ANALISA DATASET SOFTWARE DEFINED NETWORK INTRUSION MENGGUNAKAN ALGORITMA DEEP LEARNING H2O

Rizal[1], Martanto[2], Yudhistira Arie Wijaya[3]
[1)] Program Studi Teknik Informatika, STMIK IKMI Cirebon
[2)] Program Studi manajemen informatika, STMIK IKMI Cirebon
[3)] Program Studi Sistem Informasi, STMIK IKMI Cirebon
rizal120398@gmail.com

ABSTRAK

Software-defined networking Intrusion (SDNI) baru-baru ini menjadi salah satu solusi paling menjanjikan untuk Internet masa depan. Dengan sentralisasi logis dari pengontrol dan tampilan jaringan global, SDN *Intrusion* menawarkan peluang untuk meningkatkan keamanan jaringan. Pada penelitian sebelumnya oleh Omar Jamal Ibrahim, dan Wesam S. Bhaya menjelaskan tentang dataset SDN *intrusion* bahwa dengan menggunakan algoritma *Support Vector Machine* (SVM) diperoleh dengan nilai akurasi sebesar 97.77%, sehingga menurut peneliti ini masih bisa untuk di kaji lagi dengan menggunakan algortima yang berbeda. Sebagai proses pencarian informasi dari sekumpulan data yang akan dijadikan pengetahuan baru dapat dimanfaatkan maka dari itu data mining juga seringkali dikenal dengan sebutan *Knowledge Discovery in Database* (KDD). Metode klasifikasi yang digunakan yaitu *Deep Learning* H2O yaitu suatu metode menggunakan algoritma multilayer yang di sebut *neural networks.* Tujuan dari algoritma ini mencoba untuk mengambil suatu kesimpulan berdasarkan struktur logika yang di berikan secara berkelanjutan. Peneliti menggunakan software aplikasi Rapid Miner sebagai bantuan dalam menganalisis dataset. Dari hasil penelitian terbukti bahwa algoritma *Deep Learning* H2O yang digunakan lebih baik. Hal ini dibuktikan dengan hasil evaluasi penelitian bahwa algoritma *Deep Learning* H2O mampu menganalisa nilai recall 100.00% dan tingkat akurasi sebesar 99.66% sehingga model klasifikasi menggunakan algoritma *Deep Learning* H2O lebih baik saat diterapkan pada dataset yang digunakan.

Kata kunci: *Machine Learning, Deep Learning* H2O, *Software defined network Intrusion*

1. PENDAHULUAN

Software defined network intrusion (SDNI) merupakan paradigma baru dalam manajemen jaringan yang memberikan fasilitas untuk melakukan konfigurasi, virtualisasi, dan mengelola infrastruktur jaringan secara terpusat. Manajemen jaringan secara terpusat dilakukan pada SDNI Controller yang dimana memisahkan network data plane dari control functions [1]. *Software defined network intrusion* (SDNI) mengelola dan mengkonfigurasi jaringan melalui abstraksi tingkat tinggi. Sifat SDNI yang dinamis dan dapat diprogram menghadapi banyak masalah keamanan yang menuntut solusi keamanan yang inovatif [2].

Software defined network intrusion (SDNI) menawarkan potensi untuk secara efektif mendeteksi dan memantau masalah keamanan jaringan yang timbul dari fitur-fitur baru yang dapat diprogram. Namun, terlepas dari fitur jaringan ini,
ada beberapa kelemahan dari ancaman online. Penjahat dunia maya menyuntikkan lalu lintas berbahaya ke SDNI dan mencuri informasi sensitif dari mereka. Serangan jaringan pada SDNI dapat dideteksi menggunakan pemantauan lalu lintas. Data yang Anda pilih mencakup log lalu lintas waktu nyata yang dikumpulkan setiap hari. Data pertama milik file capture paket atau PCAP dan kemudian dikonversi ke file tabular [3].

Arsitektur SDN yang khas terdiri dari lapisan infrastruktur sebagai lapisan kontrol dan lapisan aplikasi. SDN menggunakan berbagai antarmuka, seperti antarmuka Northbound dan Southbound, untuk komunikasi antar pesawat. Antarmuka Northbound (NBI) digunakan antara lapisan aplikasi dan lapisan kontrol, sedangkan Antarmuka Southbound (SBI) memungkinkan lapisan kontrol untuk berkomunikasi dengan lapisan data [2].

Penelitian terdahulu dengan judul "*Intrusion Detection in Software defined network intrusion Using Deep Learning Approach* "menjelaskan bahwa SDNI menawarkan pengumpulan data implisit, kemampuan beradaptasi, programabilitas, dan tampilan sistem di seluruh dunia. Oleh karena itu, dipandang sebagai pilihan ideal untuk mengatur informasi bermacam-macam dan penyelidikan. Sedangkan penulis menganalisis dataset mengenai keakuratan SDNI dalam mengangani serangan cyber [4].

Implementasi untuk jaringan SDNI sudah pernah di bahas dalam penelitian sebelumnya, diantaranya adalah penelitian yang dilakukan oleh Huseyin Polat, Onur Polat, dan Aydin Cetin yang berjudul "*Detecting DDoS Attacks in Software-Defined Networks Through Feature Selection Methods and Machine Learning Models*" menjelaskan bahwa Jaringan yang ditentukan perangkat lunak (SDNI) memiliki beberapa keunggulan, termasuk pengelolaan, penskalaan, dan peningkatan kinerja. Fitur spesifik diperoleh dari SDNI untuk dataset dalam kondisi normal dan di bawah lalu lintas serangan DDoS. Kemudian dataset baru dibuat menggunakan metode seleksi fitur dari

dataset yang ada. Dataset dibuat, dilatih, dan diuji menggunakan model klasifikasi Support Vector Machine (SVM), Naive Bayes (NB), Artificial Neural Network (ANN), dan K-nearest Neighbor (KNN). Hasil pengujian menunjukkan bahwa penggunaan wrapping feature selection dengan classifier KNN mencapai akurasi tertinggi (98,3%) dalam mendeteksi serangan DdoS [5]. Akar permasalahan dalam penelitian ini yaitu, belum adanya Implementasi Algoritma *deep learning* H2O Pada Analisa Dataset *Software defined network intrusion (*SDNI*).* Untuk itu diperlukan pengembangan terhadap penelitian terdahulu dengan membuat model proses baru dan mengimplementasikan algoritma SDNI pada model proses tersebut. Harapan dari penelitian ini yaitu diperolehnya suatu model proses dengan nilai *recall* dan tingkat akurasi lebih baik untuk mengklasifikasikan data lebih besar lagi.

Pemasalahan pada serangan cyber tersebut menjadi perhatian besar karena meningkatnya masalah keamanan di jaringan saat ini. Berdasarkan permasalahan tersebut berbagai macam algoritma telah diusulkan yang dapat mendeteksi. Untuk mengatasi masalah tersebut dibutuhkan suatu tools yang mampu mendeteksi lebih awal terjadinya *indruder* atau kegiatan yang merugikan suatu jaringan. Dalam hal ini *deep learning* merupakan suatu solusi yang sangat tepat untuk keperluan tersebut. Pendekatan yang digunakan untuk menganalisis serangan *cyber* menggunakan dataset SDN *intrusion* adalah algoritma *deep learning*.

