Penyulingan Pengetahuan Multi Tahap dengan Konsep Sederhana Baru pada IDS untuk Penerapan Edge Ringan

Erwin Eka Syahputra
Fakultas Informatika
Universitas Telko
Bandung
erwineka@students.telkomuniversity.ac.id

Aji Gautama Putrad
Fakultas Informatik
Universitas Telkom
Bandung
ajigps@telkomuniversity.ac.id

Ryan Lingga Wicaksono
Fakultas Informatika
Universitas Telkom
Bandung
wicaksonoryan@telkomuniversity.ac.id

Penerapan deep learning dalam *Intrusion Detection System* (IDS) telah menunjukkan kemajuan yang signifikan dalam penelitian-penelitian sebelumnya. Namun, penggunaan *edge computing* dalam IDS memiliki dua keuntungan, mengurangi latensi dan meningkatkan privasi. Penelitian ini bertujuan untuk mengembangkan *Intrusion Detection System* (IDS) berbasis *edge computing* yang memanfaatkan *deep neural network* (DNN) sebagai model pembelajaran mesin dan arsitektur *Multi-Stage Knowledge Distillation* (MSKD) yang inovatif dan disederhanakan sebagai metodologi pelatihan. Data set yang digunakan berasal dari Kaggle, yaitu UNSW-NB15 benchmark dataset. Langkah berikutnya adalah menyiapkan data dan melatih proses MSKD. Pada penelitian ini, penulis membandingkan pendekatan MSKD dengan pendekatan *Knowledge Distillation* (KD) dan mengevaluasi kedua model tersebut. Metode MSKD baru menggunakan tiga model DNN: Teacher Model, Student Model 1, dan Student Model 2. Untuk melihat seberapa baik MSKD bekerja, dilakukan analisis terhadap akurasi, penurunan akurasi, ukuran model, dan tingkat kompresi (CR). Akurasi Teacher Model adalah 0,9955, Student Model 1 mendapatkan akurasi sebesar 0,9897, dan Student Model 2 mendapatkan akurasi sebesar 0,9857. Jika diperhatikan, akurasi Student Model 1 turun sebesar 0,58% dibandingkan dengan Teacher Model, dan akurasi Student Model 2 turun sebesar 0,99% dibandingkan dengan Teacher Model dan 0,40% dibandingkan dengan Student Model 1. Rasio kompresi terbaik untuk Student Model 1 adalah 101,9 kali lebih baik daripada Teacher Model dan 3,0 kali lebih baik daripada Student Model 1. Penelitian ini juga menganalisis bahwa penggunaan *soft label* menghasilkan rasio kompresi (CR) yang 1,8 kali lebih tinggi antara Student Model 2 dan Student Model 2 yang dilatih menggunakan *hard label*.

Kata kunci: multi-stage knowledge distillation, intrusion detection system, UNSW-NB15 dataset, edge computing.

The application of deep learning in intrusion detection systems (IDS) has shown significant progress in previous studies. However, the use of edge computing in IDS has two advantages: reduced latency and increased privacy. This study aims to develop an edge computing-based intrusion detection system (IDS) that utilizes deep neural networks (DNNs) as machine learning models and an innovative and simplified Multi-Stage Knowledge Distillation (MSKD) architecture as a training methodology. The dataset used comes from Kaggle, namely the UNSW-NB15 benchmark dataset. The next step is to prepare the data and train the MSKD process. In this study, the authors compare the MSKD approach with the Knowledge Distillation (KD) approach and evaluate both models. The new MSKD method uses three DNN models: Teacher Model, Student Model 1, and Student Model 2. To see how well MSKD works, an analysis of accuracy, accuracy degradation, model size, and compression rate (CR) is carried out. The accuracy of the Teacher Model is 0.9955, Student Model 1 gets an accuracy of 0.9897, and Student Model 2 gets an accuracy of 0.9857. If you pay attention, the accuracy of Student Model 1 drops by 0.58% compared to the Teacher Model, and the accuracy of Student Model 2 drops by 0.99% compared to the Teacher Model and 0.40% compared to Student Model 1. The best compression ratio for Student Model 1 is 101.9 times better than the Teacher Model and 3.0 times better than Student Model 1. This study also analyzes that the use of soft labels results in a compression ratio (CR) that is 1.8 times higher between Student Model 2 and Student Model 2 trained using hard labels.

Keywords: multi-stage knowledge distillation, intrusion detection system, UNSW-NB15 dataset, edge computing.

1. Pendahuluan

Komputasi tepi merupakan teknologi yang berkembang pesat dan rentan terhadap serangan. Seperti teknologi jaringan lainnya [1], teknologi ini juga memerlukan *Intrusion Detection System* (IDS). IDS merupakan bagian penting dari sistem keamanan jaringan yang mendeteksi perilaku mencurigakan atau ancaman serangan terhadap sistem [2]. Hal ini sangat penting bagi perkembangan komputasi tepi yang ringan karena memungkinkan pemantauan dan deteksi serangan yang menargetkan perangkat tepi tanpa mengganggu fungsi utama perangkat [3].

