Blend Sains
Jurnal Teknik
https://jurnal.ilmubersama.com/index.php/blendsains

Artikel Penelitian (Teknik Informatika)

Analisis Komparatif Pengaruh Kedalaman Arsitektur Convolutional Neural Network untuk Deteksi Anomali Jaringan

Krisna Nuresa Qodri [1], Satriawan Desmana [1], Bella Adinda Putri [1], Ratih [1], Muhammad Abdul Mu'in [2]*

[1] *Rekayasa Keamanan Siber, Jurusan Komputer dan Bisnis, Politeknik Negeri Cilacap, Cilacap, Indonesia*
[2] *Teknik Informatika, Jurusan Komputer dan Bisnis, Politeknik Negeri Cilacap, Cilacap, Indonesia*

INFORMASI ARTIKEL

Diterima Redaksi: 23 Juni 2026
Revisi Akhir: 22 Juli 2026
Diterbitkan *Online*: 30 Juli 2026

KATA KUNCI

Convolutional Neural Network
Intrusion Detection System
Deteksi Anomali Jaringan
Deep Learning
Keamanan Siber

KORESPONDENSI (*)

E-mail: krisnanuresa@pnc.ac.id

A B S T R A K

Tingkat kecanggihan serangan siber yang terus meningkat menjadi tantangan serius bagi sistem keamanan jaringan konvensional. Penelitian ini mengkaji efektivitas Convolutional Neural Network (CNN) dalam mendeteksi anomali jaringan menggunakan dataset UNSW-NB15. Dua arsitektur CNN dengan tingkat kompleksitas berbeda dibandingkan, yaitu CNN dasar dengan satu blok konvolusi dan CNN menengah dengan tiga blok konvolusi. Hasil eksperimen menunjukkan bahwa arsitektur CNN menengah memberikan kinerja yang lebih unggul dengan akurasi 97,91%, presisi 98,89%, dan nilai ROC-AUC 0,9979, jauh melampaui model dasar yang hanya mencapai akurasi 94,36%. Tingkat false positive yang rendah, yaitu 1,93%, serta tingkat false negative sebesar 2,18%, mengindikasikan kelayakan praktis yang kuat untuk diterapkan pada lingkungan operasional nyata. Temuan ini memberikan bukti empiris bahwa rekayasa kedalaman arsitektur yang moderat disertai regularisasi kuat memegang peranan krusial dalam menjaga stabilitas konvergensi dan mengeliminasi risiko overfitting ekstrem pada sistem deteksi intrusi jaringan.

PENDAHULUAN

Ancaman keamanan siber telah berkembang secara pesat selama satu dekade terakhir. Para penyerang kini memanfaatkan teknik yang jauh lebih canggih sehingga metode deteksi berbasis signature konvensional menjadi semakin tidak memadai. Organisasi di berbagai belahan dunia menghadapi tantangan yang terus-menerus dari *advanced persistent threats*, *zero-day exploit*, serta *malware* polimorfik yang mampu mengelabui mekanisme keamanan konvensional. Laporan IBM tahun 2025 mencatat bahwa rata-rata kerugian akibat kebocoran data secara global mencapai 4,44 juta dolar AS, sebuah angka yang menegaskan betapa mendesaknya kebutuhan akan mekanisme deteksi intrusi yang lebih andal [1]. *Network Intrusion Detection System* (NIDS) merupakan komponen krusial dalam infrastruktur keamanan siber modern karena berfungsi memantau lalu lintas jaringan guna mengidentifikasi aktivitas berbahaya maupun pelanggaran kebijakan. Implementasi NIDS konvensional umumnya masih sangat bergantung pada *signature* yang telah ditentukan sebelumnya beserta pencocokan berbasis aturan, sehingga muncul keterbatasan mendasar berupa ketidakmampuan mendeteksi serangan yang belum pernah dikenali sebelumnya, kerentanan terhadap teknik obfuskasi sederhana, serta beban pemeliharaan manual yang tidak berkelanjutan [2]. Berbagai kelemahan tersebut mendorong penelitian yang semakin intensif ke arah pendekatan deteksi berbasis anomali, yang bekerja dengan mengidentifikasi penyimpangan dari pola perilaku normal.

Teknik *machine learning* telah menunjukkan hasil yang menjanjikan dalam mengatasi keterbatasan tersebut melalui pengenalan pola secara otomatis berdasarkan data historis. Penelitian-penelitian awal mengeksplorasi algoritme klasik seperti *decision tree* dan *support vector machine* dengan tingkat keberhasilan yang bervariasi. Meskipun metode-metode

https://doi.org/10.56211/blendsains.v5i1.1841

tersebut mampu meningkatkan kemampuan deteksi, penerapannya tetap membutuhkan rekayasa fitur secara manual yang ekstensif, di mana pakar domain harus mengidentifikasi dan mengekstraksi karakteristik jaringan yang relevan secara manual. *Deep learning* secara fundamental mengubah paradigma ini dengan memungkinkan ekstraksi fitur otomatis langsung dari data mentah atau data yang hanya diproses secara minimal melalui representasi hierarkis [3], [4], [5]. Di antara berbagai arsitektur *deep learning*, *Convolutional Neural Network* (CNN), yang pada awalnya dikembangkan untuk pengenalan citra, menarik banyak perhatian karena kemampuannya mendeteksi pola lokal dan hierarki spasial. Arsitektur-arsitektur lain seperti AlexNet [6], ResNet [7], GoogLeNet [8], dan VGGNet [9] telah membuktikan kekuatan pembelajaran fitur konvolusi yang dalam pada skala besar. Walaupun lalu lintas jaringan tidak memiliki struktur spasial yang jelas seperti citra, para peneliti berhasil mengadaptasi CNN dengan memperlakukan urutan paket atau statistik aliran data sebagai *pseudo-image*, sehingga operasi konvolusi dapat mendeteksi pola temporal maupun korelasi antarfitur [10], [11]. Berbagai penelitian terbaru telah mengeksplorasi konfigurasi CNN yang beragam dengan hasil yang bervariasi. Vibhute dkk. mengembangkan model CNN yang dipadukan dengan seleksi fitur berbasis *random forest* untuk mereduksi 41 fitur asli UNSW-NB15 menjadi 15 fitur paling representatif, sehingga kompleksitas komputasi berkurang sembari kemampuan deteksi tetap terjaga [12]. Thaljaoui mengembangkan model hibrida CNN-LSTM dengan optimasi Bayesian dan memperoleh akurasi sekitar 98% pada dataset yang sama [13]. Wang dkk. memperoleh akurasi 97,89% menggunakan CNN-RNN hibrida pada dataset CICIDS2017 [14], [15].

Kesenjangan penting yang masih perlu dijawab adalah bagaimana kedalaman arsitektur, khususnya jumlah lapisan konvolusi, secara spesifik memengaruhi kinerja deteksi intrusi jaringan. Aliran data jaringan umumnya direpresentasikan sebagai vektor fitur yang relatif pendek dibandingkan dengan citra beresolusi tinggi, sehingga berpotensi membatasi abstraksi hierarkis yang dapat dipelajari oleh jaringan yang lebih dalam [16]. Sejauh ini belum ada kajian komprehensif yang secara sistematis mengevaluasi trade-off antara kedalaman arsitektur, kinerja deteksi, dan efisiensi komputasi untuk deteksi anomali jaringan. Penelitian ini berupaya mengisi kesenjangan tersebut melalui analisis komparatif terfokus terhadap arsitektur CNN dengan kedalaman yang berbeda pada dataset UNSW-NB15. Penelitian ini menunjukkan bahwa CNN tiga lapis yang dirancang dapat mencapai akurasi 97,91% dan ROC-AUC 0,9979, melampaui sejumlah pendekatan terbaru yang dilaporkan dalam literatur terbaru sekaligus tetap menjaga efisiensi komputasi. Analisis lebih lanjut memperlihatkan bahwa model yang ditingkatkan ini mampu menurunkan tingkat *false positive* sebesar 77,5% dan *false negative* sebesar 44,8% dibandingkan model dasar yang lebih sederhana, sehingga memberikan wawasan praktis bagi penerapannya di lapangan. Selanjutnya, penelitian ini disusun sebagai berikut. Bagian 2 menguraikan metodologi penelitian, mencakup karakteristik dataset, prosedur pra-pemrosesan, dan rancangan arsitektur CNN. Bagian 3 menyajikan hasil eksperimen beserta analisis kinerja secara komprehensif. Bagian 4 memuat pembahasan atas temuan dan implikasi praktisnya. Bagian 5 menutup makalah ini dengan ringkasan dan arah penelitian selanjutnya.

TINJAUAN PUSTAKA

Upaya menerapkan *machine learning* untuk deteksi intrusi telah berlangsung lebih dari satu dekade. Buczak dan Guven [2] dalam surveinya menunjukkan bahwa algoritme klasik seperti *decision tree* dan *support vector machine* mampu meningkatkan kemampuan deteksi dibandingkan pendekatan berbasis *signature*, meskipun masih bergantung pada rekayasa fitur manual yang ekstensif. Keterbatasan ini mendorong peralihan ke *deep learning* yang memungkinkan ekstraksi fitur otomatis. Yin dkk. [37] memanfaatkan *Recurrent Neural Network* untuk menangkap ketergantungan temporal pada lalu lintas jaringan, sementara Vinayakumar dkk. [34] mengembangkan kerangka *deep learning* terpadu yang mengevaluasi beragam arsitektur pada berbagai dataset deteksi intrusi. Beberapa survei komprehensif [33], [35], [36] menegaskan bahwa pendekatan berbasis *deep learning* secara konsisten mengungguli metode konvensional, namun juga menyoroti tantangan berupa kebutuhan data pelatihan yang besar serta ketergantungan kinerja pada relevansi dataset yang digunakan.

