkpkaryaprima.id/index.php/juktisi
DOI: https://doi.org/10.62712/juktisi.v5i1.964

2.5 Lingkungan Implementasi

Eksperimen dijalankan pada lingkungan komputasi terkontrol untuk memastikan konsistensi hasil, dengan rincian:

1. Perangkat Keras: Server dengan prosesor Intel Xeon dan RAM 32 GB
2. Perangkat Lunak: Bahasa pemrograman Python 3.10, dengan pustaka utama TensorFlow/Keras untuk pengembangan model, serta Pandas, NumPy, dan Scikit-learn untuk manipulasi data dan evaluasi.

Lingkungan ini dipilih karena mendukung implementasi Deep Learning secara efisien pada sistem skala menengah.

2.6 Skenario Pengujian dan Parameter Evaluasi

Dataset dibagi menggunakan metode *Hold-out Validation* dengan proporsi 80% untuk data latih (15.262 sampel) dan 20% untuk data uji (3.816 sampel). Proses pelatihan menggunakan parameter *Epoch* sebanyak 10 dan *Batch size* sebesar 32. Evaluasi performa model dilakukan menggunakan *Confusion Matrix* untuk menghasilkan metrik sebagai berikut:

1. Akurasi: Mengukur ketepatan prediksi model secara keseluruhan.
2. Presisi: Mengukur keakuratan model dalam mengklasifikasikan serangan.
3. Recall: Mengukur kemampuan model dalam menemukan seluruh sampel serangan.
4. F1-Score: Keseimbangan rata-rata antara presisi dan *recall*.

Selain metrik tersebut, penelitian ini juga mengukur *inference time* (waktu komputasi) untuk memvalidasi kelayakan model saat diimplementasikan pada sistem produksi yang membutuhkan respon cepat [13].

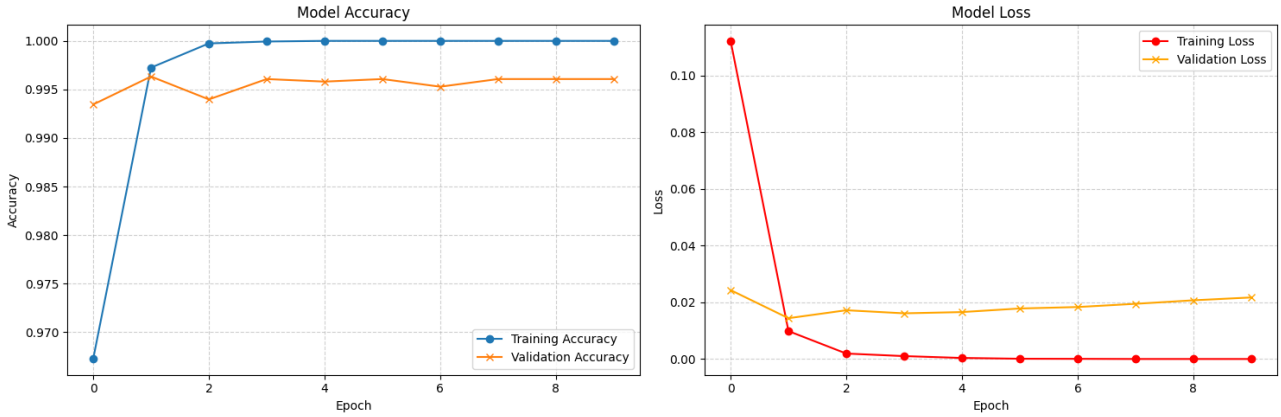
3. HASIL DAN PEMBAHASAN

3.1 Analisis Dataset dan Lingkungan Eksperimen

Eksperimen ini menggunakan dataset riil yang bersumber dari *Web Server Access Log* Datacenter Diskominfo Kota Binjai dengan total 19.078 baris data. Dataset tersebut dibagi secara sistematis menggunakan metode *Hold-out Validation* dengan rasio 80:20, sehingga menghasilkan 15.262 sampel untuk fase pelatihan dan 3.816 sampel untuk fase pengujian. Seluruh proses komputasi dijalankan pada lingkungan server berbasis prosesor Intel Xeon dengan RAM 32 GB menggunakan *framework* TensorFlow/Keras untuk menjamin stabilitas performa selama pelatihan model.