Deep Neural Network (DNN) merupakan model mendalam yang cocok untuk membangun IDS. Vigneswaran *et al*. [4], dalam penelitian mereka tentang DNN untuk IDS, menyatakan bahwa DNN lebih unggul dalam mendeteksi bagian nonlinier dan kompleks dari data lalu lintas jaringan. Penelitian DNN untuk IDS oleh Nguyen *et al*. [5] juga menunjukkan bahwa DNN sangat baik dalam mendeteksi pola serangan tersembunyi dan perilaku aneh dalam lalu lintas jaringan. Hal ini membuatnya ideal untuk membangun sistem deteksi yang akurat dan fleksibel.

Arsitektur DNN yang membutuhkan daya komputasi tinggi dan memori besar, di sisi lain, menjadi tantangan signifikan dalam konteks komputasi tepi. Oleh karena itu, penelitian Mishra *et al*. [6] menunjukkan bahwa *lightweight* DNNs sangat penting untuk menyeimbangkan kinerja dan efektivitas saat menggunakan IDS dalam komputasi tepi. *Multi-Stage Knowledge Distillation* (MSKD) merupakan salah satu metode yang menarik perhatian signifikan, karena bertujuan untuk mengoptimalkan model selama proses pembelajaran dengan ukuran yang relatif kecil.

Latar Belakang

Penelitian ini dilatar belakangi untuk mengembangkan IDS yang cukup ringan namun akurat untuk digunakan dalam lingkungan komputasi tepi. Masalah utama dalam konteks saat ini adalah, meskipun model DNN menunjukkan akurasi yang cukup tinggi dalam mengidentifikasi pola serangan jaringan, model ini

umumnya memiliki ukuran yang besar dan membutuhkan sumber daya komputasi yang signifikan. Hal ini menyebabkan model DNN tidak cocok untuk perangkat tepi yang tidak memiliki banyak memori penyimpanan. Penting untuk menemukan cara untuk menjembatani kesenjangan antara kemampuan deteksi serangan canggih dan kebutuhan efisiensi operasional pada perangkat tepi.

Penelitian ini memilih metode MSKD sebagai teknik kompresi model yang mempertahankan kinerja model. Metode ini memungkinkan model besar Teacher Model secara bertahap mentransfer pengetahuan yang dimilikinya ke dua model yang lebih kecil (Student Model 1 dan Student Model 2). MSKD dapat melakukan kompresi pada ukuran model menjadi jauh lebih kecil namun tetap memiliki akurasi yang tinggi dengan menggunakan *soft label* dan struktur distilasi bertahap. Dataset UNSW-NB15 merupakan dataset yang cocok untuk digunakan, karena dataset tersebut salah satu dataset yang menunjukkan cara kerja jaringan modern.

Multi-Stage Knowledge Distillation (MSKD) merupakan metode baru yang telah menarik perhatian signifikan karena tujuannya untuk mengoptimalkan model selama proses pembelajaran sambil mempertahankan ukuran yang relatif kompak. Xu *et al*. [7] menyatakan bahwa MSKD menggunakan strategi distilasi bertahap progresif, yang berbeda dari teknik distilasi standar yang biasanya hanya memerlukan satu tahap transfer pengetahuan dari Guru ke Murid. Hao *et al*. [8] menunjukkan bahwa dalam MSKD, proses transfer pengetahuan terjadi melalui beberapa tahap intermediet, menggunakan serangkaian Student Model bertingkat atau kombinasi teknik pembelajaran tambahan.

Banyak studi kasus telah memanfaatkan pendekatan MSKD secara efektif. Zeng *et al*. [9] menggunakannya untuk deteksi kondisi dermatologis. Salah satu artikel yang membahas metode MSKD untuk klasifikasi penyakit kulit adalah "*Dual-Stage Progressive KD for Skin Disease Classification*." Dalam studi lain, Zhou *et al*. [10] menggunakan MSKD untuk mendeteksi dan mengidentifikasi cacat permukaan pada wortel saat panen. Terdapat peluang penelitian untuk mengembangkan MSKD dalam studi kasus kompresi model pada komputasi tepi menggunakan IDS.

Topik dan Batasannya

Topik penelitian ini berfokus pada pengembangan IDS yang mengguakan model utama DNN melalui penerapan metodologi MSKD, dengan menggunakan arsitektur yang disederhanakan dan efisien yang dirancang khusus untuk memenuhi persyaratan komputasi dalam lingkungan komputasi tepi. Penelitian ini menggunakan arsitektur distilasi multi-tahap yang terdiri dari Teacher Model, Student Model 1, dan Student Model 2.

Penelitian ini melakukan analisa terhadap pemanfaatan dataset UNSW-NB15 sebagai sumber data serangan jaringan, pengaruh *soft label* dan *hard label* dalam proses distilasi, serta evaluasi kinerja berdasarkan akurasi, loss, ukuran model, dan tingkat kompresi (CR) model. Fokus penelitian ini adalah penggunaan arsitektur Jaringan Saraf Tiruan (JST) dengan hanya satu dataset benchmark UNSW-NB15 dan pengujian kinerja melalui simulasi prediksi di komputasi tepi.

Secara umum, penelitian ini menciptakan model IDS berbasis DNN yang dikompresi menggunakan metode MSKD. Hal ini membuat model menjadi jauh lebih ringan tanpa kehilangan akurasi yang signifikan. Penelitian ini bertujuan untuk menjelaskan dampak penggunaan *soft label* dan Adam optimizer terhadap ukuran model dan kinerja model, serta menghasilkan model turunan dengan tingkat kompresi yang cukup signifikan.