Pada ranah *Convolutional Neural Network*, sejumlah penelitian mengadaptasi operasi konvolusi yang semula dirancang untuk citra agar sesuai dengan karakteristik lalu lintas jaringan. Wang dkk. [10] memperlakukan lalu lintas sebagai *pseudo-image* untuk klasifikasi *malware*, dan Vinayakumar dkk. [11] membuktikan kelayakan CNN untuk deteksi intrusi. Pada dataset UNSW-NB15, Vibhute dkk. [12] memadukan CNN dengan seleksi fitur berbasis *random forest* untuk mereduksi dimensi fitur, sedangkan Thaljaoui [13] mengusulkan arsitektur hibrida CNN-LSTM dengan optimasi Bayesian dan memperoleh akurasi sekitar 98%. Pendekatan hibrida lain oleh Wang dkk. [14] menggabungkan komponen konvolusi dan rekuren pada dataset CICIDS2017 dengan akurasi 97,89%. Meskipun berbagai penelitian tersebut melaporkan kinerja yang menjanjikan, sebagian besar berfokus pada peningkatan kompleksitas melalui penggabungan

beragam jenis jaringan, dan belum ada kajian yang secara sistematis mengisolasi pengaruh kedalaman arsitektur—khususnya jumlah blok konvolusi—terhadap kinerja deteksi. Kesenjangan inilah yang menjadi fokus penelitian ini.

METODOLOGI

Deskripsi Dataset

Penelitian ini menggunakan dataset UNSW-NB15, yang merepresentasikan kemajuan signifikan dibandingkan dataset-dataset lama dalam hal relevansinya dengan kondisi jaringan kontemporer. Dataset ini dikembangkan oleh *Australian Centre for Cyber Security* di *University of New South Wales* menggunakan perangkat IXIA PerfectStorm pada *Cyber Range Lab* untuk menghasilkan campuran antara aktivitas normal modern yang nyata dan perilaku serangan kontemporer yang disimulasikan [17]. Dataset ini merekam lalu lintas jaringan selama 31 jam pada tanggal 22 dan 23 Januari 2015, dengan total sekitar 2,5 juta rekaman, di mana sebagian darinya, sebanyak 257.673 rekaman, telah dikurasi secara khusus dan umum digunakan untuk eksperimen *machine learning*. Dataset ini mencakup 49 fitur yang mencirikan aliran jaringan dari berbagai perspektif. Fitur aliran dasar meliputi alamat IP sumber dan tujuan, nomor port, jenis protokol, serta kategori layanan; fitur konten merekam karakteristik payload seperti jumlah byte tiap arah, nilai time-to-live, dan ukuran window; fitur berbasis waktu mencatat durasi aliran, waktu mulai/akhir, dan statistik selisih waktu kedatangan; sedangkan fitur turunan mencakup statistik agregat seperti rata-rata ukuran paket dan kedalaman transaksi. Kekayaan fitur ini memungkinkan model machine learning mempelajari pola kompleks yang membedakan lalu lintas normal dari serangan.

Berbeda dengan dataset lama seperti KDD99 dan NSL-KDD [18] ,UNSW-NB15 mencakup sembilan kategori serangan modern yang merefleksikan lanskap ancaman terkini: *fuzzers* (mencari celah melalui input acak), *analysis* (*port scanning*, *spamming*, pengintaian), *backdoor* (melewati autentikasi), *Denial of Service* (membuat sumber daya tidak dapat diakses), *exploits* (menyasar kerentanan yang diketahui), *generic* (menyerang *block cipher*), *reconnaissance* (pengumpulan informasi), *shellcode* (eksekusi kode berbahaya), dan *worms* (*malware* yang mereplikasi diri). Keragaman ini menjadikan dataset sebagai lahan uji yang kuat untuk mengevaluasi sistem deteksi intrusi. Untuk klasifikasi biner, kesembilan kategori serangan dikonsolidasikan menjadi satu kelas berbahaya, sehingga permasalahan dirumuskan sebagai pembedaan antara perilaku normal dan serangan. Distribusi kelasnya cukup seimbang, yaitu sekitar 67% lalu lintas normal dan 33% serangan, yang lebih mencerminkan kondisi nyata dibandingkan ketimpangan ekstrem pada dataset lama dan memungkinkan model mempelajari representasi bermakna tanpa oversampling sintetis yang ekstensif. Tabel 1 merangkum karakteristik dataset UNSW-NB15 yang digunakan.

Tabel 1. Karakteristik Dataset UNSW-NB15

No	Karakteristik	Nilai
1	Total Rekaman	257.673
2	Set Pelatihan	175.341
3	Set Pengujian	82.332
4	Total Fitur	49
5	Jenis Fitur	Campuran (numerik dan kategorikal)
6	Lalu Lintas Normal	93.000
7	Lalu Lintas Serangan	45.155
8	Kategori Serangan	9 jenis

Sifat modern dataset ini beserta cakupan fiturnya yang komprehensif menjadikannya sangat sesuai untuk mengevaluasi pendekatan deteksi intrusi kontemporer. Penelitian sebelumnya telah menunjukkan bahwa model yang dilatih menggunakan UNSW-NB15 memiliki kemampuan generalisasi yang lebih baik terhadap skenario dunia nyata dibandingkan model yang dilatih pada dataset yang sudah usang [19]. Namun demikian, peningkatan kompleksitas dan keragaman pola serangan juga menghadirkan tantangan yang lebih besar bagi algoritme klasifikasi, sehingga dataset ini menjadi tolok ukur yang ideal untuk menilai kapabilitas metode *machine learning* tingkat lanjut.

 https://doi.org/10.56211/blendsains.v5i1.1841

Pra-pemrosesan Data

Pra-pemrosesan lalu lintas jaringan yang efektif terbukti krusial bagi kinerja CNN [20], karena representasi fitur input langsung memengaruhi kemampuan model mempelajari pola diskriminatif. Tahap pembersihan awal menghapus fitur yang tidak prediktif atau berpotensi menimbulkan bias. Alamat IP sumber dan tujuan dikeluarkan karena merepresentasikan konfigurasi jaringan spesifik dan bukan pola serangan yang dapat digeneralisasi. Fitur timestamp absolut dihapus, sementara fitur temporal turunan seperti durasi aliran dan selisih waktu kedatangan tetap dipertahankan. Label kategori serangan juga dikeluarkan dari himpunan fitur. Proses ini mereduksi ruang fitur dari 49 menjadi 43 dimensi, menghilangkan noise tanpa mengurangi karakteristik yang relevan. Imputasi nilai hilang mempertimbangkan sifat statistik tiap fitur: fitur numerik seperti jumlah paket dan volume byte menggunakan imputasi median yang lebih tahan terhadap outlier, sedangkan fitur kategorikal seperti protokol, layanan, dan status koneksi menggunakan imputasi modus. Pendekatan ini lebih unggul dibandingkan membuang rekaman yang dapat menimbulkan bias seleksi. Untuk pengkodean, diterapkan label encoding yang memetakan setiap kategori ke bilangan bulat (TCP, UDP, ICMP menjadi 0, 1, 2, dan seterusnya). Meskipun one-hot encoding menghindari urutan artifisial, label encoding memadai karena konvolusi mampu menangkap relasi non-linear, sekaligus menjaga dimensi tetap ringkas. Penskalaan dilakukan melalui normalisasi z-score agar setiap fitur memiliki rata-rata nol dan varians satu, sesuai persamaan berikut:

$$z = \frac{x-\mu}{\sigma} \qquad (1)$$

dengan x menyatakan nilai fitur asli, μ menyatakan rata-rata fitur yang dihitung dari set pelatihan, σ menyatakan deviasi standar, dan z merupakan nilai hasil standardisasi. Transformasi ini mencegah fitur dengan nilai absolut besar, seperti total byte yang ditransfer, mendominasi proses pembelajaran dibandingkan fitur dengan skala yang lebih kecil seperti nilai *time-to-live*. Secara khusus, parameter normalisasi hanya dihitung dari set pelatihan dan kemudian diterapkan pada set validasi dan set pengujian untuk mencegah kebocoran informasi yang dapat menggelembungkan estimasi kinerja secara artifisial [21]. Keputusan pra-pemrosesan paling menantang adalah mengubah vektor fitur tabular menjadi format yang sesuai untuk konvolusi. Berbeda dengan citra yang memiliki struktur spasial 2D, fitur jaringan merupakan urutan pengukuran 1D tanpa relasi spasial yang jelas. Setelah eksperimen yang ekstensif, dipilih struktur 2D memanjang berdimensi (1, 43, 1) yang memperlakukan vektor 43 fitur sebagai satu baris dengan satu kanal, memungkinkan kernel konvolusi memindai antarfitur untuk mendeteksi pola lokal sembari menghindari relasi artifisial dari penyusunan ulang yang lebih kompleks. Dataset dibagi menggunakan stratified random sampling untuk menjaga konsistensi distribusi kelas: 70% untuk pelatihan (150.441 rekaman), 15% untuk validasi (26.928 rekaman), dan 15% untuk pengujian (26.928 rekaman). Prosedur stratifikasi ini menjamin proporsi kelas sekitar 67% lalu lintas normal dan 33% serangan tetap terjaga secara konsisten pada setiap partisi, sehingga evaluasi pada set validasi dan pengujian merepresentasikan distribusi kelas yang sama dengan data pelatihan. Set validasi digunakan untuk penyesuaian hiperparameter dan early stopping, sementara set pengujian tidak disentuh hingga evaluasi akhir guna memberikan estimasi kinerja yang tidak bias. Pembagian tiga arah ini mengikuti praktik terbaik dan memungkinkan perbandingan yang adil dengan penelitian lain pada dataset yang sama.