Tujuan dari penelitian yang peneliti buat ini yaitu, mengetahui tingkat akurasi dan menentukan nilai akurasi tertinggi serta membuat model proses terhadap klasifikasi dataset *Software defined network intrusion* (SDNI) dengan menggunakan algoritma *deep learning* H2O. Algoritma ini memungkinkan dapat memprediksi akurasi yang tinggi. *Deep learning* H2O sendiri didasarkan pada jaringan saraf tiruan *feed forward multi-layer* yang dilatih dengan penurunan gradien stokastik menggunakan *backpropagation***.**

Berdasarkan pada pemaparan masalah tersebut, maka judul penelitian yang di usulkan adalah "Analisa Dataset Software defined network intrusion Intrusion Menggunakan Algoritma Deep Learning H2O**".** Adapun yang menjadi alasan dilakukannya penelitian dengan judul tersebut adalah untuk mendapatkan tingkat akurasi yang di peroleh terhadap model yang dibuat sebagai sistem untuk menganalisa datasaet serangan cyber yang ada dalam *Software defined network intrusion* (SDNI) dengan menggunakan metode *deep learning*. Data yang dipilih berisi catatan lalu lintas real-time yang telah ditangkap setiap hari. Oleh karena itu semua orang dapat melihat dan mengetahui serangan apa saja yang ada di dunia maya setiap harinya.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Pada penelitian dengan judul A Survey on Intrusion Detection System for Software defined network intrusions (SDNI), tujuan dari penelitian ini adalah *Software defined network intrusion* (SDNI) suatu bentuk sistem keamanan yang terbaru yang dapat di program dan terpusat guna menghadapi serangan cyber dan keamanan yang menuntut mekanisme keamanan inovatif seperti sistem deteksi intrusion (IDS) [2].

Pada penelitian dengan judul "Voting-based intrusion detection framework for securing software-defined networks" mengulas ancaman yang terjadi ketika di dunia internet seperti serangan *Dis- tributed denial of service* (DDoS) yang di anggap sebagai serangan cyber destruktif utama yang membuat layanan controller tidak tersedia untuk pengguna yang yang sah. DDoS berusaha untuk menghabiskan sumber daya sistem dengan membanjirnya lalu lintas di server atau layanan yang ditargetkan. Banjir lalu lintas ini memaksa penurunan kinerja sistem, yang membuat layanan tidak tersedia untuk pengguna yang sah. Serangan DDoS menargetkan sumber daya pengontrol dan saluran komunikasi antara pengontrol dan sakelar. Ancaman seperti DDoS pada pengontrol SDNI dapat melumpuhkan seluruh jaringan. Oleh karena itu, perlu dicari mekanisme pendeteksian serangan DDoS yang efisien dan akurat di *software defined network intrusion* (SDNI) [6].

Pada penelitian dengan judul "Survey on SDNI based network intrusion detection system using machine learning approaches" mengulas tentang mendeteksi adanya serangan mencurigakan yang melewati jaringan diperlukan bantuan dari *Network intrusion detection systems* (NIDS). *Network intrusion detection systems* (NIDS) dirancang untuk mendeteksi aktivitas berbahaya termasuk virus, worm, serangan DDoS. Faktor penentu keberhasilan untuk NIDS adalah kecepatan, akurasi, dan keandalan deteksi abnormalitas. Teknik machine Learning (ML) diterapkan untuk mengembangkan NIDS guna meningkatkan akurasi deteksi dan tingkat alarm palsu yang rendah. Algoritma C4.5 merupakan salah satu algoritma yang digunakan guna membuat pohon keputusan. Metode pohon keputusan mengganti fakta yang sangat besar menjadi pohon keputusan yang merepresentasikan ketentuan / aturan. Ketentuan / aturan bisa dengan mudah dipahami menggunakan bahasa alami. Dengan mengklasifikasi data log IDS dengan algoritma C4.5 bisa mengurangi terbentuknya kesalahan IDS dalam memastikan kegiatan yang termasuk serangan ataupun bukan. Hasil penelitian menampilkan data log IDS bisa diklasifikasikan dengan algoritma C4.5 dengan tingkat akurasi model yaitu 96.371% yang membuktikan kalau model ini bisa digunakan dalam memastikan kegiatan yang termasuk serangan ataupun bukan [3].

Pada penelitian dengan judul "Intrusion Detection in Software defined network intrusion Using Deep Learning Approach" yang mengulas tentang Teknologi Software-Defined Networking (SDNI), yang semakin berkembang dan digunakan secara luas, juga merupakan salah satu metode untuk menghadapi serangan yang terjadi. Software-Defined Networking (SDNI) secara keseluruhan dengan packet sampling dan berbasis pembelajaran mendalam telah mendapatkan banyak pertimbangan dari para ilmuwan. Kerangka kerja SDNI terdiri dari tiga area terpisah diantaranya yaitu bidang kontrol data, bidang kontrol dan bidang aplikasi. Convolutional Neural Network memberikan akurasi yang lebih baik untuk klasifikasi multiclass dalam dataset IDS-2018, Peningkatan zaman menyebabkan sedikit penurunan untuk akurasi BoT-IoT selama pengujian dengan 128 dan 256 batch. Sebaliknya, dalam percobaan dengan 512 batch ada sedikit peningkatan dalam ketepatan. Untuk kumpulan data IDS-2018, peningkatan zaman dan ukuran batch tidak berpengaruh signifikan [4].

2.2. Rapid Miner

Rapid Miner ialah aplikasi data mining berbasis open-source yang terkemuka dan ternama. Didalamnya terdapat aplikasi yang berdiri sendiri untuk analisis data dan sebagai mesin data mining seperti untuk loading data, transformasi data, pemodelan data, dan metode visualisasi data. Aplikasi ini digunakan buat aplikasi bisnis serta komersial dan buat riset, pembelajaran, pelatihan, pembuatan *prototype* dengan kilat, serta pengembangan aplikasi dan menunjang seluruh langkah proses pendidikan mesin tercantum persiapan data, visualisasi hasil, validasi serta pengoptimalan. RapidMiner dibesarkan dengan model *open core* [7].

2.3. Data Mining

Data mining memiliki pengertian lain yaitu *knowledge discovery* ataupun *pattern* recognition merupakan suatu istilah yang digunakan untuk mendapatkan pengetahuan yang tersembunyi dari kumpulan data aturan sangat besar. Tujuan utama data mining adalah untuk menemukan, menggali, atau menambang pengetahuan dari data atau informasi yang kita miliki.

proses data mining adalah menemukan pola yang bermakna dari pola dengan menggunakan teknik statistik dan matematika untuk mengurutkan sejumlah besar data yang disimpan dalam repositori, yang menggunakan teknologi penalaran pola serta teknik-teknik statistik dan matematika.