Tujuan

Tujuan dari penelitian ini adalah untuk mengembangkan model pada IDS berbasis model DNN yang akurat dan efektif untuk lingkungan komputasi tepi dengan menggunakan arsitektur MSKD yang mencakup proses distilasi dari model utama ke model turunan. Penelitian ini juga bertujuan untuk menganalisis dampak penggunaan *soft label* dan *hard label* serta peran Adam optimizer terhadap akurasi, kerugian akurasi, ukuran model, dan tingkat kompresi, guna menghasilkan model turunan yang optimal dengan ukuran yang jauh lebih ringan, prediksi cepat, dan kinerja kompetitif untuk implementasi pada perangkat tepi seperti Raspberry Pi.

Organisasi Tulisan

Struktur dari penulisan jurnal ini adalah, pada bab selanjutnya akan membahas studi terkait, kemudian dilanjutkan bab berikutnya yang membahas metodologi penelitian. Bagian bab keempat membahas hasil pengujian. Bagian terakhir yaitu kesimpulan dan saran.

2. Studi Terkait

Penelitian sebelumnya telah menunjukkan bahwa MSKD efektif dalam mengidentifikasi permukaan wortel selama proses panen. MSKD juga telah terbukti bekerja dengan baik pada dataset MovieLens-1M, Amazon Game, dan Steam dalam studi lain [11]. Membandingkan metode ini dengan teknik diagnostik lainnya, metode ini dapat menghasilkan rekomendasi model kerja yang lebih akurat dan tepat waktu. Kami juga telah mengkaji pada penelitian lain yang membahas metode MSKD pada industri manufaktur aditif [12]. Dengan begitu, kami dapat melakukan eksplorasi agar metode MSKD dapat diterapkan dalam perangkat tepi di IDS.

TABEL 1
Penelitian terkait tentang implementasi mskd untuk sistem pengendali data jaringan berbasis komputasi tepi dengan unsw-nb15

Ref	MSKD	IDS	DNN	UNSW-NB15	Adam Optimization
[11]	✓	✗	✗	✗	✗
[12]	✓	✗	✗	✗	✗
[13]	✗	✓	✗	✗	✗
[14]	✗	✓	✓	✗	✗
[15]	✗	✓	✓	✓	✗
[16]	✗	✗	✓	✗	✓
Metode yang Diusulkan	✓	✓	✓	✓	✓

Kami juga telah mengkaji berbagai metodologi dalam membangun IDS, sebagaimana dipaparkan dalam penelitian [1] - [8]. Yuliswar *et al*. [13] meneliti sistem deteksi serangan Distributed Denial of Service (DDoS) berbasis pohon keputusan yang memanfaatkan manfaat

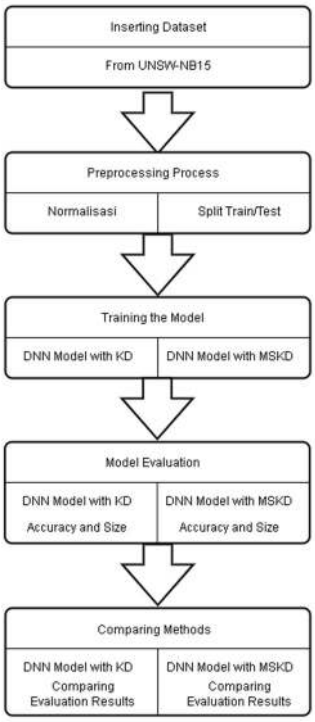
deep learning menggunakan dataset CIC-DDoS2019. Hasilnya menunjukkan bahwa sistem dapat mendeteksi serangan siber sambil menjaga informasi penting tetap aman. Illy *et al*. [14] mengusulkan sistem deteksi DNN *multistage* hibrida untuk meningkatkan akurasi perlindungan terhadap serangan siber pada Industrial Internet of Things (IIoT) dibandingkan dengan IDS tradisional. Model DNN telah terbukti mampu dengan baik dalam proses mendeteksi perilaku tidak biasa dan memulai tanggapan darurat saat serangan siber terjadi.

Dalam penelitian sebelumnya, kami menggunakan dataset UNSW-NB15 [15] untuk menerapkan teknik kuantisasi pada IDS. Model utama yang digunakan dalam studi ini adalah DNN, sementara Pearson correlation coefficient (PCC) digunakan untuk seleksi fitur. Akibatnya, studi ini berhasil menerapkan model DNN dalam IDS untuk pengujian dengan dataset UNSW-NB15. Shweta More *et al*. [16] menguji algoritma Logistic Regression, Support Vector Machine (SVM), Decision Tree, dan Random Forest, lalu membandingkannya.

Terakhir, sejumlah penelitian telah meneliti bagaimana optimizer Adam dan format file H5 memengaruhi aplikasi deep learning. Mageed *et al*. [17] mengevaluasi Adam optimizer dibandingkan dengan berbagai metode lain untuk menilai dampaknya pada deteksi kanker. Studi tersebut juga membahas antarmuka pengguna yang dapat membaca dan menulis data dari file H5. File H5 ini sering digunakan untuk proses menyimpan model *deep learning*, terutama pada saat menggunakan pustka Keras dalam Python. Oleh karena itu, pembahasan mengenai hubungan antara Adam optimizer dan penggunaan file H5, khususnya pada penggunaan *soft label* selama proses pelatihan Student Model, dapat membuka peluang untuk penelitian lebih lanjut bagi kami. TABEL 1 merupakan ringkasan dari penelitian yang telah dibahas sebelumnya.