Rancangan Arsitektur CNN

Penelitian ini merancang dua arsitektur CNN yang merepresentasikan titik berbeda pada spektrum kompleksitas guna mengkaji secara sistematis bagaimana kedalaman arsitektur memengaruhi kinerja deteksi. Kedua arsitektur tersebut berbagi prinsip rancangan yang sama, meliputi penggunaan fungsi aktivasi ReLU [22] karena efektivitasnya yang telah terbukti dalam *deep learning*, *batch normalization* [23] untuk menstabilkan proses pelatihan, dan regularisasi *dropout* [24] untuk mencegah *overfitting*.

CNN Dasar (CNN-1)

Model dasar, yang diberi nama CNN-1, menerapkan struktur CNN minimal yang berfungsi sebagai titik acuan kinerja. Arsitektur ini secara sengaja dirancang sesederhana mungkin untuk mengevaluasi apakah ekstraksi fitur konvolusi yang mendasar sudah cukup untuk mendeteksi anomali jaringan. Model ini dimulai dengan satu lapisan konvolusi yang memuat 32 filter dengan ukuran kernel (1, 3). Dimensi kernel yang tidak biasa ini merefleksikan bentuk input memanjang pada penelitian ini, di mana dimensi tinggi bernilai 1 dan dimensi lebar mencakup 43 fitur. Konfigurasi ini memungkinkan

setiap filter memeriksa tiga fitur berdekatan secara bersamaan, sehingga berpotensi menangkap korelasi lokal seperti relasi antarstatistik protokol terkait atau pengukuran temporal.

Setelah lapisan konvolusi, *batch normalization* menormalkan aktivasi pada dimensi batch, sehingga mempercepat konvergensi pelatihan dan mengurangi sensitivitas terhadap inisialisasi. Daripada menggunakan *max pooling* konvensional yang berpotensi menghilangkan informasi pada cakupan spasial yang terbatas [25], penelitian ini menerapkan *global average pooling* yang menghitung rata-rata aktivasi untuk setiap filter pada keseluruhan dimensi fitur. Operasi ini secara bersamaan mereduksi representasi menjadi vektor berukuran tetap yang sesuai untuk lapisan *dense* sekaligus memberikan bentuk regularisasi dengan meringkas respons setiap filter ke dalam satu nilai tunggal. Arsitektur ini selanjutnya beralih ke lapisan *fully connected* untuk keputusan klasifikasi akhir. Dua lapisan *dense* dengan masing-masing 64 dan 32 neuron memproses fitur hasil *pooling*, dengan aktivasi ReLU yang menghadirkan non-linearitas yang diperlukan. Regularisasi *dropout* dengan rasio 0,3 dan 0,2 secara acak menonaktifkan neuron selama pelatihan, sehingga jaringan dipaksa mengembangkan representasi yang tangguh dan tidak bergantung pada kombinasi neuron tertentu. Lapisan output akhir terdiri atas satu neuron dengan aktivasi *sigmoid* yang menghasilkan skor probabilitas antara 0 dan 1 yang merepresentasikan kemungkinan lalu lintas berbahaya.

CNN Menengah (CNN-2)

Model menengah, yang diberi nama CNN-2, menghadirkan kedalaman tambahan melalui tiga blok konvolusi, sehingga memungkinkan pembelajaran fitur secara hierarkis pada berbagai tingkat abstraksi [26]. Arsitektur ini merepresentasikan hipotesis bahwa kedalaman yang moderat memungkinkan jaringan membangun representasi yang semakin kompleks, di mana lapisan awal mendeteksi pola dasar dan lapisan yang lebih dalam mengombinasikan pola tersebut menjadi signature serangan yang kompleks.

Blok konvolusi pertama memuat 32 filter dengan kernel (1, 3), diikuti batch normalization dan max pooling stride (1, 2) yang memadatkan dimensi fitur menjadi separuh sembari mempertahankan pola paling menonjol. Blok kedua meningkatkan jumlah filter menjadi 64 dan blok ketiga menjadi 128, mengikuti pola rancangan umum CNN di mana lapisan lebih dalam memperoleh kapasitas representasional lebih besar untuk mengkodekan konsep yang lebih abstrak; pooling tidak diterapkan setelah blok terakhir untuk menjaga informasi sebelum agregasi global. Seluruh blok menggunakan kernel (1, 3) yang sama, dan batch normalization setelah setiap konvolusi menjaga aktivasi tetap stabil meskipun kedalaman bertambah. Setelah ketiga blok, global average pooling mengagregasikan 128 peta fitur menjadi vektor berdimensi 128, yang kemudian melewati dua lapisan fully connected dengan 128 dan 64 neuron—lebih besar dibandingkan CNN-1 untuk mengakomodasi fitur yang lebih kaya. Rasio dropout 0,4 dan 0,3 memberikan regularisasi lebih kuat guna mencegah overfitting meskipun parameter lebih banyak, sementara lapisan output sigmoid mempertahankan struktur klasifikasi biner yang sama seperti CNN-1. Tabel 2 menyajikan perbandingan rinci spesifikasi arsitektur kedua model.

Tabel 2. Spesifikasi Arsitektur CNN

No	Komponen	CNN-1 (Dasar)	CNN-2 (Menengah)
1	Bentuk Input	(1, 43, 1)	(1, 43, 1)
2	Blok Konvolusi	1	3
3	Filter per Blok	32	32, 64, 128
4	Ukuran Kernel	(1, 3)	(1, 3) untuk seluruh blok
5	Jenis Pooling	Global Average	Max (1,2) + Global Average
6	Batch Normalization	Setelah konvolusi	Setelah setiap konvolusi
7	Lapisan Fully Connected	2 (64, 32)	2 (128, 64)
8	Rasio Dropout	0,3; 0,2	0,4; 0,3
9	Aktivasi Output	Sigmoid	Sigmoid
10	Total Parameter	~45.000	~128.000
11	Regularisasi L2	0,001	0,001

 https://doi.org/10.56211/blendsains.v5i1.1841

Rasionalisasi rancangan di balik kedua arsitektur ini menyeimbangkan beberapa pertimbangan yang saling bersaing. Strategi pengubahan bentuk memanjang (1, 43, 1) mengakui bahwa fitur jaringan tidak memiliki struktur spasial 2D seperti citra, namun tetap memungkinkan pemrosesan konvolusi. Ukuran kernel (1, 3) merepresentasikan lebar minimum yang diperlukan untuk mendeteksi pola lokal yang bermakna tanpa jumlah parameter yang berlebihan. Perluasan filter secara progresif pada CNN-2 memberikan kapasitas representasional yang semakin besar pada lapisan yang lebih dalam, tempat pola yang lebih kompleks muncul. Kombinasi *batch normalization* dan *dropout* menangani aspek regularisasi yang berbeda, dengan *batch normalization* mengendalikan *internal covariate shift* dan *dropout* mencegah ko-adaptasi antarneuron.

Konfigurasi Pelatihan dan Optimasi

Pelatihan model memerlukan konfigurasi yang cermat terhadap algoritme optimasi, fungsi *loss*, dan strategi regularisasi guna mencapai kinerja yang baik sembari menghindari masalah umum seperti *overfitting* atau ketidakstabilan pelatihan. Penelitian ini menggunakan optimizer Adam [27], sebuah varian *stochastic gradient descent* yang menyesuaikan *learning rate* untuk setiap parameter berdasarkan estimasi momen pertama dan kedua dari gradien [28]. *Learning rate* awal ditetapkan sebesar 0,001, nilai default yang umum digunakan dan memberikan keseimbangan yang wajar antara kecepatan dan stabilitas pelatihan. *Binary cross-entropy* digunakan sebagai fungsi *loss*, sesuai untuk permasalahan klasifikasi dua kelas pada penelitian ini. Untuk satu sampel, *loss* ini dihitung sebagai:

$$Loss = -[y \log \hat{y} + (1 - y) \log 1 - \hat{y}] \qquad (2)$$

Dengan y menyatakan label biner sebenarnya (0 untuk normal, 1 untuk serangan) dan $\hat{y}$ menyatakan probabilitas prediksi model. Fungsi *loss* ini memberikan penalti yang besar terhadap prediksi salah yang percaya diri, sementara penalti yang moderat diberikan untuk prediksi yang tidak pasti, sehingga mendorong model agar terkalibrasi dengan baik dalam estimasi keyakinannya. *Loss* total pada satu batch yang terdiri atas N sampel diperoleh dengan merata-ratakan *loss* masing-masing sampel, yang dioptimasi melalui *gradient descent* secara iteratif [28].

Pelatihan dilakukan dalam mini-batch berukuran 128 sampel, kompromi yang baik antara kualitas estimasi gradien dan efisiensi komputasi. Batch lebih kecil menambah noise gradien yang dapat membantu keluar dari local minima buruk namun butuh lebih banyak iterasi, sedangkan batch lebih besar memberi gradien stabil tetapi boros memori dan berpotensi konvergen pada minima tajam dengan generalisasi buruk. Early stopping memantau validation loss dan menghentikan pelatihan bila tidak ada peningkatan selama 10 epoch berturut-turut, dengan bobot dikembalikan ke epoch berkinerja validasi terbaik, sehingga mencegah komputasi berlebih sekaligus overfitting.

Reduksi learning rate pada kondisi plateau melengkapi early stopping: bila validation loss tidak membaik selama 6 epoch, learning rate dikalikan 0,5 hingga batas minimum 1×10^{-7}, sehingga optimizer dapat menyempurnakan model atau keluar dari local minima dangkal—umumnya menurun 2–3 kali selama pelatihan. Strategi alternatif seperti cyclical learning rate [29] dipertimbangkan namun pendekatan berbasis plateau memadai. Model checkpointing menyimpan bobot setiap kali akurasi validasi membaik, memastikan model terbaik tetap tersimpan dan memungkinkan pelatihan dilanjutkan bila terhenti.