Menurut Larose, data mining dibagi menjadi beberapa kelompok berdasarkan tugas tugas yang dapat dilakukan, yaitu:

1. Deskripsi
   Untuk mencari metode yang menggambarkan pola serta kecenderungan yang ada dalam informasi. Deskripsi dari pola serta kecenderungan yang kerap membagikan mungkin uraian buat sesuatu pola ataupun kecenderungan.
2. Estimasi
   Ditaksir nyaris sama dengan klasifikasi, kecuali variabel sasaran ditaksir lebih kearah numeric daripada kearah jenis. Model dibentuk memakai record lengkap yang sediakan nilai dari variabel sasaran selaku nilai prediksi.
3. Prediksi
   Prediksi hampir sama dengan klasifikasi dan estimasi, kecuali bahwa dalam prediksi nilai dari hasil akan ada di masa mendatang.
4. Klasifikasi
   Dalam klasifikasi, terdapat target variabel kategori sebagai contoh, penggolongan pendapatan dapat dipisahkan dalam tiga kategori, yaitu pendapatan tinggi, sedang dan rendah.
5. Pengklusteran
   Pengklusteran ialah pengelompokkan record, pengamatan ataupun mencermati serta membentuk kelas objek- objek yang mempunyai kemiripan. Kluster merupakan kumpulan record yang mempunyai kemiripan satu dengan yang yang lain serta mempunyai ketidakmiripan dengan record dalam kluster lain. Algoritma pengklusteran berupaya buat melaksanakan pembagian terhadap totalitas informasi jadi kelompok- kelompok yang mempunyai kemiripan (homogen), yang mana kemiripan record dalam satu kelompok hendak bernilai optimal, sebaliknya kemiripan dengan record dalam kelompok lain hendak bernilai minimun.
6. Asosiasi
   Tugas asosiasi dalam data mining merupakan menciptakan atribut yang timbul dalam sesuatu waktu. Dalam dunia bisnis lebih universal diucap analisis keranjang belanja [8].

2.4. Algoritma H2O

Deep learning H20 adalah framework machine learning open source dengan implementasi teruji penuh dari beberapa algoritma Machine learning yang diterima secara luas. Kerangka kerja DLH2O yang diusulkan digunakan untuk mengoptimalkan masalah klasifikasi multikelas untuk unit perawatan intensif. H2O terdiri dari tiga lapisan. Lapisan pertama adalah lapisan pra-pemrosesan yang terutama bertanggung jawab untuk integrasi data dan pembersihan data. Pada lapisan ini, dataset juga dibagi menjadi subset pelatihan, validasi dan pengujian. Lapisan kedua adalah lapisan pemilihan fitur dimana usulan ACP-WOA (*Whale Optimization Algorithm*) digunakan untuk memilih fitur terbaik yang akan digunakan pada lapisan ketiga yaitu lapisan *Deep Learning*.

Lapisan DL menggunakan fitur pilihan terbaik bersama dengan konfigurasi jaringan saraf terbaik untuk melatih jaringan saraf. Pada sub bagian berikutnya, lapisan kerangka DLH2O yang diusulkan akan dijelaskan secara rinci. Kerangka kerja DLH2O bertujuan untuk menemukan subset fitur yang optimal

dan meminimalkan kesalahan klasifikasi melalui varian baru yang diusulkan dari Whale Optimization Algorithm (WOA) yang disebut ACP-WOA [9].

2.5. Klasifikasi

Klasifikasi merupakan proses untuk menemukan suatu kelas data suatu objek yang belum diketahui berdasarkan data sebelumnya, klasifikasi termasuk kedalam metode pembelajaran atau *supervised* karena membutuhkan pembelajaran data sebelumnya untuk menentukan hasil dari data baru [10].

Klasifikasi memiliki 4 komponen dasar yaitu:

1. *Class*, merupakan variabel yang menjadi label atau hasil suatu objek.
2. *Predictor*, merupakan variabel yang menjadi atribut dari data yang akan digunakan pada klasifikasi.
3. *Training dataset*, merupakan data yang telah memiliki label sebelumnya.
4. *Testing dataset*, merupakan data baru yang akan dilakukan proses klasifikasi.

3. METODOLOGI PENELITIAN

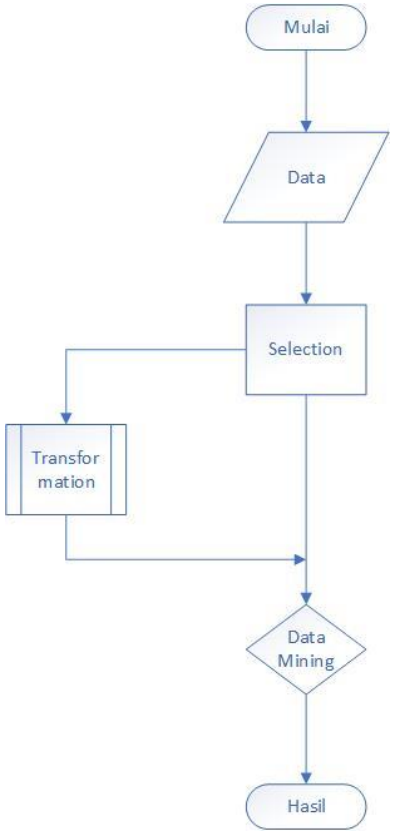
3.1. Metode Pengumpulan Data

Pada penelitian ini penulis menggunakan metode penelitian yang di gunakan adalah eksperimen. Penelitian eksperimen adalah suatu penelitian yang bertujuan untuk menguji pengaruh suatu variabel terhadap variabel lain dan menjelaskan hal hal yang terjadi melalui upaya maipulasi atau pengontrolan varibel – variabel tersebut atau hubungan diantara mereka agar ditemukan hubungan, pengaruh atau perbedaan salah satu variabel atau lebih. Metode penelitian eksperimen jelas berbeda dengan metode penelitian lainnya. Penelitian eksperimen dilakukan dengan membandingkan satu atau lebih kelompok eksperimen yang diberi perlakuan dengan satu atau lebih kelompok pembanding yang tidak diberi perlakuan.

Pengumpulan data informasi dilakukan dengan cara membaca referensi dari buku maupun internet terkait serangan *cyber*, dataset SDNI (*Software defined network intrusion*) dan deep learning. Observasi dilakukan dengan mengakses sumber dari https://archive.ics.uci.edu/ yang menyediakan dataset SDNI (*Software defined network intrusion).*

3.2. Metode Analisis Data

Pada metode analisis data ini penulis melakukan pencarian data mengenai serangan cyber, dataset SDNI (*Software defined network intrusion*). Observasi dilakukan dengan mengakses sumber dari https://www.kaggle.com/datasets/subhajournal/sdn-intrusion-detection sehiingga penulis mendapatkan dataset SDNI tersebut. Penulis membuat tahapan penelitian ini dengan menggunakan flowchart, yang berfungsi sebagai untuk memberikan sebuah gambaran alur pengerjaan atau proses. Proses digambarkan melalui bagan-bagan atau simbol agar informasi yang disajikan lebih mudah dipahami.



Gambar 1. Flowchat Proses algoritma *deep learning* H2O

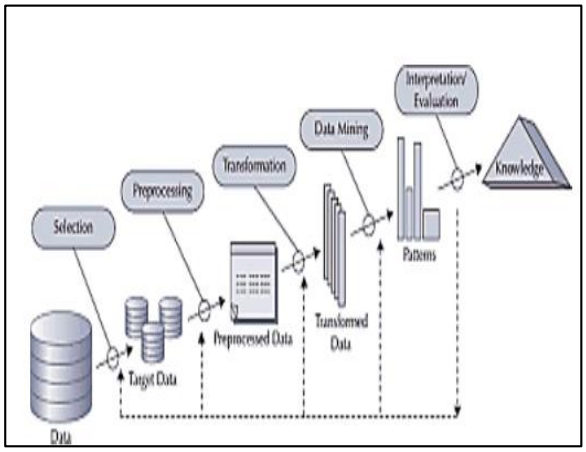
Dalam pengujian data set yang pertama yaitu melakukan proses pencarian data di website www.kaggle.com. Kemudian berlanjut ketahap selection yang berfungsi sebagai pemisah antara data yang ingin di proses ke *data mining*. Selanjutnya masuk ke tahap transformation yang berfungsi untuk mentransformasi bentuk data yang belum memiliki entitas yang lebih jelas ke dalam bentuk data yang *valid* atau siap untuk dilakukan proses *Data Mining.* Pada pengujian validasi menggunakan split validation dengan ratio data training 80% dan ratio data testing 20%, proses dilakukan pengujian split validation menggunakan RapidMiner versi 9.9 dengan melakukan pemodelan algoritma deep learning H2O. Selanjutnya memasukan apply model dan performance, validation merupakan proses untuk mengevaluasi keakuratan prediksi dari model Setelah itu pada proses ini menghasilkan confusion matrix dan akurasi dari model.