3. Konsep Baru MSKD pada IDS untuk *Edge* Ringan

Kami telah membuat alur penelitian untuk membantu dalam mencapai tujuan penelitian ini. Dataset IDS yang digunakan berasal dari UNSW-NB15. Langkah berikutnya adalah mempersiapkan data dan melatihnya menggunakan metode distilasi pengetahuan multi-tahap (MSKD). Kemudian kami melakukan perbandingan metode distilasi pengetahuan (KD) dengan metode distilasi pengetahuan multi-tahap (MSKD) dan melakukan evaluasi kinerja dari kedua model tersebut. Pada langkah terakhir, dalam upaya menghasilkan akurasi yang tinggi dengan ukuran model yang rendah, kami melakukan pengembangan metode MSKD untuk model DNN. GAMBAR *1* menunjukkan metodologi penelitian kami dalam bentuk diagram alir.



GAMBAR 1
Metodologi penelitian

3.1 Dataset IDS UNSW-NB15

Dataset UNSW-NB15 IDS dibuat oleh Cyber Range Lab di Australian Centre for Cyber Security (ACCS), yang merupakan bagian dari Universitas New South Wales (UNSW) di Australia [18]. Dataset ini dibuat bertujuan untuk membantu peneliti dalam mempelajari IDS. Perangkat IXIA PerfectStorm dari Laboratorium Cyber Range ACCS digunakan untuk membuat UNSW-NB15, yang memiliki data lalu lintas jaringan modern yang lebih representatif[19]. Perangkat ini dapat menghasilkan lalu lintas jaringan yang realistis untuk menguji jaringan.

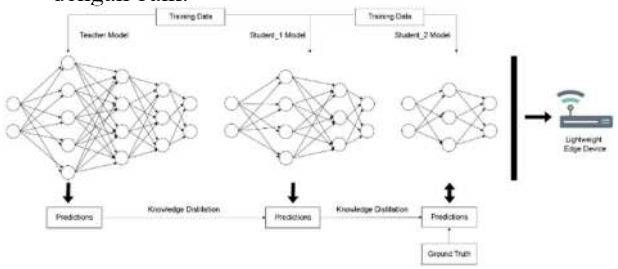
Data set UNSW-NB15 terdiri dari 49 fitur input dan 1 fitur label. Label-label tersebut menunjukkan kelas normal dan berbagai jenis serangan. Jenis serangan yang dapat dilakukan meliputi fuzzer, analisis, backdoor, DoS, exploit, generik, pengintaian, shellcode, dan worm. Data set ini terbagi menjadi dua bagian: sekitar 175.341 data untuk pelatihan dan sekitar 82.332 data untuk pengujian.

3.2 Metode MSKD untuk Model DNN

Jenis jaringan saraf tiruan (JST) yang disebut DNN adalah model yang bekerja seperti jaringan saraf manusia [20]. Arsitektur DNN disebut “dalam” karena terdapat lebih dari satu lapisan tersembunyi antara lapisan input dan output. Lapisan pertama dalam DNN belajar fitur, yang menggantikan cara lama dalam mengekstrak fitur. Lapisan terakhir, di sisi lain, disebut lapisan terhubung sepenuhnya, dan tugasnya adalah membuat keputusan atau klasifikasi.

Distilasi Pengetahuan (KD) adalah metode kompresi model yang bertujuan untuk membuat model *deep learning* lebih kecil sambil tetap mempertahankan kemampuannya untuk membuat prediksi yang baik [21]. KD menggunakan dua model, yaitu Teacher Model dan Student Model. Teacher Model adalah model besar dan kompleks yang sangat akurat. Student model, adalah model yang lebih kecil dan sederhana, tetapi kinerjanya tidak sebaik Teacher model pada awalnya. Teacher Model menciptakan *soft label* dalam proses KD. Label ini

adalah hasil prediksi dalam bentuk probabilitas dari dataset asli yang menggunakan fungsi aktivasi *softmax*. *Soft label* ini memberikan informasi lebih banyak daripada *hard label* (label asli). Student Model biasanya bekerja lebih baik saat dilatih dengan *soft label* daripada saat dilatih dengan *hard label*. Jadi, *student* model menjadi model yang lebih kecil tetapi tetap bekerja dengan baik.



GAMBAR 2
Konsep Metode MSKD Sederhana untuk Arsitektur DNN

MSKD adalah proses multi-tahap yang dibangun di atas distilasi pengetahuan [22]. TeacherModel yang melatih Student Model 1 adalah langkah pertama dalam proses MSKD. Kemudian Student Model 1 melatih Student Model 2. Pendekatan bertahap ini bertujuan agar setiap tahap menghasilkan model kecil yang tetap akurat. DNN yang kami usulkan didasarkan pada arsitektur MSKD yang ditunjukkan pada GAMBAR 2.