Regularisasi L2 diterapkan pada seluruh lapisan fully connected dengan weight decay 0,001, menambahkan penalti proporsional terhadap kuadrat magnitudo bobot agar jaringan tidak mengembangkan bobot terlalu besar yang mengindikasikan overfitting [21]. Nilai ini dipilih melalui eksperimen pendahuluan agar memberi manfaat regularisasi tanpa membatasi kapasitas model. Jumlah maksimum epoch ditetapkan 100, meskipun kedua model konvergen jauh lebih awal melalui early stopping.

Metrik Evaluasi

Evaluasi yang komprehensif terhadap sistem deteksi intrusi memerlukan pemeriksaan berbagai dimensi kinerja, tidak hanya akurasi sederhana [30], terutama mengingat perbedaan biaya antara *false positive* dan *false negative* dalam konteks keamanan. Penelitian ini menghitung enam metrik yang saling melengkapi untuk mencirikan kinerja model secara kolektif. Akurasi mengukur proporsi keseluruhan prediksi yang benar pada kedua kelas:

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (3)$$

dengan TP menyatakan *true positive* (serangan yang terklasifikasi dengan benar), TN menyatakan *true negative* (lalu lintas normal yang terklasifikasi dengan benar), FP menyatakan *false positive* (lalu lintas normal yang salah diklasifikasikan sebagai serangan), dan FN menyatakan *false negative* (serangan yang salah diklasifikasikan sebagai normal). Meskipun akurasi memberikan ringkasan statistik yang bermanfaat, metrik ini dapat menyesatkan pada dataset yang tidak seimbang karena akurasi tinggi dapat diperoleh hanya dengan memprediksi kelas mayoritas. Presisi mengukur proporsi prediksi positif yang benar-benar tepat:

$$Precision = \frac{TP}{(TP+FP)} \quad (4)$$

Metrik ini berkaitan langsung dengan tingkat *false alarm* pada sistem deteksi intrusi. Presisi yang rendah mengindikasikan banyaknya *false positive*, yang dalam kondisi operasional dapat memicu *alert fatigue*, yaitu kondisi ketika analis keamanan kewalahan oleh peringatan yang salah dan berpotensi melewatkan ancaman yang sesungguhnya. Presisi yang tinggi sangat penting untuk menjaga efektivitas analis dan kredibilitas sistem. *Recall*, yang juga disebut sensitivitas atau *true positive rate*, mengukur proporsi serangan sesungguhnya yang berhasil terdeteksi:

$$Recall = \frac{TP}{(TP+FN)} \quad (5)$$

Metrik ini merepresentasikan kemampuan sistem dalam mengidentifikasi aktivitas berbahaya ketika aktivitas tersebut terjadi. *Recall* yang rendah berarti banyak serangan yang tidak terdeteksi, yang berpotensi memungkinkan penyerang mengompromikan sumber daya jaringan. Dalam aplikasi keamanan, *recall* sering diprioritaskan dibandingkan presisi karena biaya akibat melewatkan serangan yang sesungguhnya umumnya lebih besar dibandingkan biaya menyelidiki *false alarm*. F1-score memberikan rata-rata harmonik antara presisi dan *recall*, sehingga menghasilkan satu metrik yang menyeimbangkan kedua aspek tersebut:

$$F1 = \frac{2\times(Precision \times Recall)}{(Precision+Recall)} \quad (6)$$

Rata-rata harmonik memberikan bobot yang lebih besar pada nilai yang lebih kecil, sehingga F1-score hanya bernilai tinggi apabila presisi maupun *recall* sama-sama tinggi. Sifat ini menjadikan F1-score sangat bermanfaat untuk membandingkan model-model yang salah satunya mungkin mencapai presisi tinggi dengan mengorbankan *recall*, atau sebaliknya. *Receiver Operating Characteristic Area Under Curve* (ROC-AUC) mengevaluasi kinerja pengklasifikasi pada seluruh ambang klasifikasi yang mungkin [31]. Kurva ROC memplot *true positive rate* terhadap *false positive rate* seiring ambang keputusan berubah dari 0 hingga 1. Luas area di bawah kurva ini berkisar dari 0,5 untuk pengklasifikasi acak hingga 1,0 untuk pengklasifikasi sempurna. ROC-AUC memberikan penilaian kualitas model yang tidak bergantung pada ambang tertentu, sehingga dapat memperlihatkan apakah probabilitas yang diprediksi model secara efektif memisahkan kedua kelas terlepas dari ambang klasifikasi yang ditetapkan. Matriks konfusi memberikan wawasan tambahan dengan menampilkan distribusi lengkap prediksi pada kelas sebenarnya dan kelas hasil prediksi. Penelitian ini menghitung matriks konfusi untuk kedua model pada set pengujian, sehingga memungkinkan analisis rinci terhadap pola kesalahan serta perbandingan bagaimana kedalaman arsitektur memengaruhi jenis kesalahan klasifikasi yang berbeda. Matriks ini terbukti sangat bermanfaat untuk memahami apakah model menunjukkan bias sistematis terhadap jenis kesalahan tertentu.

 https://doi.org/10.56211/blendsains.v5i1.1841

HASIL DAN PEMBAHASAN

Analisis Kinerja Keseluruhan

Evaluasi eksperimental memperlihatkan perbedaan kinerja yang substansial antara arsitektur CNN dasar dan menengah, dengan model yang lebih dalam menunjukkan keunggulan yang jelas pada seluruh metrik yang diukur. Tabel 3 menyajikan perbandingan komprehensif indikator kinerja untuk kedua model pada set pengujian yang terpisah (*held-out test set*).

Tabel 3. Perbandingan Kinerja Arsitektur CNN

Model	Akurasi (%)	Presisi (%)	Recall (%)	F1-Score (%)	ROC-AUC	Waktu Latih (menit)
CNN-1 (Dasar)	94,36	95,18	96,04	95,61	0,9898	245
CNN-2 (Menengah)	97,91	98,89	97,82	98,35	0,9979	387
Peningkatan Absolut	*+3,55*	*+3,71*	*+1,78*	*+1,78*	*+0,0081*	*+142*
Peningkatan Relatif	*+3,76%*	*+3,90%*	*+1,85%*	*+2,87 %*	*+0,82%*	*+57,96%*

CNN-2 mencapai akurasi 97,91%, mengklasifikasikan dengan benar 37.842 dari 38.651 rekaman, peningkatan 3,55 poin persentase dibandingkan CNN-1 (94,36%) atau setara sekitar 1.000 klasifikasi benar tambahan. Presisi meningkat dari 95,18% menjadi 98,89%, mengindikasikan bahwa ketika CNN-2 menandai lalu lintas sebagai berbahaya, model benar hampir 99% kasus, sehingga menghasilkan sangat sedikit false alarm dalam penerapan operasional. Recall meningkat dari 96,04% menjadi 97,82%, setara dengan teridentifikasinya sekitar 450 serangan tambahan yang akan terlewat oleh CNN-1. F1-score 98,35% merepresentasikan keseimbangan presisi dan recall yang sangat baik tanpa mengorbankan salah satunya. Nilai ROC-AUC 0,9979 mendekati maksimum teoretis, berarti terdapat probabilitas 99,79% bahwa model memberi skor berbahaya lebih tinggi pada aliran serangan dibandingkan aliran normal, menandakan model mempelajari representasi pola serangan yang tangguh dan bukan menghafal contoh pelatihan. Perbedaan waktu pelatihan juga patut dipertimbangkan. CNN-2 membutuhkan 387 menit dibandingkan 245 menit untuk CNN-1, peningkatan 58% yang beralasan mengingat perolehan kinerjanya. Untuk skenario pengembangan offline dan pelatihan ulang periodik yang umum pada sistem deteksi intrusi, tambahan waktu ini tidak menimbulkan kendala praktis. Kedua model konvergen jauh sebelum batas 100 epoch melalui early stopping, dengan CNN-1 berhenti pada epoch ke-55 dan CNN-2 pada epoch ke-60, menandakan optimasi yang efisien.

Analisis Kesalahan Secara Rinci Melalui Matriks Konfusi

Matriks konfusi memberikan wawasan yang lebih terperinci mengenai perilaku model dengan memperlihatkan pola spesifik dari prediksi yang benar maupun kesalahan yang terjadi. Matriks ini menjadi sangat bermanfaat khususnya untuk aplikasi keamanan, mengingat biaya akibat *false positive* dan *false negative* berbeda secara substansial.