3.3. Tahapan Penelitian

Tahapan penelitian yang penulis gunakan dalam penelitian Analisa Dataset *Software defined network intrusion* Menggunakan Algoritma Deep Learning H2O dimana didalamnya terdapat proses data mining dan proses data mining yang digunakan yaitu klasifikasi Algoritma H2O. H2O adalah perangkat lunak untuk pembelajaran mesin dan analisis data,

yang dapat mengerjakan proses seperti *tree, linier model, unsupervised lierning.*

Tujuan dikembangkannya H2O adalah untuk memudahkan user dalam penggunaan seperti, melakukan penskalaan pada big data, pengkodean yang terdokumentasi dengan baik untuk mendukung proses komersil, berjalan pada sistem pihak ketiga, dan memiliki *support programming language* yang luas [11].



Gambar 2. Proses Knowledge Discovery in Database [11]

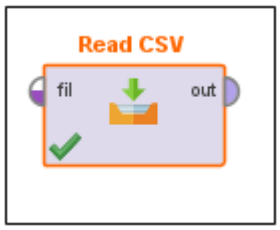
Berdasarkan gambar 2 berikut tahapan – tahapan yang akan dilakukan dalam penelitian ini:

1. Data *Selection*
   Peneliti melakukan Seleksi data dari sekumpulan data operasional dilakukan saat sebelum sesi penggalian data dalam KDD diawali. Data hasil seleksi hendak digunakan untuk proses Data mining ditaruh dalam suatu berkas, terpisah dari basis data operasional.
2. *Preprocessing / Cleaning*
   Peneliti melakukan Proses pembersihan mencakup antara lain membuang duplikasi data, memeriksa data yang inkosisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak (tipografi).
3. *Trasformation*
   Pada fase ini peneliti melakukan transformasi bentuk data yang belum memiliki entitas yang lebih jelas ke dalam bentuk data yang *valid* atau siap untuk dilakukan proses *Data Mining.*
4. Data *Mining*
   Peneliti melakukan proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode, atau algoritma dalam Data mining sangat bervariasi.
5. *Interpratation / evaluation*
   Pada fase terakhir peneliti melakukan proses pembentukan keluaran yang bersumber pada proses Data Mining pola informasi.

4. HASIL DAN PEMBAHASAN

4.1. Data Selection

Pada tahapan yang pertama dilakukan data selection menggunakan operator Read CSV yang berfungsi untuk memanggil data yang akan digunakan dengan berekstensi csv seperti tampak pada gambar 3.



Gambar 3. Operator Read CSV

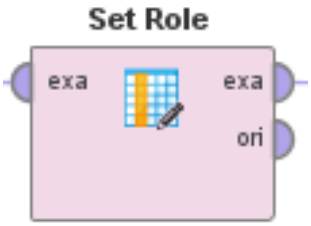
Setelah memilih operator yang akan digunakan lakukan Import Configuration Wizard untuk memanggil data yang akan di proses pada operator Read CSV seperti tampak pada gambar tabel 1

Tabel 1. Parameter Read CSV

Parameter	**Nilai**
csv file	SDN_Intrusion.csv
column separator	,
trim lines	false (default)
use quotes	true (default)
starting row	1 (default)
parse numbers	true (default)
decimal character	. (default)
first row as names	true (default)
data set meta data information	semua atribut (80 atribut) di baca
Read not matching values as missing	true (default)

4.2. Set Rule

Langkah selanjutnya yaitu memilih operator Set Role yang berfungsi untuk mengubah peran lebih *attributes.*



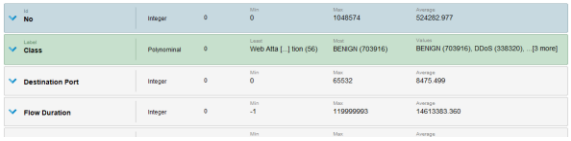
Gambar 4. Operator Set Role

Pada operator Set Role, parameter yang digunakan tampak pada tabel 2 sebagai berikut.

Tabel 2. Parameter Set Role

Parameter	**Nilai**	
attribute name	No	
target role	Id	
set additional roles	Attribute name	Target role
	Class	label

Hasil dari penggunaan operator Set Role tampak pada gambar 5 Dari 80 atribut, terdapat 2 atribut spesial yaitu No dan Class, dan 78 atribut merupakan atribut reguler dengan jumlah record sebanyak 1048083 record.



Gambar 5. Statistik Hasil Pembacaan Set Role

4.3. Preprocessing

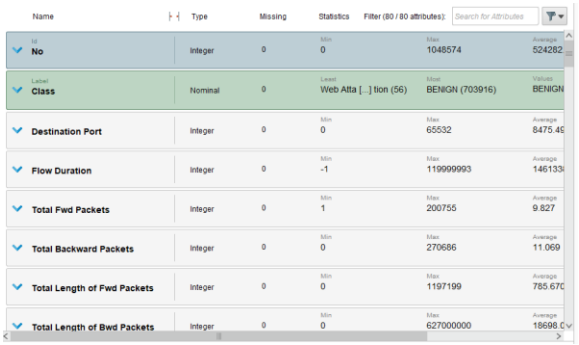
Pada tahap *preprocessing* ini bertujuan untuk menghilangkan *missing value* pada data yang akan digunakan sehingga pada saat diproses tidak muncul kesalahan atau masalah. Berikut table sebelum dilakukan *preprocessing* sebagai berikut.

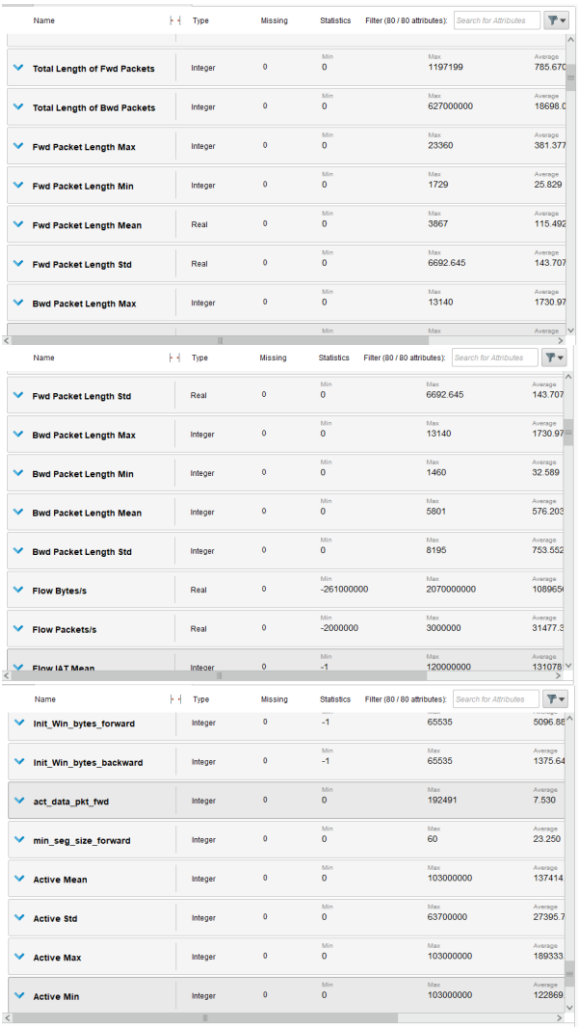
Tabel 3. Data sebelum dilakukan *Preprocessing*