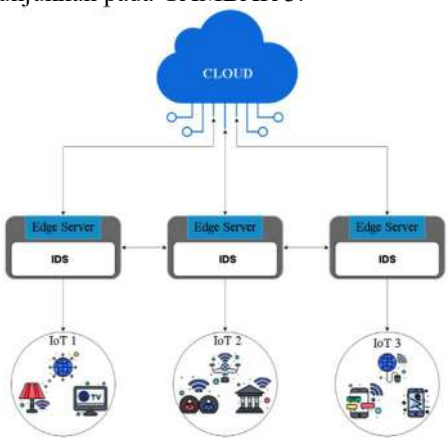
Model DNN sebagai guru adalah arsitektur pertama yang kami buat. Hal pertama yang terjadi di lapisan input Teacher Model adalah pemisahan fitur dan label. Kemudian, fitur kategori dienkode, fitur dibuat sama, dan data dibagi menjadi set pelatihan dan pengujian. Teacher Model ini memiliki tiga lapisan tersembunyi. Lapisan tersembunyi pertama memiliki 512 neuron dan menggunakan aktivasi ReLU. Lapisan tersembunyi kedua juga memiliki lapisan padat dengan 256 neuron dan menggunakan ReLU. ReLU digunakan pada lapisan tersembunyi ketiga, yang memiliki 128 neuron. Aktivasi *softmax* digunakan pada lapisan output, yang merupakan lapisan terhubung sepenuhnya dengan 10 neuron.

Langkah berikutnya adalah membuat Student Model pertama. Lapisan input Student Model pertama sama dengan Teacher Model, dan juga menggunakan DNN. Namun, untuk membuat model lebih kecil, jumlah lapisan tersembunyi dikurangi. Student Model 1 menggunakan dua lapisan tersembunyi. Lapisan pertama menggunakan 128 neuron dan fungsi aktivasi ReLU, sedangkan untuk lapisan kedua menggunakan 64 neuron dan fungsi aktivasi ReLU. Fungsi aktivasi *softmax* digunakan pada lapisan output dengan menggunakan 10 neuron.

Kemudian kami membuat model akhir, yaitu Student Model kedua. Student Model 2 juga menggunakan base mode DNN. Dilakukan sedikit penyesuaian pada Student Model 2 ini, strukturnya lebih sederhana daripada Student Model 1. Lapisan inputnya sama dengan Teacher Model. Model ini hanya memiliki satu lapisan tersembunyi dengan aktivasi ReLU dan lapisan padat dengan 64 neuron. Lapisan outputnya masih menggunakan *softmax* dan memiliki lapisan padat dengan 10 neuron.

3.3 IDS Ringan dalam Penerapan di Tepi Jaringan

Arsitektur jaringan komputer dalam penelitian ini untuk implementasi IDS berbasis UNSW-NB15 merupakan perluasan dari pendekatan terdistribusi berbasis tepi [23]. Pendekatan tepi adalah cara merancang jaringan yang tersebar, sementara konsep jaringan terpusat adalah cara merancang jaringan yang berada di satu tempat. Ada dua manfaat utama dari jaringan tepi. Pertama, beberapa tugas komputasi dipindahkan ke tepi, yang lebih dekat dengan node pemrosesan. Hal ini membuat jaringan menjadi lebih responsif dan sistem merespons lebih cepat [24]. Kedua, ketika komputasi dilakukan di tepi, informasi pribadi yang dapat melanggar privasi tidak perlu dikirim atau dibagikan kepada banyak orang. Terdapat sistem IDS di server tepi yang beroperasi secara mandiri tetapi masih dapat berkomunikasi dengan awan utama. Kami mengusulkan arsitektur IDS berbasis tepi yang ditunjukkan pada GAMBAR 3.



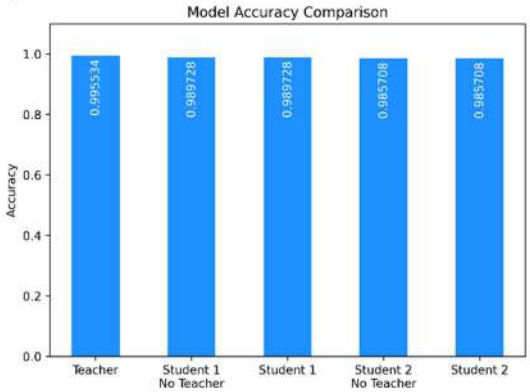
GAMBAR 3
IDS ringan untuk perangkat tepi.

4. Evaluasi

Pada bagian ini akan kami membahas hasil pengujian dan analisis hasil pengujian.Terdapat beberapa hasil pengujian yang kami bahas yaitu perbandingan akurasi setiap model, ukuran setiap model, dan waktu eksekusi setiap model.

4.1 Hasil Pengujian

Kami menggunakan dataset UNSW-NB15, yang memiliki 82.332 data point, untuk pengujian. Dataset ini memiliki 43 fitur yang memberikan informasi tentang jaringan, seperti layanan, sttl, ct dst src ltm, dan ct src dport ltm. Dataset ini memiliki sepuluh label: satu label normal dan sembilan label serangan. Kelas serangan dengan jumlah data point terbanyak adalah Generic, yang memiliki 18.871 data point. Kelas dengan jumlah data point terendah adalah Worm, yang memiliki 44 data point.

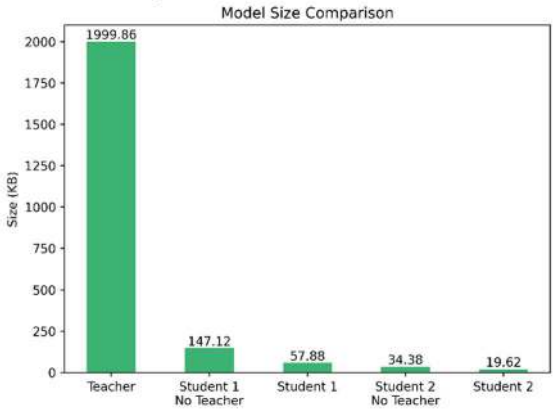


GAMBAR 4
Perbandingan akurasi model pada metode MSKD untuk IDS.