3.2 Visualisasi Proses Pelatihan

Proses pembelajaran model 1D-CNN dipantau melalui grafik akurasi dan *loss* untuk memastikan model berkembang secara optimal pada setiap *epoch*. Grafik ini memberikan gambaran visual mengenai stabilitas model dalam mengenali pola serangan *SQL Injection* dari data latih dan validasi.



Gambar 2. Grafik Model Accuracy dan Model Loss

Berdasarkan Gambar 2, terlihat bahwa kurva akurasi meningkat secara signifikan dan mencapai titik stabil pada *epoch* akhir, sejalan dengan nilai akurasi pengujian sebesar 99,61%. Sementara itu, kurva *loss* menunjukkan penurunan yang konsisten tanpa adanya indikasi *overfitting* yang drastis. Hal ini membuktikan bahwa konfigurasi arsitektur 1D-CNN yang dirancang sangat adaptif dalam mengekstraksi fitur leksikal dari log *server* Diskominfo Binjai secara efisien

3.2 Evaluasi Confusion Matrix

Kemampuan klasifikasi model 1D-CNN diuji secara mendalam melalui instrumen *Confusion Matrix* guna memetakan akurasi prediksi terhadap kondisi aktual data uji. Hasil pemetaan data menunjukkan performa yang sangat presisi sebagaimana dirinci pada Tabel 2.

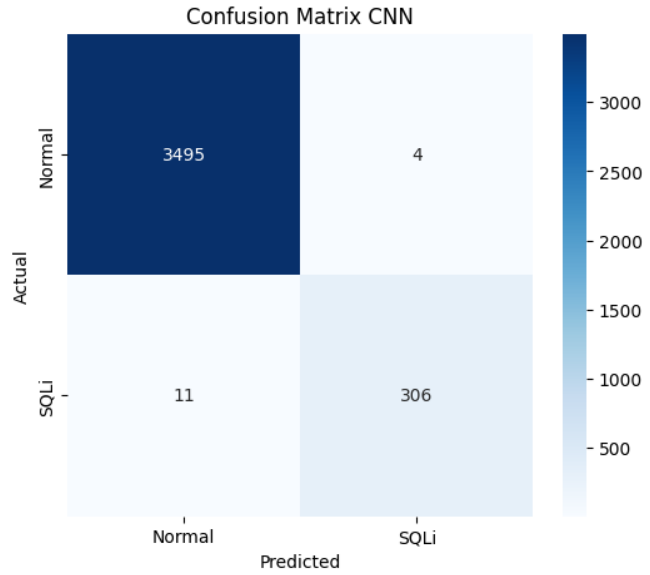
Tabel 2. Rincian Nilai Confusion Matrix Model 1D-CNN

Kategori	Deskripsi	Jumlah (Sampel)
True Negative (TN)	Trafik normal diprediksi benar sebagai normal	**3.495**

JURNAL KOMPUTER TEKNOLOGI INFORMASI SISTEM KOMPUTER
p-ISSN: 2962-3022 | e-ISSN: 2963-7104
Volume 5 No 1 Juni 2026
Hal: 320-326
http://ejurnal.lkpkaryaprima.id/index.php/juktisi
DOI: https://doi.org/10.62712/juktisi.v5i1.964

True Positive (TP)	Serangan SQLi diprediksi benar sebagai serangan	**306**
False Positive (FP)	Trafik normal salah diprediksi sebagai serangan (*False Alarm*)	**4**
False Negative (FN)	Serangan nyata salah diprediksi sebagai normal (Kebobolan)	**11**