	Destination Port	Flow Duration	Total Fwd Packets	Total Backward Packets	Total Length of Fwd Pac	Total Leng
0	80	9865922	5	0	30	
1	443	158423	24	22	703	
2	443	61163904	14	12	993	
3	443	110544045	18	18	1213	
4	53	185	2	2	104	
5	53	672	1	1	79	
6	51323	490655	6	2	11607	
7	53	163154	2	2	70	
8	80	1875517	3	6	26	
9	53	23800	2	2	64	
10	60799	69	1	1	6	
11	80	60510	3	6	26	
12	80	81875912	8	5	56	
13	80	550776	3	4	26	
14	53	30749	2	2	64	
15	51356	736	1	1	0	
16	80	8082373	4	0	24	
17	53	232	2	2	84	
18	53	156	2	2	136	
19	137	1	2	0	124	
20	53	328	2	2	102	
21	51255	77	1	1	0	

SDN_Intrusion

N	O	P	Q	R
0	0	10.34600846	1.724334743	724917.25
359.5	564.3708484	41.88777854	0.21219746	4908949.75
340	713.7394483	62.27650749	0.348483255	2941317.8
359.0909091	569.9183195	946.1853763	4.066990657	258823.2105
133	0	2156.033867	24.22509963	55039.33333
0	0	29742.76527	3215.434084	414.6666667
0	0	16.64333133	2.773888555	480673
0	0	62827.22513	10471.20419	191
6	0	4.998230388	0.833038398	1400495.667
2900.25	3540.00286	17755.69917	10.6897647	109138.6667
0	0	0	15384.61538	130
6	0	inf	inf	0
163	0	1903846.154	19230.76923	69.33333333
1934.5	2538.919278	1214.981641	1.458436335	738409.3846
2321.4	3173.373883	18046.40282	12.41048935	92088
2321.4	4413.201989	9644.921306	6.632800692	172303.8571
0	0	4.0269505	0.671158417	1862451.5
1934.5	2177.344966	158.6468505	0.190436072	5655037.231
179	0	1339552.239	7462.686567	178.6666667
61	0	1333333.333	28368.79433	47
1934.5	2538.919278	1329.968082	1.483193366	730406
1934.5	2538.919278	6745.132783	5.218447095	215581.375
60	0	952380.9524	21164.02116	63





Gambar 6. *Missing Value*

Pada tahap ini peneliti harus menghapus beberapa record karena berisi *INF* supaya data berjalan dengan baik dan normal. *Preprocessing* data dilakukan menggunakan *software* RapidMiner, setelah preprocessing data dilakukan didapat hasil bahwa dalam data tersebut tidak ada *missing value*, baik dalam *atributes* maupun *label*. Hal ini dapat dilihat pada gambar 6.

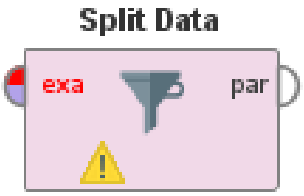
4.4. Transformation

Pada tahap transformation ini untuk penelitian yang penulis buat tidak riperlukan dikarenakan data yang ada sudah melewati tahap *cleansing* dan tidak ada lagi data yang *missing*. Untuk algoritma yang penulis buat menggunakan algortima deep learning H2O yang mana bisa membaca beberapa karakteristik type data seperti polynominal, integer, numerik.

4.5. Data Mining

Setelah melakukan preprocessing pada data tahapan selanjutnya yaitu data mining menggunakan operator *split data* yang berfungsi untuk menghasilkan jumlah subset yang diinginkan dari ExampleSet yang

diberikan. ExampleSet dipartisi menjadi subset sesuai dengan ukuran relatif yang ditentukan.



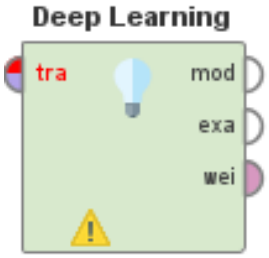
Gambar 7. Operator Split Data

Setelah memilih operator yang akan digunakan, parameter yang digunakan tampak pada tabel 4 sebagai berikut.

Tabel 4. Parameter Split Data

Parameter	**Nilai**	
partitions	Training	Testting
	0.8	0.2
Sampling type	Linier sampling	
Used local random seed	false (default)	

Langkah berikutnya untuk menentukan model pada penelitian yang sedang dilakukan pilih operator algoritma Deep Learning H2O.



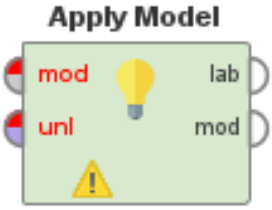
Gambar 8. Operator Deep Learning H2O

Setelah memilih operator yang akan digunakan, parameter yang digunakan tampak pada tabel 5 sebagai berikut.

Tabel 5. Parameter Deep Learning H2O

Parameter	**Nilai**
activation	rectifier
hidden layer sizes	50,100,50
Epochs	2
train samples per iteration	-2 (default)
adaptive rate	true (default)
Epsilon	1.0e-8
Rho	0.99
standardize	true
L1	1.0e-5 (default)
L2	0.0 (default)
max w2	10.0 (default)
expert parameters	semua atribut (80 atribut) di baca

Untuk parameters deep learning sendiri pada bagian activation yang di pilih adalah rectifier yang berfungsi sebagai memilih nilai maksimum (0,X) dimana X adalah nilai input, dan untuk hidden layer sizes peneliti memilih 50, 100, 50 yang berfungsi untuk melihat jumlah dan ukuran setiap lapisan tersembunyi dalam model. Fungsi dari operator Deep Learning yaitu memungkinkan dapat memprediksi akurasi yang tinggi, karena deep learning sendiri didasarkan pada jaringan saraf tiruan *feed forward multi-layer* yang dilatih dengan penurunan gradien stokastik menggunakan *backpropagation***.** Lankah selanjutnya adalah yaitu memilih operator Apply Model yang berfungsi untuk untuk mendapatkan prediksi pada data yang tidak terlihat atau untuk memodifikasi data dengan menerapkan model preprocessing.



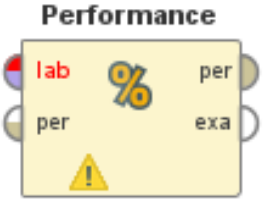
Gambar 9. Operator Apply Model

Setelah memilih operator yang akan digunakan, parameter yang digunakan tampak pada tabel 6 sebagai berikut.