Uji hasil dilakukan terkait presisi dan cakupan metode MSKD baik dengan maupun tanpa KD. Metode

MSKD menghasilkan akurasi 0,9897 pada Student Model 2, yang telah berhasil menyelesaikan tahap distilasi pembelajaran. Hasil ini diperoleh dari fase awal Student Model, yang menggunakan fungsi kerugian komposit yang menggabungkan *KL divergence* dan *categorical-cross entropy* dari Teacher Model dengan *soft label*. Teacher Model, di sisi lain, memiliki kerugian 0,0973 dan akurasi 0,9955. Di sisi lain, Student Model 1 memiliki kerugian 0,3523. Pada tahap Student Model 2, model ini belajar dari Student Model 1, yang belajar dari Teacher Model. Hasil akurasi Student Model 2, yaitu 0,9857 dengan kerugian 0,1917, menunjukkan bahwa model ini belajar dari Student Model 1. Kita juga dapat melihat perbandingan antara Student Model 1 dan Student Model 2. Disajikan

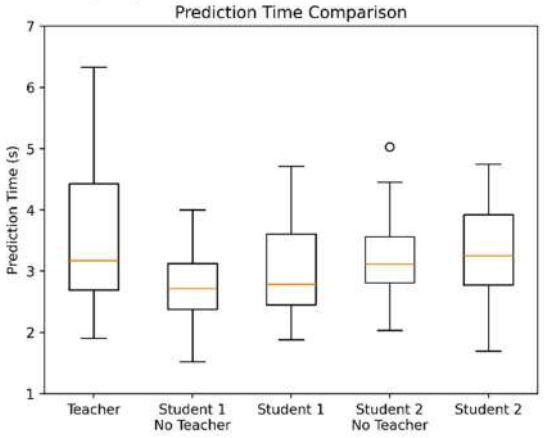
GAMBAR 4 yang menunjukkan seberapa baik setiap model bekerja dalam MSKD.



GAMBAR 5
Perbandingan ukuran model

Kemudian, kami melakukan perbandingan ukuran pada setiap model dan metode. Student Model merupakan model dengan ukuran terbesar, yaitu 1.99 MB, tetapi juga memiliki akurasi dan tingkat loss terbaik. Tingkat losss akurasi pada tahap Student Model 1 adalah 0.58%, yang merupakan yang terbaik kedua. Ukuran model juga berkurang menjadi 57.88 KB, selain kerugian akurasi yang kecil. Artinya, laju kompresi (CR) 34.6 kali lebih cepat daripada Teacher Model dan 2.5 kali lebih cepat daripada Student Model 1 dengan *hard label*. Selain itu, Student Model 2 0.99% kurang akurat daripada Teacher Model dan 0.40% kurang akurat daripada Student Model 1. Ukuran Student Model 2 juga berkurang, menjadi 19.62 KB. Angka-angka ini menunjukkan bahwa CR 101,9 kali lebih tinggi daripada Teacher Model, 3.0 kali lebih tinggi daripada Student Model 1, dan 1.8 kali lebih tinggi daripada student Model 2. Waktu prediksi rata-rata antara Teacher Model dan Student Model signifikan dengan nilai p 0,01 dan 0.03, keduanya di bawah . Hal menarik lainnya adalah waktu prediksi rata-rata untuk Student Model1 tidak jauh berbeda dengan Student Model 2, meskipun kedua model memiliki ukuran yang berbeda.

GAMBAR 5 menunjukkan seberapa besar setiap model yang digunakan dalam MSKD untuk IDS.



GAMBAR 6
Boxplot perbandingan setiap model MSKD dalam waktu prediksi

Terakhir, dilakukan uji hasil berapa lama waktu yang dibutuhkan setiap model untuk melakukan prediksi saat dijalankan pada simulasi Raspberry Pi untuk melihat seberapa baik kinerjanya pada tepi IDS. Kami menyajikan GAMBAR *6* untuk menunjukkan perbandingan waktu eksekusi setiap model. Student Model memiliki rata-rata dan simpangan baku waktu prediksi tertinggi, yaitu 0,16 dan 0,08. Waktu rata-rata yang dibutuhkan oleh Student Model 1 dan Student Model 2 untuk melakukan prediksi adalah 0,12 dan 0,13, masing-masing. Fakta menarik lainnya adalah bahwa rata-rata waktu prediksi untuk Student Model 1 tidak berbeda secara signifikan dengan Student Model 2, meskipun ukuran kedua model tersebut berbeda.

4.2 Analisis Hasil Pengujian

Mandal *et al*. [25] menyatakan bahwa pada Student Model memerlukan jumlah neuron yang lebih sedikit karena belajar dari *soft label* yang dihasilkan oleh Teacher Model, yang mengandung informasi yang lebih komprehensif dibandingkan *hard label*. Temuan studi ini menunjukkan bahwa Student Model dapat mencapai ukuran model yang lebih kecil tanpa mengubah jumlah parameter, dengan pelatihan menggunakan *soft label* menghasilkan ukuran file yang lebih efisien. Studi ini juga meneliti bagaimana optimizer Adam memengaruhi ukuran file H5, yang belum diteliti dalam studi lain. Studi ini juga menambah pemahaman kita tentang bagaimana optimizer ini memengaruhi efisiensi ukuran model dalam MSKD.