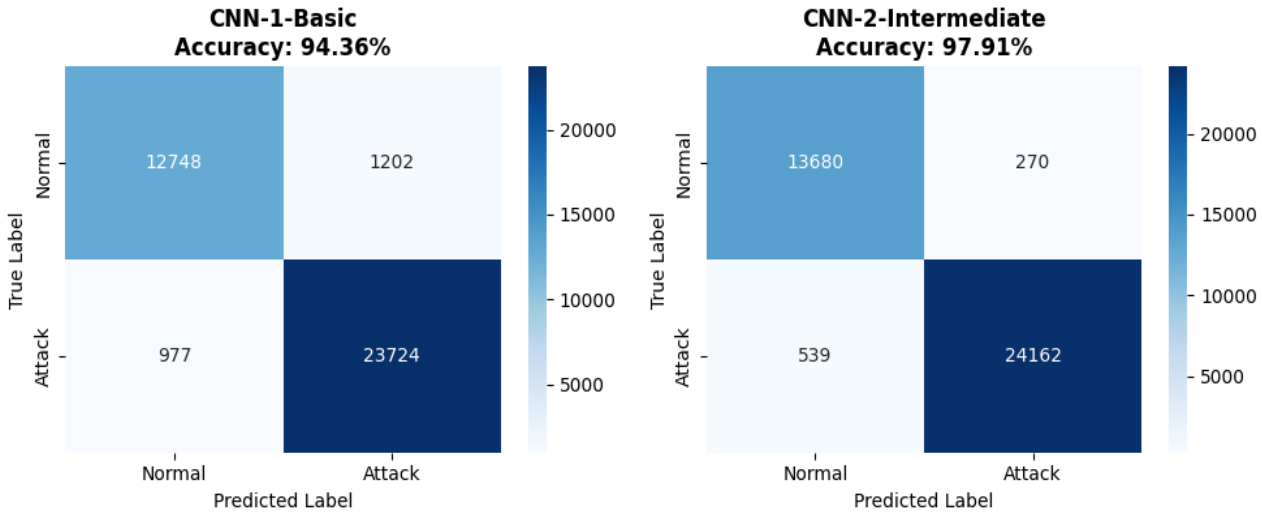
Tabel 4. Matriks Konfusi untuk CNN-1 (Arsitektur Dasar)

	Prediksi Normal	Prediksi Serangan	Total
Aktual Normal	12.748 (91,4%)	1.202 (8,6%)	13.950
Aktual Serangan	977 (4,0%)	23.724 (96,0%)	24.701
Total	13.725	24.926	38.651

Matriks konfusi CNN-1 menunjukkan bahwa dari 13.950 aliran normal, model mengidentifikasi 12.748 dengan benar namun salah mengklasifikasikan 1.202 sebagai serangan. Tingkat false positive 8,6% ini berarti sekitar satu dari dua belas aliran normal memicu alarm, berpotensi menghasilkan puluhan hingga ratusan peringatan palsu harian pada jaringan sibuk dan memicu alert fatigue, yaitu kondisi analis mulai mengabaikan peringatan otomatis. Dari 24.701 aliran serangan, CNN-1 mendeteksi 23.724 namun melewatkan 977 (false negative 4,0%). Serangan yang terlewat ini lebih mengkhawatirkan dibandingkan false positive karena penyerang berpotensi mengeksfiltrasi data, menyebarkan malware, atau menanamkan backdoor persisten, sehingga serangan yang gagal terdeteksi perlu diselidiki lebih lanjut sebagai dasar perbaikan model.

Tabel 5. Matriks Konfusi untuk CNN-2 (Arsitektur Menengah)

	Prediksi Normal	**Prediksi Serangan**	**Total**
Aktual Normal	13.680 (98,1%)	270 (1,9%)	13.950
Aktual Serangan	539 (2,2%)	24.162 (97,8%)	24.701
Total	14.219	24.432	38.651



Gambar 1. Matriks konfusi untuk kedua model CNN.
(Kiri) CNN-1-Basic menunjukkan 12.748 TN, 1.202 FP, 977 FN, dan 23.724 TP dengan akurasi 94,36%.
(Kanan) CNN-2-Intermediate menunjukkan 13.680 TN, 270 FP, 539 FN, dan 24.162 TP dengan akurasi 97,91%.

CNN-2 menunjukkan karakteristik kesalahan yang membaik dramatis, dengan hanya 270 false positive dan 539 false negative. Tingkat false positive 1,93% merepresentasikan penurunan 77,5% dibandingkan CNN-1, yaitu dari satu false alarm per dua belas aliran menjadi satu per lima puluh dua aliran, sehingga secara substansial mengurangi beban kerja analis sekaligus menjaga kewaspadaan dan kepercayaan terhadap sistem otomatis. Tingkat false negative turun menjadi 2,18%, berarti CNN-2 menangkap tambahan 438 serangan yang akan terlewat oleh CNN-1 (penurunan 44,8%), yang krusial untuk mencegah intrusi sebelum menimbulkan kerugian serius. Perbaikan simultan pada kedua jenis kesalahan—bukan sekadar mengorbankan salah satu—mengindikasikan CNN-2 mempelajari fitur yang lebih diskriminatif dan bukan sekadar menyesuaikan ambang. Dari 270 false positive yang tersisa, banyak melibatkan lalu lintas normal atipikal seperti transfer bervolume tinggi atau kombinasi protokol tak lazim, sementara 539 false negative terkonsentrasi pada serangan berfrekuensi rendah dengan representasi terbatas pada data pelatihan—menandakan varian serangan langka masih menjadi tantangan yang dapat dibantu oleh sampling khusus atau deteksi anomali tambahan.

Dinamika Pelatihan dan Perilaku Konvergensi

Analisis terhadap kurva pelatihan memberikan wawasan yang bermanfaat mengenai dinamika optimasi dan membantu memverifikasi bahwa perbedaan kinerja yang teramati berasal dari peningkatan pembelajaran yang sesungguhnya dan bukan sekadar artefak dari inisialisasi acak atau ketidakstabilan optimasi [32]. Gambar 2 menampilkan riwayat pelatihan lengkap yang memperlihatkan metrik akurasi, *loss*, presisi, dan *recall* pada seluruh epoch pelatihan untuk kedua model.

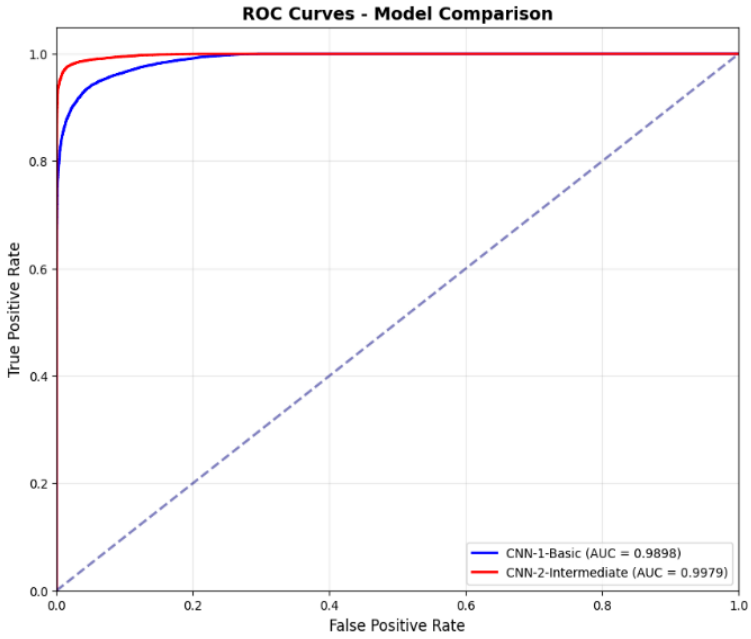
 https://doi.org/10.56211/blendsains.v5i1.1841

Gambar 2. Riwayat pelatihan model selama 60 epoch yang menampilkan empat metrik utama. (Kiri-atas) Progresi akurasi dengan CNN-2 secara konsisten mengungguli CNN-1. (Kanan-atas) Penurunan loss yang menunjukkan konvergensi yang stabil. (Kiri-bawah) Perkembangan presisi yang mencapai 98,89%. (Kanan-bawah) Perkembangan recall untuk kedua model.

Kedua arsitektur menunjukkan dinamika konvergensi yang sangat kontras. CNN-2 menunjukkan progresi yang stabil, monoton, dan konvergen secara harmonis pada seluruh metrik. Sebaliknya, CNN-1 memperlihatkan ketidakstabilan optimasi dan osilasi fluktuatif yang cukup tajam setelah *epoch* ke-30. CNN-1 menunjukkan peningkatan awal yang cepat pada 20 epoch pertama seiring model mempelajari pola diskriminatif mendasar, diikuti penyempurnaan lambat hingga *epoch* ke-55 ketika *early stopping* terpicu, dengan akurasi pelatihan sekitar 93.5% dan validasi tertahan di kisaran 92.5%–94%. Karakteristik ini mengindikasikan adanya *overfitting* dan osilasi kasar pada arsitektur dangkal tersebut. CNN-2 mengikuti pola serupa namun terus meningkat stabil hingga *epoch* ke-60, dengan konvergensi sedikit lebih lambat yang merefleksikan waktu tambahan untuk mengoptimasi parameter yang lebih banyak. Menariknya, CNN-2 mempertahankan *generalization gap* lebih ketat (1–2 poin) meskipun memiliki parameter hampir tiga kali lipat, berkat regularisasi yang lebih kuat berupa dropout lebih tinggi (0.4 dan 0.3 dibandingkan 0.3 dan 0.2) serta batch normalization setelah setiap konvolusi. Kurva validation loss memperlihatkan pola divergensi yang sangat informatif. Pada model CNN-2, *loss* menurun secara konsisten dan stabil mencapai nilai minimum rendah di sekitar *epoch* ke-55, mengonfirmasi keberhasilan model menghindari *overfitting* parah. Namun pada CNN-1, terjadi gejala *overfitting* yang parah; setelah *epoch* ke-30, meskipun *training loss* terus menurun, kurva *validation loss* (garis oranye) justru melonjak naik secara drastis dari 0.14 hingga menyentuh angka 0.19. Kurva presisi dan recall CNN-1 juga menunjukkan fluktuasi tajam menyerupai pola gergaji, berbeda dengan CNN-2 yang berkembang mulus beriringan tanpa *trade-off* dan konsisten berada di atas CNN-1 setelah *epoch* ke-15 dengan kesenjangan yang melebar. Mekanisme reduksi learning rate teraktivasi dua kali selama pelatihan CNN-2 pada epoch ke-38 dan ke-52, yang ditunjukkan oleh akselerasi mendadak pada tingkat penurunan *loss* guna membantu model keluar dari *plateau* dan mencapai kinerja yang tak tercapai dengan *learning rate* tetap [29]. Sebagai komparasi, CNN-1 hanya mengalami satu kali reduksi *learning rate* pada *epoch* ke-45, sebuah intervensi yang terbukti terlambat menstabilkan fluktuasi gradien akibat lemahnya kendala regularisasi pada arsitektur dasar tersebut.