Tabel 6. Paramater Apply Model

Parameter	**Nilai**
application parameters	default
create view	default

Apply Model sendiri Model pertama kali dilatih dalam ExampleSet oleh operator lain. Ini sering merupakan algoritma pembelajaran. Tujuannya biasanya untuk mendapatkan prediksi data yang tidak terlihat, atau untuk menerapkan model pra-pemrosesan untuk memodifikasi data. Setelah memasukkan operator Apply Model tambahkan operator Performance untuk mengevaluasi kinerja statistic dari kalsifikasi.



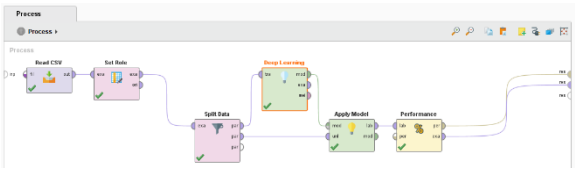
Gambar 10. Operator Performance

Setelah memilih operator yang akan digunakan, parameter yang digunakan tampak pada tabel 7 sebagai berikut.

Tabel 7. Parameter Performance

Parameter	Nilai
Main criterion	First
accuracy	True (default)
Classification eror	False (default)
Kappa	False (default)
Weighted mean recall	False (default)
Weighted mean precision	False (default)
Spearman rho	False (default)
Kendall tau	False (default)
Absolute error	False (default)
Skip underfined labels	true (default)
Use exsample weights	true (default)
Class weights	semua atribut (80 atribut) di baca

Setiap proses harus berisi hanya satu operator dari kelas ini, yang harus menjadi operator root dari proses tersebut. Operator ini menyediakan satu set parameter yang relevan secara global dengan proses, seperti logging dan parameter inisialisasi generator angka acak. Jika semua operator sudah di gunakan akan menghasilkan model klasifikasi seperti pda gambar 11.



Gambar 11. Model Klasifikasi Deep Learning H2O

4.6. Eksperimen

Berdasarkan hasil dari penelitian yang penulis buat bahwasanya untuk pengujian/eksperimen yang pertama dalam algortima deep learning H2O ini mengambil tingkat value sebesar (50,50), (50,100,50), (30,30,40) untuk epoch sendiri sebesar 10, dan tingkat Accuracy nya sebesar 99,66%. Pada eksperimen yang kedua penulis mengambil tingkat value sebesar (50,50), (50,100,50), (30,30,40), dan epoch sendiri sebesar 2, dan menghasilkan tingkat accuracy sebesar 99,66%. Pada eksperimen yang ketiga penulis mengambil tingkat value sebesar (50,50), (50,100,50), (30,30,40), dan epoch sendiri sebesar 100, dan menghasilkan tingkat accuracy sebesar 99,66%. Seperti pada tabel 8.

Tabel 8. Parameter Deep Learning H2O

Parameter	Nilai
hidden layer sizes	2 (50, 50) (default)
	3 (50, 100, 50)
	3 (30, 30, 40)
Epochs	10 (default)
	2
	100

Pada tabel 9 dijelaskan bahwa dalam parameter deep learning H2O yang di ubah adalah hidden layer sizes dan epochs. Terdapat tiga eksperimen yang peneliti buat yaitu untuk hidden layer sizes di buat menjadi beberapa nilai di antaranya 2 (50,50) (default), 3 (50,100,50), 3 (30,30,40) dan untuk nilai epochs nya peneliti mengambil 3 nlai diantaranya 10 (default), 2, dan 100.

Tabel 9. Eksperimen terhadap algoritma Deep Learning H2O

No. Eksperimen	Split	Value	Epoch	Accuracy
1	80%, 20%	(50,50)	10	99,66%
		(50,100,50)	10	99,66%
		(30,30,40)	10	99,66%
2		(50,50)	2	99,66%
		(50,100,50)	2	99,66%
		(30,30,40)	2	99,66%
3		(50,50)	100	99,66%
		(50,100,50)	100	99,66%
		(30,30,40)	100	99,66%

No. Eksperimen	Split	Value	Epoch	Accuracy
1	70%, 30%	(50,50)	10	99,66%
		(50,100,50)	10	99,66%
		(30,30,40)	10	99,66%
2		(50,50)	2	99,66%
		(50,100,50)	2	99,66%
		(30,30,40)	2	99,66%
3		(50,50)	100	99,66%
		(50,100,50)	100	99,66%
		(30,30,40)	100	99,66%

Dari hasil eksperimen tesebut dapat di jelaskan bahwasanya untuk penelitian yang penulis buat ini hasilnya tetap sama, walaupun sudah banyak eksperimen yang penulis buat ini mulai dari perubahan hidden layernya, epochsnya, semunya sama dengan nilai accuracy 99,66%. Dikarenakan penulis menggunakan sampling typenya yaitu *linier sampling* yang berfungsi sebagai dengan sampling linier, ExampleSet hanya dipartisi tanpa mengubah urutan.

4.7. Eksperimen Split Data

Dalam operator Split Data peneliti mengambil dua eksperimen untuk nilai Training dan Testting, Untuk Paramater yang di ubah iyalah parameter partitions dan sampling type. Peneliti mengambil nilai Eksperimen untuk nilai Training yaitu 0,8 (80%) dan 0,7 (70%) dan nilai Testting yaitu 0,2 (20%), dan 0,3(30%). Sedangakan untuk sampling typenya peneliti mengambil linier sampling.

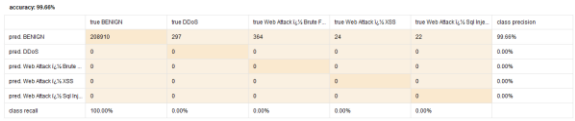
Tabel 10. Eksperimen dari Operator Split Data

Parameter	Value	
Partitions	Training	Testting
	0,8 (80%)	0,2 (20%)
	0,7 (70%)	0,3 (30%)
Sampling Type	Linier Sampling	

4.8. Tingkat Accuracy Performance

a. Untuk serangan yang terjadi di pred BENIGN bagian atribut benign memprediksi sebanyak 208910 atribut.
b. Untuk serangan yang terjadi di pred BENIGN bagian atribut DDOS menghasilkan 297 atribut.
c. Untuk serangan yang terjadi di pred BENIGN bagian atribut Web Attack Brute Force mengahsilkan 364 atribut.
d. Untuk serangan yang terjadi di pred BENIGN bagian atribut Web Attack XXS menghasilkan 24 atribut.
e. Untuk serangan yang terjadi di pred BENIGN bagian atribut Web Attack Sql Injection menghasilkan 22 atribut.
f. Untuk *class precision* atau tingkat kecocokan antara bagian data yang di ambil dengan informasi yang di butuhkan menghasilkan tingkat akurasi sebesar 99.66%. Sedangkan untuk atribut *Class Recall* menghasilkan 100.00% tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi.