Studi ini juga memperkenalkan konsep MSKD yang sederhana, di mana Student Model 2 mencapai kerugian akurasi maksimal dan rasio kompresi 101,9× dibandingkan dengan Teacher Model dan 3,0× dibandingkan dengan Student Model 1. Penelitian sebelumnya telah menggunakan MSKD dalam sistem rekomendasi [11] dan pembelajaran incremental dalam manufaktur [12]. Studi ini menerapkan MSKD sebagai teknik kompresi model IDS dalam arsitektur komputasi tepi, berdasarkan penelitian sebelumnya yang menerapkan metode kuantisasi pada dataset UNSW-NB15 [15]. Selain itu, meskipun berbagai studi, termasuk [7] dan [9], telah mengembangkan konsep MSKD, tidak ada yang menerapkan pendekatan MSKD yang sederhana yang diusulkan dalam studi ini. Hasil pengujian juga menunjukkan bahwa Student Model 2 memiliki ukuran dan rasio kompresi yang lebih baik daripada Student Model 1.

5. Kesimpulan

Tujuan penelitian ini adalah implementasi konsep *Multi-stage Knowledge Distillation* (MSKD) yang sederhana dan mudah digunakan untuk membantu mengompres model dalam IDS berbasis komputasi tepi. Kami menggunakan dataset UNSW-NB15 untuk membangun model DNN yang dapat mengklasifikasikan serangan. Skema MSKD yang diusulkan mencakup tiga model, yaitu Teacher Model, Student Model 1, dan Student Model 2. Teacher Model mencapai akurasi 0.9955, sementara Student Model 1 dan Student Model 2 mencapai akurasi 0.9897 dan 0.9857, masing-masing. Kerugian akurasi untuk Student Model 1 adalah 0,58%, dan untuk Student Model 2, kerugiannya adalah 0,99% dibandingkan dengan Teacher Model dan 0,40% dibandingkan dengan Student Model 1. Dalam hal efisiensi model, Student Model 1 memiliki tingkat kompresi (CR) terbaik sebesar 101,9× dibandingkan dengan Teacher Model dan 3,0× dibandingkan dengan Student Model 2. Selain itu, penggunaan *soft label* terbukti meningkatkan CR sebesar 1,8 kali dibandingkan dengan melatih Student Model 2 menggunakan *hard label*.

Penelitian selanjutnya dapat dikembangkan untuk melakukan perbandingan langsung antara MSKD dengan konsep MSKD lainnya, seperti konsep born-again networks (BAN), self-distillation, and progressive knowledge distillation.

Daftar Pustaka

[1] R. Devi, A. Kumar, V. Kumar, A. Saini, A. Kumari, and V. Kumar, "A Review Paper on IDS in Edge Computing or EoT," in *2022 International Conference on 4th Industrial Revolution Based Technology and Practices, ICFIRTP 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 30–35. doi: 10.1109/ICFIRTP56122.2022.10059442.

[2] W. AlAqqad, M. Nijim, U. Onyeakazi, and H. Albataineh, "Cyber Edge: Mitigating Cyber-Attacks in Edge Computing Using Intrusion Detection System," 2024, pp. 292–305. doi: 10.1007/978-3-031-56950-0_25.

[3] T.-M. Hoang, T.-A. Pham, V.-V. Do, V.-N. Nguyen, and M.-H. Nguyen, "A Lightweight DNN-based IDS for Detecting IoT Cyberattacks in Edge Computing," in *2022 International Conference on Advanced Technologies for Communications (ATC)*, IEEE, Oct. 2022, pp. 136–140. doi: 10.1109/ATC55345.2022.9943049.

[4] R. K. Vigneswaran, R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Evaluating Shallow and Deep Neural Networks for Network Intrusion Detection Systems in Cyber Security," in *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, IEEE, Jul. 2018, pp. 1–6. doi: 10.1109/ICCCNT.2018.8494096.

[5] N. N. Tai, H. Do Hoang, P. T. Duy, and V. H. Pham, "An interpretable approach for trustworthy intrusion detection systems against evasion samples," *CTU Journal of Innovation and Sustainable Development*, vol. 15, no. Special Issue (ISDS), pp. 12–19, Oct. 2023, doi: 10.22144/ctujoisd.2023.030.

[6] R. Mishra and H. P. Gupta, "Designing and Training of Lightweight Neural Networks on Edge Devices Using Early Halting in Knowledge Distillation," *IEEE Trans Mob Comput*, vol. 23, no. 5, pp. 4665–4677, May 2024, doi: 10.1109/TMC.2023.3297026.

[7] C. Xu *et al.*, "Multiple-Stage Knowledge Distillation," *Applied Sciences*, vol. 12, no. 19, p. 9453, Sep. 2022, doi: 10.3390/app12199453.

[8] Z. Hao, J. Guo, K. Han, H. Hu, C. Xu, and Y. Wang, "VanillaKD: Revisit the Power of Vanilla Knowledge Distillation from Small Scale to Large Scale," May 2023, [Online]. Available: http://arxiv.org/abs/2305.15781

[9] X. Zeng *et al.*, "DSP-KD: Dual-Stage Progressive Knowledge Distillation for Skin Disease Classification," *Bioengineering*, vol. 11, no. 1, p. 70, Jan. 2024, doi: 10.3390/bioengineering11010070.