Berdasarkan data tersebut, model menunjukkan kemampuan yang luar biasa dalam membedakan antara aktivitas legal dan anomali berbahaya. Rendahnya angka *False Positive* (4 kasus) menjadi indikator penting bahwa sistem ini memiliki tingkat spesifisitas yang sangat tinggi, mencapai 99,91%, sehingga meminimalkan gangguan operasional akibat alarm palsu (*false alarm*). Model berbasis CNN sangat tangguh digunakan pada lingkungan aplikasi real-time karena mampu membedakan anomali dengan presisi tinggi dan mencapai angka false positive (alarm palsu) hingga nol [14].



Gambar 3. *Visualisasi Confusion Matrix Model 1D-CNN*

3.3 Metrik Performa Klasifikasi

Untuk memberikan penilaian kuantitatif yang objektif, dilakukan perhitungan metrik performa berdasarkan laporan klasifikasi yang mencakup akurasi, presisi, *recall*, dan *F1-score*.

3.3.1 Analisis Akurasi dan Presisi

Secara keseluruhan, model 1D-CNN mencapai tingkat akurasi sebesar 99,61% (dibulatkan menjadi 1.00 pada *weighted average*). Perhitungan presisi pada kelas serangan menghasilkan nilai 99%, yang mengindikasikan bahwa hampir seluruh peringatan yang dikeluarkan oleh sistem merupakan serangan *SQL Injection* yang valid..

$$\text{Accuracy}=\frac{TP+TN}{TP+TN+FP+FN}=\frac{306+3.495}{3816}=\frac{3.801}{3.816}=0{,}9961\approx\mathbf{99{,}61\%}$$

$$\text{Precision}=\frac{TP}{TP+FP}=\frac{306}{306+4}=\frac{306}{310}=0{,}9871\approx\mathbf{98{,}71\%}$$

3.3.2 Analisis Recall dan F1-Score

Metrik *recall* pada kelas serangan mencapai 97%, menunjukkan sensitivitas model yang sangat baik dalam mengidentifikasi pola serangan meskipun terdapat teknik penyamaran (*obfuscation*). Keseimbangan antara presisi dan *recall* yang direpresentasikan melalui *F1-score* sebesar 0,98 membuktikan bahwa arsitektur 1D-CNN sangat stabil dan handal untuk tugas klasifikasi teks sekuensial yang kompleks.

$$\text{Recall}=\frac{TP}{TP+FN}=\frac{306}{306+11}=\frac{306}{317}=0{,}9653\approx\mathbf{96{,}53\%}$$

3.4 Karakteristik Ekstraksi Fitur Leksikal

Pencapaian performa yang superior ini mengonfirmasi efektivitas filter konvolusi pada arsitektur 1D-CNN dalam mengekstraksi fitur spasial lokal (*n-grams*) secara otomatis langsung dari log mentah (*raw logs*). Kemampuan *sliding window* pada model berhasil menangkap hubungan kontekstual antar karakter dalam permintaan HTTP, seperti penggunaan kata kunci UNION, SELECT, atau karakter komentar -- yang sering kali disisipkan oleh penyerang untuk mengeksploitasi basis data.

JURNAL KOMPUTER TEKNOLOGI INFORMASI SISTEM KOMPUTER
p-ISSN: 2962-3022 | e-ISSN: 2963-7104
Volume 5 No 1 Juni 2026
Hal: 320-326
http://ejurnal.lkpkaryaprima.id/index.php/juktisi
DOI: https://doi.org/10.62712/juktisi.v5i1.964

3.5 Analisis Kesalahan (*False Negative*)

Meskipun mencapai performa tinggi, terdapat 11 kasus *False Negative* yang menjadi bahan evaluasi. Kegagalan deteksi ini diidentifikasi berasal dari beberapa faktor teknis:

1. *Obfuscation* Tingkat Tinggi: Penggunaan teknik pengodean ganda (double encoding) yang sangat kompleks oleh penyerang.
2. *Zero-day Payload*: Adanya variasi sintaks serangan baru yang belum terwakili secara signifikan dalam data pelatihan.
3. *Payload Pendek*: Serangan yang memiliki karakter sangat singkat sehingga menyerupai parameter input teks legal.