Klasifikasi Software defined Network Intrusion menggunakan algoritma deep learning H2O didapat dengan tingkat nilai akurasi sebesar 99.66% dengan ratio 0.8:0.2 dapat di lihat pada gambar 12.



accuracy: 99.66%

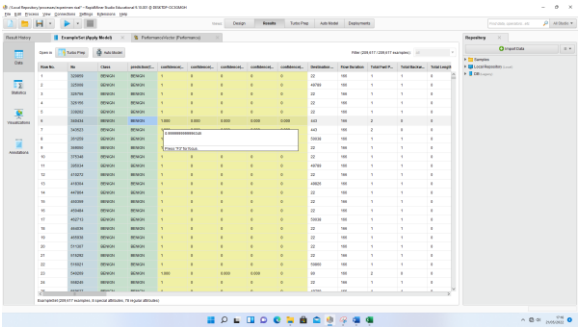
	true BENIGN	true DDoS	true Web Attack I¿½ Brute F...	true Web Attack I¿½ XSS	true Web Attack I¿½ Sql Inje...	class precision
pred. BENIGN	208910	297	364	24	22	99.66%
pred. DDoS	0	0	0	0	0	0.00%
pred. Web Attack I¿½ Brute ...	0	0	0	0	0	0.00%
pred. Web Attack I¿½ XSS	0	0	0	0	0	0.00%
pred. Web Attack I¿½ Sql Inj...	0	0	0	0	0	0.00%
class recall	100.00%	0.00%	0.00%	0.00%	0.00%	

Gambar 12. Hasil Akurasi Algortima Deep Learning

4.9. Tingkat Exsample Set Apply Model

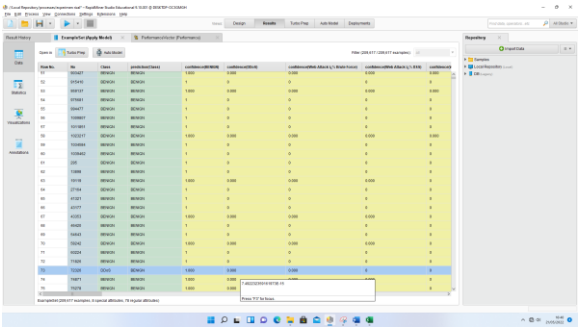
Terdapat empat penjelasan yang penulis buat mengenai exsample set apply model ini di antaranya:

1. Dalam record ke 340434, dan class BENIGN ternyata data memprediksi class yang hasilnya BENIGN kembali dengan nilai confidence (BENIGN) sebesar 0.9999999999996345 yang di bulatkan menjadi 1.000, untuk class BENIGN dan prediction classnya BENIGN yang bernilai 1 itu sudah benar tidak adanya pembulatan lagi, untuk confidence (DDOS) sebesar 0.000, confidence (web Attack Brute Force) sebesar 0.000, confidence (Web Attack XXS) sebesar 0.000, confidence (SQL Injection) sebesar 0.000 seperti pada gambar di bawah ini.



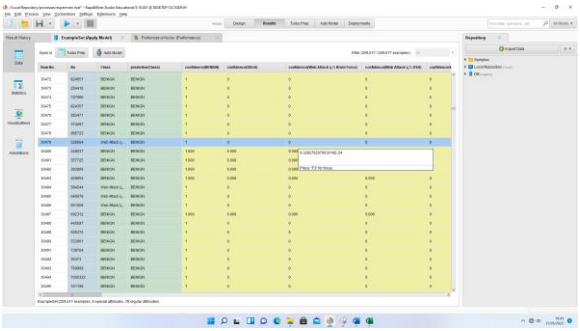
Gambar 13. Hasil Prediction class Benign

2. Dalam record ke 72326, dan class DDoS ternyata data memprediksi class yang hasilnya BENIGN kembali dengan nilai confidence (Benign) sebesar 0.9999999999996345 yang di bulatkan menjadi 1.000, untuk confidence (DDOS) sebesar 7.4922323591673E-15 yang di bulatkan menjadi 0.000, confidence (web Attack Brute Force) sebesar 0.000, confidence (Web Attack XXS) sebesar 0, confidence (SQL Injection) sebesar 0.000 seperti pada gambar di bawah ini.



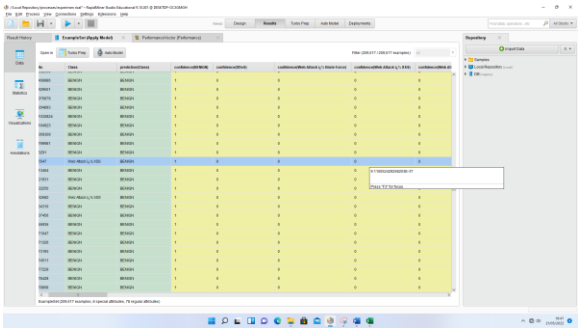
Gambar 14. Hasil Prediction Class DDOS

3. Dalam record 328064, dan class Web Attack Brute Force ternyata data memprediksi class yang hasilnya BENIGN kembali dengan nilai confidence (Benign) sebesar1, untuk confidence (DDOS) sebesar 0, confidence (Web Attack Brute Force) sebesar 9.328278207903015E-24 yang di bulatkan menjadi 0, confidence (Web Attack XXS) sebesar 0, confidence (SQL Injection) sebesar 0, seperti pada gambar di bawah ini.
4.



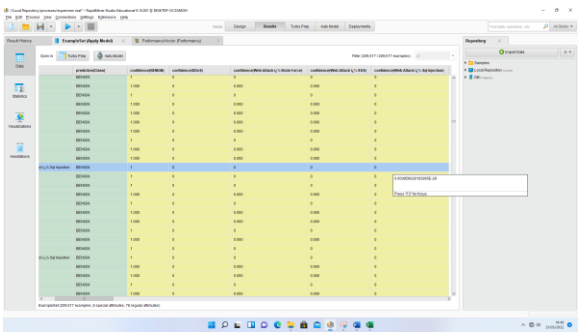
Gambar 15. Hasil Prediction Class Web Attack Brute Force

5. Dalam Record 725547, dan class Web Attack XXS ternyata data memprediksi class yang hasilnya BENIGN kembali dengan nilai confidence (BENIGN) sebesar 1, untuk confidence (DDOS) sebesar 0, confidence (Web Attack Brute Force) sebesar 0, confidence (Web Attack XXS) sebesar 9.110052429258203E-37 yang di bulatkan menjadi 0, confidence (SQL Injection) sebesar 0, seperti pada gambar di bawah ini



Gambar 16. Hasil Prediction Class Web Attack XXS

6. Dalam record 112843, dan class SQL Injection ternyata data memprediksi class yang hasilnya BENIGN kembali dengan nilai confidence (BENIGN) sebesar 1, untuk confidence (DDOS) sebesar 0, confidence (Web Attack Brute Force) sebesar 0, confidence (Web Attack XXS) sebesar 0, confidence (SQL Injection) sebesar 5.600858028193965E-29 yang di bulatkan menjadi 0, seperti pada gambar di bawah ini.



Gambar 17. Hasil Prediction Class SQL Injection

4.10. Pembahasan

Dari data di atas dapat di simpulkan dari 1048572 data *training* yang di proses bahwa model klasifikasi menggunakan algoritma *Deep Learning* memiliki tingkat akurasi yang lebih baik dengan tingkat akurasi 99.66%. dibandingkan dengan penelitian sebelumnya yang dilakukan oleh Husein Polat, Onur Polat, dan Aydin Cetin dengan menggunakan algoritma KNN (K-Nearest Neighbors) diperoleh dengan nilai akurasi sebesar 98.3%. Penelitian sebelumnya yang di lakukan oleh Omar Jamal Ibrahim, dan Wesam S. Bhaya dengan algoritma Support Vector Machine (SVM) diperoleh dengan nilai akurasi 97.77%. Hal ini menandakan bahwa algoritma deep learning lebih baik dari algoritma KNN, SVM pada penelitian sebelumnya, dan hasil dari dataset tersebut di tentukan oleh beberapa kriteria dari setiap atribut.