Analisis Kurva ROC dan Pemilihan Ambang

Kurva *Receiver Operating Characteristic* [31] untuk kedua model mengungkap kapabilitas diskriminasi pada seluruh spektrum ambang klasifikasi. Sebagaimana ditunjukkan pada Gambar 3, kurva ROC CNN-2 (garis merah, AUC=0,9979) merapat lebih ketat ke sudut kiri-atas dibandingkan kurva CNN-1 (garis biru, AUC=0,9898), yang mengindikasikan kinerja yang lebih unggul pada hampir seluruh titik operasi.



Gambar 3. Perbandingan kurva ROC menunjukkan CNN-1-Basic (AUC=0,9898) dan CNN-2-Intermediate (AUC=0,9979) Kurva yang lebih landai pada CNN-2 menunjukkan kapabilitas diskriminasi yang superior pada seluruh ambang keputusan.

Pada ambang standar 0,5, CNN-2 mencapai true positive rate 97,82% dengan false positive hanya 1,93%. Bahkan pada ambang lebih konservatif yang menghasilkan false positive 1%, CNN-2 masih mendeteksi sekitar 96% serangan sementara CNN-1 menurun hingga sekitar 92% pada tingkat yang sama. Fleksibilitas ambang ini bermanfaat secara operasional: organisasi dengan kapasitas analis terbatas dapat memilih false positive sangat rendah agar setiap peringatan dapat diselidiki menyeluruh, sedangkan lingkungan berkeamanan tinggi dapat menerima false positive lebih besar untuk memaksimalkan deteksi—kurva ROC yang landai pada kedua model menunjukkan trade-off ini dapat dioptimasi tanpa pengorbanan kinerja drastis. Skor ROC-AUC 0,9898 (CNN-1) dan 0,9979 (CNN-2) berarti bahwa dalam perbandingan acak antara aliran normal dan serangan, model memberi probabilitas serangan lebih tinggi pada serangan sesungguhnya dengan keberhasilan 98,98% dan 99,79%, mendekati batas kinerja maksimum dan sebanding dengan literatur terbaru pada dataset yang sama.

Analisis Komparatif

Hasil penelitian ini perlu disandingkan dengan studi terdahulu yang menggunakan dataset serupa untuk mendapatkan gambaran riset yang utuh. Melalui Tabel 6, perbandingan tersebut diperluas dengan melibatkan beberapa metode mutakhir lintas dataset dan metodologi

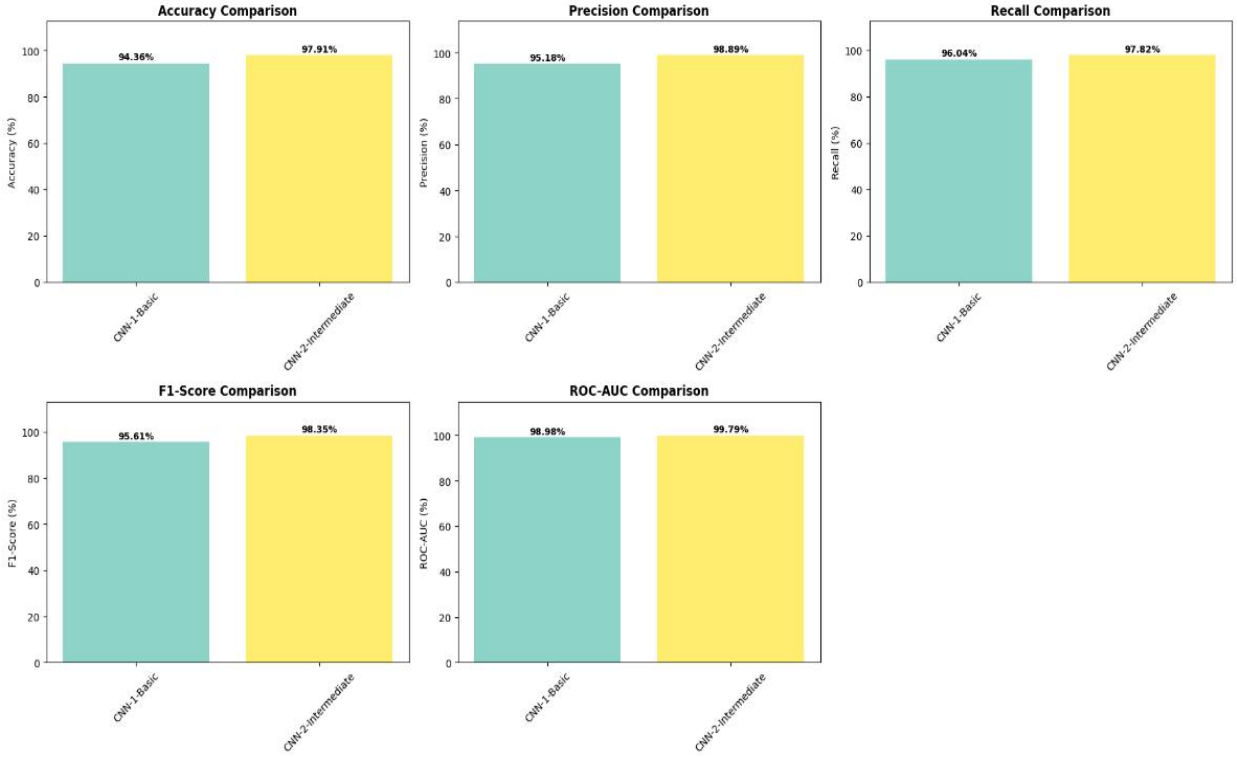
Tabel 6. Perbandingan dengan Metode Deteksi Intrusi Mutakhir

Penelitian	Metode	Dataset	Tahun	Akurasi (%)	Presisi (%)	Recall (%)	F1-Score (%)	AUC
Hozouri dkk. [33]	Survei IDS Deep Learning	Multipel	2025	-	-	-	-	-
Vibhute dkk. [12]	CNN + RF Feature Selection	UNSW-NB15	2024	-	-	-	-	-
Thaljaoui [13]	CNN-LSTM Hibrida	UNSW-NB15	2025	98,00	-	-	97,00	-
Vinayakumar dkk. [34]	Deep CNN	UNSW-NB15	2019	95,67	94,33	96,21	95,26	0,9689
Wang dkk. [14]	CNN-RNN Hibrida	CICIDS2017	2021	97,89	97,45	98,12	97,78	0,9834

 https://doi.org/10.56211/blendsains.v5i1.1841

Khraisat dkk. [35]	Ensemble DL	UNSW-NB15	2019	95,25	94,78	95,67	95,22	0,9645
Hindy dkk. [36]	Hybrid DNN	UNSW-NB15	2020	96,78	96,23	96,89	96,56	0,9756
Yin dkk. [37]	RNN	NSL-KDD	2017	95,89	95,34	96,12	95,73	0,9712
Rahman dkk. [38]	ML Ensemble	IoT-23	2020	94,56	93,89	95,23	94,55	0,9589
CNN-1 (Penelitian Ini)	**CNN Dasar**	**UNSW-NB15**	**2026**	**94,36**	**95,18**	**96,04**	**95,61**	**0,9898**
CNN-2 (Penelitian Ini)	**CNN Menengah**	**UNSW-NB15**	**2026**	**97,91**	**98,89**	**97,82**	**98,35**	**0,9979**

Ketika dibatasi pada metode yang dievaluasi khusus pada UNSW-NB15, CNN-2 mencapai akurasi tertinggi yang terdokumentasi di antara pendekatan deep learning yang dibandingkan. Vibhute dkk. memadukan CNN dengan seleksi fitur random forest [12], strategi yang menambah kompleksitas alur kerja. Thaljaoui melaporkan akurasi sekitar 98% dengan CNN-LSTM hibrida beroptimasi Bayesian [13], sebanding namun lebih kompleks dibandingkan CNN-2 yang murni konvolusional. Vinayakumar dkk. mencapai 95,67% [34] dan Hindy dkk. 96,78% [36], keduanya di bawah 97,91% penelitian ini. Yang lebih signifikan, ROC-AUC 0,9979 secara substansial melampaui hampir seluruh metode pembanding pada UNSW-NB15. Pada perbandingan antardataset, Wang dkk. melaporkan 97,89% pada CICIDS2017 dengan CNN-RNN hibrida [14], hampir menyamai namun dengan arsitektur lebih kompleks. Gambar 4 menyajikan perbandingan visual seluruh metrik antarmodel.



Gambar 4. Perbandingan kinerja secara komprehensif pada lima metrik (Akurasi, Presisi, Recall, F1-Score, ROC-AUC) untuk CNN-1-Basic dan CNN-2-Intermediate. CNN-2 menunjukkan keunggulan yang konsisten pada seluruh dimensi evaluasi, dengan kinerja yang sangat kuat khususnya pada presisi dan ROC-AUC.