Munculnya 11 kasus kebobolan (False Negative) pada model ini adalah hal yang wajar dan sejalan dengan temuan riset global terbaru, yang menegaskan bahwa meskipun 1D-CNN sangat tangguh dalam menekan False Negative hingga rasio terkecil, mutasi serangan siber selalu menyisakan celah marjinal [15].

3.6 Signifikansi dan Kelayakan Implementasi

Hasil penelitian ini memberikan implikasi praktis yang besar bagi Diskominfo Kota Binjai. Dengan tingkat alarm palsu yang minimal, tim IT dapat fokus pada mitigasi ancaman nyata tanpa terganggu oleh kebisingan data. Efisiensi waktu komputasi yang rendah pada model *Single* 1D-CNN memastikan bahwa sistem deteksi ini layak diintegrasikan ke dalam infrastruktur produksi tanpa membebani sumber daya server.

4. KESIMPULAN

Berdasarkan hasil analisis dan eksperimen yang telah dilakukan, dapat disimpulkan bahwa penerapan arsitektur *1D-Convolutional Neural Network* (1D-CNN) terbukti sangat efektif dalam melakukan deteksi dini serangan *SQL Injection* pada infrastruktur Datacenter Diskominfo Kota Binjai. Model ini berhasil mengatasi keterbatasan sistem keamanan konvensional yang bersifat statis melalui kemampuan ekstraksi fitur leksikal secara otomatis dari data log mentah tanpa memerlukan rekayasa fitur manual yang kompleks. Melalui pengujian terhadap 19.078 baris data log akses riil, model menunjukkan performa impresif dengan tingkat akurasi mencapai 99,61% (pembulatan 1.00 pada laporan klasifikasi), presisi 99%, dan *recall* 97% pada identifikasi serangan. Hasil ini menegaskan bahwa sistem sangat handal dalam memitigasi mayoritas ancaman *SQL Injection* sebelum mengeksploitasi basis data sensitif pemerintah, seperti data kependudukan dan perencanaan anggaran. Efisiensi waktu komputasi yang dihasilkan oleh model *Single* 1D-CNN terbukti lebih ringan dibandingkan model *hybrid*, sehingga sangat layak untuk diimplementasikan pada server produksi guna memperkuat resiliensi Sistem Pemerintahan Berbasis Elektronik (SPBE). Sebagai pengembangan selanjutnya, disarankan agar cakupan deteksi diperluas ke jenis serangan siber lainnya serta dilakukan integrasi model ke dalam *dashboard* pemantauan keamanan secara *real-time*.

REFERENCES

[1] R. Israyudin, F. M. Arrofi, and A. R. Dwiardi, “Digital Transformation through Electronic-Based Government System Policy in Indonesia: A Policy Narrative Analysis,” *J. La Soc.*, vol. 6, no. 2, pp. 281–292, 2025, doi: 10.37899/journal-la-sociale.v6i2.1825.

[2] H. P. Fitrian, M. N. Khaerudin, M. R. Umarulloh, R. Ahmad, and A. R. Agustin, “Systematic Literature Review : Peran Artificial Intelligence dalam Meningkatkan Akurasi Deteksi SQL Injection pada Aplikasi Web,” no. 1, pp. 1–10, 2026.

[3] S. S. Ahmed and M. L. Al Dabag, “Machine Learning and Deep Learning Approaches for Accent Recognition: A Review,” *IEEE Access*, vol. 13, pp. 51527–51550, 2025, doi: 10.1109/ACCESS.2025.3552935.