Pada penelitian sebelumnya Husein Polat, Onur Polat, dan Aydin Cetin menggunakan *feature selection* yang mana dataset dibuat menggunakan metode pemilihan fitur dataset yang sudah ada. Metode pemilihan fitur direkomendasikan untuk menyederhanakan model, memfasilitasi interpretasi, dan mengurangi waktu pelatihan. Kedua set data yang dibuat dengan dan tanpa seleksi fitur menggunakan model klasifikasi support vector machine (SVM), naive Bayes (NB), artifical neural networks(ANN), dan k-nearest neighbor (KNN). Sedangkan untuk peneliti bereksperimen menggunakan feature *selection parameter* yang berfungsi untuk mengubah suatu layer yang menjadikan tingkat akurasi yang tinggi. Peneliti juga bereksperimen menambahkan operator split data dan mengubah *paremeter partitions* dan *sampling type* dan mengahasilkan tingkat akurasi yang baik dan penelitian sebelumnya.

Disamping menggunakan metode *feature selection*, peneliti terdahulu juga menggunakan *Embedded-based feature selection* yaitu algoritma pemilihan fitur yang disematkan dengan algoritma klasifikasi. Algoritma ini melakukan seleksi fitur dengan mengidentifikasi fitur-fitur yang berkontribusi paling besar terhadap akurasi model. Pada metode ini, algoritma pembelajaran menggunakan proses seleksi variabel untuk melakukan seleksi fitur dan klasifikasi secara bersamaan. Untuk penelitian yang peneliti buat yaitu menggunakan algoritma *deep learning H2O* yang berfungsi untuk memproses data yang tidak tersturktur, yang memungkinkan dapat memprediksi akurasi yang tinggi, karena deep learning sendiri didasarkan pada jaringan saraf tiruan *feed forward multi-layer* yang dilatih dengan penurunan gradien stokastik menggunakan *backpropagation***.** Di samping itu peneliti melakukan ekperimen terhadap parameternya, yaitu mengubah nilai epochs, dan hidden layersnya yang membuat tingkat akurasi yang tinggi.

Penelitian ini menggunakan algoritma Lasso yang merupakan salah satu algoritma pemilihan fitur berbasis embedding. Fitur penting dari algoritma Lasso menghilangkan bobot dari fitur yang paling tidak penting. Artinya, pemilihan atribut terjadi secara alami seperti ini, mengurangi kumpulan fitur. Sedangakan peneliti menggunakan tetap menggunakan algortima deep learning H2O akan tetapi peneliti bereksperimen untuk menambahkan operator aplly model dan performance yang berfungsi untuk mendapatkan prediksi data yang tidak terlihat, atau untuk menerapkan model pra-pemrosesan untuk memodifikasi data. Peneliti mengubah hidden layers menjadi beberapa menjadi 3 nodes yaitu 50,50 (default) 50,100,50, dan 200,300,200 dan epochs menjadi 2, 10 (default), 3. Ketika bereskperimen tersebut peneliti menghasilnya tingkat akurasi tingga pada 3 node 50,100,50 dan untuk epochs 2 yaitu

dengan tingkat akurasi sebesar 99.66%. Hal ini menunjukan bahwa menggunakan algortiman deep learning H2O dengan algoritma KNN, SVM, dan NB lebih baik di bandingkan dengan penelitian sebelumnya

5. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian yang telah dilakukan didapatkan kesimpulan sebagai berikut: Peneliti menerapkan metode deep learning H2O untuk menganalisis dataset SDN *intrusion* dengan menggunakan *tools* rapid miner dan menggunakan beberapa operator diantaranya: Operator Read CSV, Operator Set Role, Operator Split Data, Operator Deep Learning H2O, Operator Apply Model, Operator Performance. Cara untuk menganalisis dataset SDN *intrusion* menggunakan algortima *deep Learning* ini peneliti menggunakan *Knowledge Discovery in Databases* (KDD) yang menjadi nilai penting yaitu, berdasarkan nilai recall sebesar 100.00%, tingkat akurasi sebesar 99.66% membuktikan bahwa algoritma Deep Learning pada penelitian ini berjalan dengan baik.

DAFTAR PUSTAKA

[1] Syahputra, M. Q., Akbi, D. R., & Risqiwati, D. (2020). Deteksi Dan Mitigasi Serangan DDoS Pada Software Defined Network Menggunakan Algoritma Decision Tree. *Jurnal Repositor*, *2*(11), 1491. https://doi.org/10.22219/repositor.v2i11.795

[2] Hande, Y., & Muddana, A. (2020). A survey on intrusion detection system for software defined networks (SDN). *International Journal of Business Data Communications and Networking*, *16*(1), 28–47. https://doi.org/10.4018/IJBDCN.2020010103

[3] Sultana, N., Chilamkurti, N., Peng, W., & Alhadad, R. (2019). Survey on SDN based network intrusion detection system using machine learning approaches. *Peer-to-Peer Networking and Applications*, *12*(2), 493–501. https://doi.org/10.1007/s12083-017-0630-0

[4] Susilo, B., & Sari, R. F. (2021). Intrusion Detection in Software Defined Network Using Deep Learning Approach. *2021 IEEE 11th Annual Computing and Communication Workshop and Conference, CCWC 2021*, 807–812. https://doi.org/10.1109/CCWC51732.2021.9375951

[5] Polat, H., & Polat, O. (2020). Detecting DDoS Attacks in Software-Defined Networks Through Feature Selection Methods and Machine Learning Models. *Mdpi*.

[6] Swami, R., Dave, M., & Ranga, V. (2020). Voting-based intrusion detection framework for securing software-defined networks. *Concurrency and Computation: Practice and Experience*, *32*(24), 1–16. https://doi.org/10.1002/cpe.5927

[7] Nofitri, R., & Irawati, N. (2019). Integrasi Metode Neive Bayes Dan Software Rapidminer Dalam Analisis Hasil Usaha Perusahaan Dagang. *JURTEKSI (Jurnal Teknologi Dan Sistem Informasi)*, *6*(1), 35–42. https://doi.org/10.33330/jurteksi.v6i1.393

[8] Supriadi, A., P, P., & Qurniawan, H. (2021). Metode Data Mining Klasifikasi Pada Kualitas Pelayanan Terhadap Nasabah Bank Syariah Mandiri dengan Model C4.5. *Jurasik (Jurnal Riset Sistem Informasi Dan Teknik Informatika)*, *6*(1), 150. https://doi.org/10.30645/jurasik.v6i1.279

[9] Elsayad, A. S., Desouky, A. I. E., Salem, M. M., & Badawy, M. (2020). A Deep Learning H2O Framework for Emergency Prediction in Biomedical Big Data. *IEEE Access*, *8*, 97231–97242.

[10] Ayudhitama, A. P., & Pujianto, U. (2020). Analisa 4 Algoritma Dalam Klasifikasi Liver Menggunakan Rapidminer. *Jurnal Informatika Polinema*, *6*(2), 1–9. https://doi.org/10.33795/jip.v6i2.274

[11] Mazdadi, M. I., Ramadhani, R., Saragih, T. H., & Haekal, M. (2021). Klasifikasi Tanaman Jarak Pagar Menggunakan Algoritme Deep Learning H2O. *Jurnal Komputasi*, *9*(1). https://doi.org/10.23960/komputasi.v9i1.2774