Beberapa aspek dari perbandingan ini patut digarisbawahi. Pertama, arsitektur CNN-2 yang relatif lebih sederhana pada penelitian ini mencapai kinerja yang kompetitif, bahkan superior, dibandingkan model hibrida kompleks yang mengombinasikan berbagai jenis jaringan. Hal ini mengindikasikan bahwa kompleksitas arsitektur di luar suatu ambang tertentu dapat menghasilkan keuntungan yang semakin berkurang (*diminishing returns*) untuk deteksi intrusi jaringan, setidaknya pada dataset acuan saat ini. Kedua, presisi penelitian ini sebesar 98,89% sangat menonjol, melampaui seluruh

metode pembanding dengan selisih yang substansial dan mengindikasikan keandalan yang luar biasa dalam klasifikasi serangan. Ketiga, kinerja model dasar CNN-1 yang tetap kuat menunjukkan bahwa arsitektur konvolusi yang relatif sederhana sekalipun dapat mencapai hasil yang kompetitif, dengan akurasi 94,36% dan AUC 0,9898 yang menempatkannya pada tingkat menengah di antara pendekatan *deep learning* terbaru. Perbandingan-perbandingan ini memvalidasi pilihan arsitektur dan metodologi pelatihan pada penelitian ini sekaligus menyoroti kemajuan yang berkelanjutan dalam deteksi intrusi berbasis *deep learning*. Peningkatan yang konsisten pada berbagai dimensi kinerja, dan bukan hanya unggul pada satu metrik saja, semakin memperkuat keyakinan terhadap penerapan praktis dari pendekatan yang diusulkan.

Kedalaman Arsitektur dan Pembelajaran Fitur

Progresi kinerja dari CNN-1 ke CNN-2 memberikan bukti empiris mengenai nilai dari kedalaman arsitektur yang moderat untuk analisis lalu lintas jaringan. Relasi ini tampak lebih kompleks dibandingkan pada domain *computer vision*, di mana jaringan yang lebih dalam hampir selalu mengungguli jaringan yang lebih dangkal, sebagaimana ditunjukkan oleh progresi dari AlexNet menuju VGGNet hingga ResNet. Peningkatan akurasi sebesar 3,55% dan peningkatan AUC sebesar 0,0081 yang diperoleh melalui penambahan dua blok konvolusi merepresentasikan peningkatan yang substansial namun tidak dramatis, yang mengindikasikan bahwa data lalu lintas jaringan mungkin memiliki kebutuhan kedalaman yang berbeda dibandingkan data visual. Struktur tiga blok pada CNN-2 memungkinkan suatu bentuk pembelajaran fitur hierarkis yang terwujud secara berbeda dibandingkan pada klasifikasi citra. Blok konvolusi pertama dengan 32 filter mempelajari pola statistik dasar pada fitur jaringan, seperti distribusi ukuran paket yang tidak lazim, kombinasi *flag* protokol yang abnormal, atau pola waktu yang mencurigakan. Fitur tingkat rendah ini kurang lebih setara dengan fitur *hand-crafted* yang mungkin dirancang oleh pakar keamanan berdasarkan pengetahuan domain. Blok kedua dengan 64 filter mengombinasikan pola-pola dasar tersebut menjadi representasi tingkat menengah yang menangkap perilaku pada tingkat aliran, seperti signature *port scanning*, pola koneksi yang tidak lazim, atau karakteristik eksfiltrasi data. Blok ketiga dengan 128 filter mensintesiskan fitur-fitur tingkat menengah tersebut menjadi signature serangan tingkat tinggi yang membedakan pola serangan terkoordinasi dari aktivitas yang sah. Bukti untuk pembelajaran hierarkis semacam ini diperoleh dari pemeriksaan pola aktivasi filter antarblok, meskipun visualisasi yang rinci cukup menantang mengingat sifat data yang non-visual [39]. Analisis statistik terhadap respons filter mengungkap bahwa filter pada blok pertama teraktivasi secara kuat sebagai respons terhadap anomali fitur lokal terlepas dari jenis serangannya, sementara filter pada blok ketiga menunjukkan aktivasi yang lebih selektif untuk kategori serangan tertentu. Progresi dari deteksi anomali umum menuju pengenalan serangan spesifik ini mencerminkan hierarki yang teramati pada sistem deteksi intrusi berbasis aturan yang dirancang dengan baik.

Peningkatan yang relatif moderat akibat penambahan kedalaman, dibandingkan dengan aplikasi *computer vision*, kemungkinan merefleksikan perbedaan fundamental antara data lalu lintas jaringan dan data citra. Citra memiliki hierarki spasial yang kuat, di mana piksel yang berdekatan saling berkaitan secara erat sementara piksel yang berjauhan dapat bersifat independen, sehingga secara alami sesuai dengan *receptive field* yang semakin besar pada jaringan yang lebih dalam. Fitur jaringan tidak memiliki struktur lokalitas inheren semacam ini, dengan relasi antarfitur lebih bergantung pada kesamaan semantik dibandingkan posisi pada vektor fitur. Akibatnya, manfaat kedalaman dalam menangkap pola yang semakin global mungkin kurang signifikan untuk data jaringan. Strategi regularisasi terbukti esensial untuk merealisasikan manfaat dari peningkatan kedalaman tanpa terjerumus pada *overfitting*. Rasio *dropout* progresif yang meningkat dari 0,2 menjadi 0,4 pada lapisan-lapisan serta *batch normalization* setelah setiap konvolusi membantu CNN-2 mempertahankan generalisasi yang baik meskipun memiliki parameter tiga kali lebih banyak dibandingkan CNN-1. Eksperimen pendahuluan tanpa teknik regularisasi ini menunjukkan CNN-2 mengalami *overfitting* yang parah, dengan akurasi pelatihan melampaui 99% sementara akurasi validasi mengalami stagnasi di sekitar 93%, lebih buruk dibandingkan CNN-1 yang teregularisasi dengan baik. Hal ini menunjukkan bahwa kedalaman arsitektur semata tidak menjamin peningkatan kinerja tanpa regularisasi yang tepat [40].

Implikasi Praktis bagi Penerapan Operasional

Temuan ini memiliki implikasi praktis bagi organisasi yang mempertimbangkan penerapan deteksi intrusi berbasis CNN. Presisi CNN-2 yang tinggi (98,89%) berarti analis dapat mempercayai peringatan sistem, sehingga mengurangi waktu menyelidiki false alarm. Dengan asumsi volume satu juta aliran harian pada jaringan perusahaan menengah, false positive CNN-2 sebesar 1,93% menghasilkan sekitar 19.300 peringatan, sementara CNN-1 dengan 8,6% menghasilkan sekitar 86.000 peringatan—perbedaan yang sangat signifikan terhadap beban kerja Security Operations Center. Selisih ini

 https://doi.org/10.56211/blendsains.v5i1.1841

berimplikasi langsung pada efisiensi alokasi sumber daya analis dan membebaskan kapasitas untuk tugas investigasi yang lebih bernilai. Persyaratan komputasi kedua model tetap wajar untuk penerapan operasional. Meskipun CNN-2 membutuhkan waktu pelatihan 58% lebih lama, latensi inferensi pada produksi tetap minimal karena kompleksitasnya rendah dibandingkan model deep learning berskala besar, menjadikan keduanya sesuai untuk deteksi mendekati waktu nyata. Dari sudut pandang penulis, arsitektur CNN-2 paling ideal diimplementasikan pada perangkat keras tingkat server (server-grade), misalnya sebagai mesin inferensi terpusat pada platform Security Operations Center atau sensor NIDS khusus, karena ketersediaan memori dan kemampuan paralelisasi pada kelas perangkat tersebut memungkinkan inferensi ber-throughput tinggi tanpa modifikasi model. Sebaliknya, penerapan langsung pada perangkat jaringan edge seperti router atau firewall appliance yang sumber dayanya terbatas kemungkinan besar menuntut kompresi model terlebih dahulu melalui teknik seperti pruning atau kuantisasi, sehingga CNN-1 yang lebih ringan dapat menjadi alternatif yang lebih realistis untuk kelas perangkat ini. Bagi organisasi dengan sumber daya terbatas, CNN-1 tetap layak dengan akurasi 94,36% yang melampaui banyak pendekatan konvensional, sementara organisasi berkapasitas lebih besar memperoleh manfaat dari kinerja CNN-2. Hal ini sejalan dengan tren pengembangan arsitektur ringan untuk perangkat tepi, seperti kerangka berbasis genetic algorithm dan LSTM untuk lingkungan IoT [41], sehingga kedua arsitektur berpotensi diadaptasi lebih lanjut.

Keterbatasan Penelitian

Beberapa keterbatasan perlu diperhatikan. Pertama, evaluasi dibatasi pada satu dataset (UNSW-NB15) sehingga generalisasi pada lingkungan atau jenis serangan lain memerlukan validasi lanjutan. Kedua, penelitian berfokus pada klasifikasi biner dan belum mengeksplorasi klasifikasi multikelas yang dapat mengidentifikasi jenis serangan spesifik. Ketiga, dua arsitektur yang dibandingkan hanya merepresentasikan dua titik dalam spektrum kedalaman; eksplorasi rentang yang lebih luas (empat blok konvolusi atau lebih) dapat memberikan pemahaman lebih mendalam. Keempat, analisis komputasi terbatas pada waktu pelatihan dan tidak mengkaji konsumsi energi atau kelayakan penerapan pada perangkat tepi.

Penelitian mendatang dapat menelusuri beberapa arah. Transfer learning antar lingkungan dan jenis serangan [42], [43] berpotensi mengurangi kebutuhan data pelatihan saat model diterapkan pada konteks baru, mengingat representasi fitur lapisan awal cenderung dapat ditransfer antar-domain serupa. Pengembangan varian lebih ringan melalui pruning atau knowledge distillation memungkinkan penerapan pada gateway IoT dan network switch dengan sumber daya terbatas. Terakhir, penggabungan pelatihan adversarial berpotensi meningkatkan ketahanan terhadap serangan evasion, pertimbangan yang semakin penting mengingat kerentanan model deep learning terhadap manipulasi input halus.