[4] S.R. Menaka, “An Efficient SQL Injection Detection with a Hybrid CNN & Random Forest Approach,” *J. Inf. Syst. Eng. Manag.*, vol. 10, no. 18s, pp. 664–673, 2025, doi: 10.52783/jisem.v10i18s.2979.

[5] M. Shahbaz, G. Mumtaz, S. Zubair, and M. Rehman, “Evaluating CNN Effectiveness in SQL Injection Attack Detection,” *J. Comput. \& …*, vol. 07, no. 02, 2024.

[6] H. Sun, Y. Du, and Q. Li, “Deep Learning-Based Detection Technology for SQL Injection Research and Implementation,” *Appl. Sci.*, vol. 13, no. 16, 2023, doi: 10.3390/app13169466.

[7] Y. Chen, G. Liang, and Q. Wang, “Research on SQL Injection Detection Technology Based on Content Matching and Deep Learning,” *Comput. Mater. Contin.*, vol. 84, no. 1, pp. 1145–1167, 2025, doi: 10.32604/cmc.2025.063319.

[8] F. D. Hafriadi and R. Ardiansyah, “NETWORK’S ACCESS LOG CLASSIFICATION FOR DETECTING SQL INJECTION ATTACKS WITH THE LSTM ALGORITHM,” *J. Tek. Inform.*, vol. 5, no. 4 SE-Articles, pp. 745–752, Sep. 2024, doi: 10.52436/1.jutif.2024.5.4.2157.

JURNAL KOMPUTER TEKNOLOGI INFORMASI SISTEM KOMPUTER
p-ISSN: 2962-3022 | e-ISSN: 2963-7104
Volume 5 No 1 Juni 2026
Hal: 320-326
http://ejurnal.lkpkaryaprima.id/index.php/juktisi
DOI: https://doi.org/10.62712/juktisi.v5i1.964

[9] F. Alghamdi and B. Ben Ammar, "Enhancing SQL Code Security and Maintainability: A Deep Learning Based Approach," *Int. J. Adv. Artif. Intell. Mach. Learn.*, vol. 2, no. 3, pp. 160–169, 2025, doi: 10.58723/ijaaiml.v2i3.515.
[10] H. Yu *et al.*, "Multi-Agent Honeypot-Based Request-Response Context Dataset for Improved SQL Injection Detection Performance," pp. 2–6, 2026.
[11] R. T. Lo, W. J. Hwang, and T. M. Tai, "SQL Injection Detection Based on Lightweight Multi-Head Self-Attention," *Appl. Sci.*, vol. 15, no. 2, pp. 1–17, 2025, doi: 10.3390/app15020571.
[12] B. Turarov and S. Kabdrakhova, "OPTIMIZING CONVOLUTIONAL NEURAL NETWORKS FOR REAL-TIME SQL INJECTION DETECTION IN POSTGRESQL DATABASES WITH A FOCUS ON REDUCING FALSE POSITIVES," vol. 6, no. 135, 2025.
[13] N. Rachana, N. Kshama, S. S. Venki, R. Prajwal, and N. Thanuja, "Injectiq : ML-Powered SQL Injection Detection For Saas Applications," vol. 13, no. 12, 2025.
[14] D. S. W. Nguyen, D. F. Alrubie, D. S. W. Nguyen, and D. F. Alrubie, "Designing a Detection Model for SQL Injection Attack," *J. Comput. Commun.*, vol. 13, no. 08, pp. 40–79, Aug. 2025, doi: 10.4236/jcc.2025.138003.
[15] B. M. Hassn, E. S. Alomari, J. S. Alrubaye, and O. A. Hassen, "Adversarially Robust 1D-CNN for Malicious Traffic Detection in Network Security Applications," *J. Cybersecurity Inf. Manag.*, vol. 16, no. 1, pp. 162–175, 2025, doi: 10.54216/JCIM.160113.