[10] W. Zhou, C. Song, K. Song, N. Wen, X. Sun, and P. Gao, "Surface Defect Detection System for Carrot Combine Harvest Based on Multi-Stage Knowledge Distillation," *Foods*, vol. 12, no. 4, p. 793, Feb. 2023, doi: 10.3390/foods12040793.

[11] Y. Du, J. Niu, Y. Wang, and X. Jin, "Multi-stage knowledge distillation for sequential recommendation with interest knowledge," *Inf Sci (N Y)*, vol. 654, p. 119841, Jan. 2024, doi: 10.1016/j.ins.2023.119841.

[12] Z. Shi, Y. Li, and C. Liu, "Knowledge Distillation-enabled Multi-stage Incremental Learning for Online Process Monitoring in Advanced Manufacturing," in *2022 IEEE International Conference on Data Mining Workshops (ICDMW)*, IEEE, Nov. 2022, pp. 860–867. doi: 10.1109/ICDMW58026.2022.00154.

[13] T. Yuliswar, I. Elfitri, and O. W purbo, "Optimization of Intrusion Detection System with Machine Learning for Detecting Distributed Attacks on Server," *INOVTEK Polbeng - Seri Informatika*, vol. 10, no. 1, pp. 367–376, Feb. 2025, doi: 10.35314/vem9da98.

[14] P. Illy, G. Kaddoum, P. F. de Araujo-Filho, K. Kaur, and S. Garg, "Retracted: A Hybrid Multistage DNN-Based Collaborative IDPS for High-Risk Smart Factory Networks," *IEEE Transactions on Network and Service Management*, vol. 19, no. 4, pp. 4273–4283, Dec. 2022, doi: 10.1109/TNSM.2022.3202801.

[15] A. G. Putrada, N. Alamsyah, M. N. Fauzan, and I. D. Oktaviani, "Pearson Correlation for Efficient Network Anomaly Detection with Quantization on the UNSW-NB15 Dataset," in *2024 International Conference on ICT for Smart Society (ICISS)*, IEEE, Sep. 2024, pp. 1–6. doi: 10.1109/ICISS62896.2024.10751550.

[16] S. More, M. Idrissi, H. Mahmoud, and A. T. Asyhari, "Enhanced Intrusion Detection Systems Performance with UNSW-NB15 Data Analysis," *Algorithms*, vol. 17, no. 2, p. 64, Feb. 2024, doi: 10.3390/a17020064.

[17] A. Mageed, A. EL-SAYED, M. A. Gaber, and T. Abdelhamid, "A Comprehensive Approach to Skin Cancer Detection: Mixed Dataset, Optimizer Enhancement, and Web Integration," *Menoufia Journal of Electronic Engineering Research*, vol. 34, no. 1, pp. 26–39, Jan. 2025, doi: 10.21608/mjeer.2025.316087.1093.

[18] S. Moualla, K. Khorzom, and A. Jafar, "Improving the Performance of Machine Learning-Based Network Intrusion Detection Systems on the UNSW-NB15 Dataset," *Comput Intell Neurosci*, vol. 2021, no. 1, Jan. 2021, doi: 10.1155/2021/5557577.

[19] A. Gupta and R. Simon, "Enhancing Security in Cloud Computing With Anomaly Detection Using Random Forest," in *2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, IEEE, Mar. 2024, pp. 1–6. doi: 10.1109/ICRITO61523.2024.10522227.

[20] I. Tavares *et al.*, "Comparison of PV Power Generation

Forecasting in a Residential Building using ANN and DNN," *IFAC-PapersOnLine*, vol. 55, no. 9, pp. 291–296, 2022, doi: 10.1016/j.ifacol.2022.07.051.

[21] M. N. Adiputra, A. G. Putrada, and I. D. Oktaviani, "DistilQ-NILM: A Hybrid Quantization-Knowledge Distillation LSTM Model for Edge-Based Load Monitoring," in *2024 International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA)*, IEEE, Dec. 2024, pp. 284–289. doi: 10.1109/ICICYTA64807.2024.10913187.

[22] B. Fan, H. Ma, Y. Liu, X. Yuan, and W. Ke, "KDTM: Multi-Stage Knowledge Distillation Transfer Model for Long-Tailed DGA Detection," *Mathematics*, vol. 12, no. 5, p. 626, Feb. 2024, doi: 10.3390/math12050626.

[23] A. Gautama, P. #1, N. Alamsyah, M. N. Fauzan, S. Prabowo, and I. D. Oktaviani, "QUIDS: A Novel Edge-Based Botnet Detection with Quantization for IoT Device Pairing", doi: 10.34818/indojc.2023.8.3.878.

[24] A. G. Putrada and D. A. S. Ilhami, "G-Mean for Optimum Threshold in Anomaly Detection with Autoencoder: Cyber Security on The RT-IOT2022 Dataset," in *2024 IEEE 22nd Student Conference on Research and Development (SCOReD)*, IEEE, Dec. 2024, pp. 433–438. doi: 10.1109/SCOReD64708.2024.10872727.

[25] S. Mandal, X. Lin, and R. Srikant, "A Theoretical Analysis of Soft-Label vs Hard-Label Training in Neural Networks," Dec. 2024, [Online]. Available: http://arxiv.org/abs/2412.09579