KESIMPULAN DAN SARAN

Penelitian ini menyajikan analisis komparatif dua arsitektur Convolutional Neural Network dengan tingkat kedalaman yang berbeda untuk deteksi anomali jaringan menggunakan dataset UNSW-NB15. Arsitektur CNN menengah dengan tiga blok konvolusi, yang diberi nama CNN-2, secara konsisten mengungguli arsitektur CNN dasar dengan satu blok konvolusi (CNN-1) pada seluruh metrik evaluasi yang diukur. CNN-2 mencapai akurasi 97,91%, presisi 98,89%, recall 97,82%, F1-score 98,35%, dan ROC-AUC 0,9979, merepresentasikan peningkatan absolut sebesar 3,55 poin persentase pada akurasi dibandingkan CNN-1. Yang lebih penting bagi penerapan praktis, CNN-2 berhasil menurunkan tingkat false positive sebesar 77,5% dan tingkat false negative sebesar 44,8%, sehingga secara signifikan meningkatkan kelayakan operasional sistem dengan mengurangi beban kerja analis keamanan sekaligus menangkap lebih banyak serangan yang sesungguhnya. Temuan ini memberikan bukti bahwa kedalaman arsitektur memainkan peran penting dalam kinerja deteksi intrusi jaringan berbasis CNN, meskipun relasi tersebut tampak berbeda dari pola yang umum diamati pada aplikasi computer vision. Penambahan dua blok konvolusi disertai strategi regularisasi yang sesuai, meliputi peningkatan rasio dropout dan penerapan batch normalization yang lebih ekstensif, terbukti sukses memperkeras arsitektur CNN-2 dari risiko severe overfitting dan osilasi fluktuatif kasar yang justru terjadi pada model CNN-1 setelah epoch ke-30. Perbandingan dengan metode-metode mutakhir pada literatur terkini menunjukkan bahwa kinerja CNN-2 pada penelitian ini kompetitif, bahkan pada beberapa aspek superior, dibandingkan pendekatan hibrida yang jauh lebih kompleks, sehingga menegaskan bahwa kompleksitas arsitektur tambahan tidak selalu sepadan dengan manfaat kinerja yang diperoleh untuk tugas deteksi intrusi jaringan. Penelitian selanjutnya diarahkan untuk mengeksplorasi rentang kedalaman arsitektur yang lebih luas guna mengidentifikasi titik optimal antara kompleksitas model dan kinerja deteksi secara lebih sistematis. Validasi pada dataset tambahan, termasuk data lalu lintas jaringan produksi dunia nyata, akan memperkuat

generalisasi temuan ini di luar lingkungan UNSW-NB15. Perluasan ke arah klasifikasi multikelas yang dapat mengidentifikasi kategori serangan tertentu, alih-alih sekadar klasifikasi biner, akan memberikan nilai operasional yang lebih besar bagi tim respons insiden keamanan. Evaluasi terhadap ketahanan model dalam menghadapi serangan adversarial serta kajian mengenai interpretabilitas model melalui teknik seperti explainable AI juga merupakan arah penelitian yang penting untuk meningkatkan kepercayaan dan akuntabilitas sistem deteksi intrusi berbasis deep learning sebelum diterapkan secara luas pada infrastruktur keamanan siber.

DAFTAR PUSTAKA

[1] IBM, "Cost of a Data Breach Report 2025," 2025. [Online]. Available: https://www.ibm.com/reports/data-breach

[2] A. L. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016, doi: 10.1109/COMST.2015.2494502.

[3] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.

[4] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, Jan. 2015, doi: 10.1016/j.neunet.2014.09.003.

[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. [Online]. Available: https://www.deeplearningbook.org/

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas: IEEE, 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[8] C. Szegedy dkk., "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2015, pp. 1–9. doi: 10.1109/CVPR.2015.7298594.

[9] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *CoRR*, vol. abs/1409.1556, 2014, [Online]. Available: https://api.semanticscholar.org/CorpusID:14124313

[10] Wei Wang, Ming Zhu, Xuewen Zeng, Xiaozhou Ye, and Yiqiang Sheng, "Malware traffic classification using convolutional neural network for representation learning," in *2017 International Conference on Information Networking (ICOIN)*, IEEE, 2017, pp. 712–717. doi: 10.1109/ICOIN.2017.7899588.

[11] R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Applying convolutional neural network for network intrusion detection," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, IEEE, Sep. 2017, pp. 1222–1228. doi: 10.1109/ICACCI.2017.8126009.

[12] A. D. Vibhute, M. Khan, C. H. Patil, S. V. Gaikwad, A. V. Mane, and K. K. Patel, "Network anomaly detection and performance evaluation of Convolutional Neural Networks on UNSW-NB15 dataset," *Procedia Comput. Sci.*, vol. 235, pp. 2227–2236, 2024, doi: 10.1016/j.procs.2024.04.211.

[13] A. Thaljaoui, "Intelligent network intrusion detection system using optimized deep CNN-LSTM with UNSW-NB15," *International Journal of Information Technology*, Feb. 2025, doi: 10.1007/s41870-025-02416-0.

[14] Z. Wang, Y. Liu, D. He, and S. Chan, "Intrusion detection methods based on integrated deep learning model," *Comput. Secur.*, vol. 103, p. 102177, Apr. 2021, doi: 10.1016/j.cose.2021.102177.

[15] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, SCITEPRESS - Science and Technology Publications, 2018, pp. 108–116. doi: 10.5220/0006639801080116.

[16] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," Feb. 2017.

[17] N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *2015 Military Communications and Information Systems Conference (MilCIS)*, IEEE, Nov. 2015, pp. 1–6. doi: 10.1109/MilCIS.2015.7348942.

[18] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, IEEE, Jul. 2009, pp. 1–6. doi: 10.1109/CISDA.2009.5356528.

[19] N. Moustafa and J. Slay, "The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set," *Information Security Journal: A Global Perspective*, vol. 25, no. 1–3, pp. 18–31, Apr. 2016, doi: 10.1080/19393555.2015.1125974.
[20] François. Chollet, *Deep Learning with Python*, 2nd ed. Shelter Island, NY.: Manning Publications, 2021.
[21] Y. Bengio, "Practical Recommendations for Gradient-Based Training of Deep Architectures," 2012, pp. 437–478. doi: 10.1007/978-3-642-35289-8_26.
[22] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, Fort Lauderdale, FL, USA, 2011, pp. 315–323.
[23] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," Mar. 2015.
[24] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014, [Online]. Available: http://jmlr.org
[25] D. Scherer, A. Müller, and S. Behnke, "Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition," 2010, pp. 92–101. doi: 10.1007/978-3-642-15825-4_10.
[26] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 International Conference on Engineering and Technology (ICET)*, IEEE, Aug. 2017, pp. 1–6. doi: 10.1109/ICEngTechnol.2017.8308186.
[27] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," Jan. 2017.
[28] S. Ruder, "An overview of gradient descent optimization algorithms," Jun. 2017.
[29] L. N. Smith, "Cyclical Learning Rates for Training Neural Networks," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, Mar. 2017, pp. 464–472. doi: 10.1109/WACV.2017.58.
[30] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, p. 6, Dec. 2020, doi: 10.1186/s12864-019-6413-7.
[31] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006, doi: 10.1016/j.patrec.2005.10.010.
[32] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning (still) requires rethinking generalization," *Commun. ACM*, vol. 64, no. 3, pp. 107–115, Mar. 2021, doi: 10.1145/3446776.
[33] A. Hozouri, A. Mirzaei, and M. Effatparvar, "A comprehensive survey on intrusion detection systems with advances in machine learning, deep learning and emerging cybersecurity challenges," *Discover Artificial Intelligence*, vol. 5, no. 1, p. 314, Nov. 2025, doi: 10.1007/s44163-025-00578-1.
[34] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," *IEEE Access*, vol. 7, pp. 41525–41550, 2019, doi: 10.1109/ACCESS.2019.2895334.
[35] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, p. 20, Dec. 2019, doi: 10.1186/s42400-019-0038-7.
[36] H. Hindy dkk., "A Taxonomy of Network Threats and the Effect of Current Datasets on Intrusion Detection Systems," *IEEE Access*, vol. 8, pp. 104650–104675, 2020, doi: 10.1109/ACCESS.2020.3000179.
[37] C. Yin, Y. Zhu, J. Fei, and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017, doi: 10.1109/ACCESS.2017.2762418.
[38] M. A. Rahman, A. T. Asyhari, L. S. Leong, G. B. Satrya, M. Hai Tao, and M. F. Zolkipli, "Scalable machine learning-based intrusion detection system for IoT-enabled smart cities," *Sustain. Cities Soc.*, vol. 61, p. 102324, Oct. 2020, doi: 10.1016/j.scs.2020.102324.
[39] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," 2014, pp. 818–833. doi: 10.1007/978-3-319-10590-1_53.
[40] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, May 2013, pp. 8609–8613. doi: 10.1109/ICASSP.2013.6639346.
[41] Y. K. Saheed, O. H. Abdulganiyu, and T. A. Tchakoucht, "Modified genetic algorithm and fine-tuned long short-term memory network for intrusion detection in the internet of things networks with edge capabilities," *Appl. Soft Comput.*, vol. 155, p. 111434, Apr. 2024, doi: 10.1016/j.asoc.2024.111434.
[42] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks ?," in *27th International Conference on Neural Information Processing Systems*, MIT Press, 2014, pp. 3320–3328.

[43] K. Weiss, T. M. Khoshgoftaar, and D. Wang, “A survey of transfer learning,” *J. Big Data*, vol. 3, no. 1, p. 9, Dec. 2016, doi: 10.1186/s40537-016-0043-6.