

# **Laporan SDI Data Mining**

**Mencari Perbandingan TP Rate Antara Normalisasi Log dan Normalisasi Zscore  
Pada Dataset NSL-KDD Menggunakan Metode Search Ranker dan Attribute  
Selection InfoGainAttributeEval**



**Nama Kelompok :**

**Achsan Noorsalam (05311840000021)**

**Christopher Benedict (05311840000024)**

**Justin Alfonsius S (05311840000043)**

**Bryan Yehuda M (05311940000021)**

**Zuhairaja MT (05311940000033)**

**Muhammad Zakky Ghufon (05311940000038)**

**Calvin Simatupang (05311940000049)**

**TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**2021**

## **I. Latar Belakang**

Sistem deteksi intrusi (IDS) secara dinamis memantau aktivitas sistem di lingkungan tertentu dan memutuskan apakah suatu aktivitas akan dianggap sebagai serangan atau tidak. Berdasarkan metode pendeteksiannya, IDS diklasifikasikan menjadi dua kategori, yaitu berbasis penyalahgunaan dan berbasis anomali. IDS berbasis penyalahgunaan menggunakan signature yang tersimpan dari serangan yang sudah diketahui untuk mengidentifikasi malicious behavior. IDS berbasis anomali adalah pendekatan untuk mendeteksi intrusi dengan terlebih dahulu dari aktivitas normal, dan jika ada jejak lalu lintas yang menyimpang dari data, maka sistem akan menandainya sebagai berbahaya.

### **1.1. Pengertian Machine Learning**

Pembelajaran mesin (Machine Learning) adalah pendekatan kecerdasan buatan (Artificial Intelligence (AI)) yang berfokus pada pembuatan mesin yang dapat belajar tanpa diprogram secara eksplisit. Jika kita akan membangun AI yang dapat melakukan tugas dengan kecerdasan seperti manusia, maka kita perlu membuat mesin yang bisa belajar sendiri, berdasarkan pengalaman masa lalu mereka.

Machine learning telah banyak mempengaruhi dunia industri, sebagian besar dunia industri yang bekerja dengan sejumlah besar data telah mengakui nilai teknologi penggunaan machine learning. Tujuannya adalah untuk mendapatkan wawasan dari data yang mereka miliki, dan dengan adanya teknologi ini dapat membuat pekerjaan menjadi lebih efisien atau lebih cepat dengan adanya data baru yang cenderung sama.

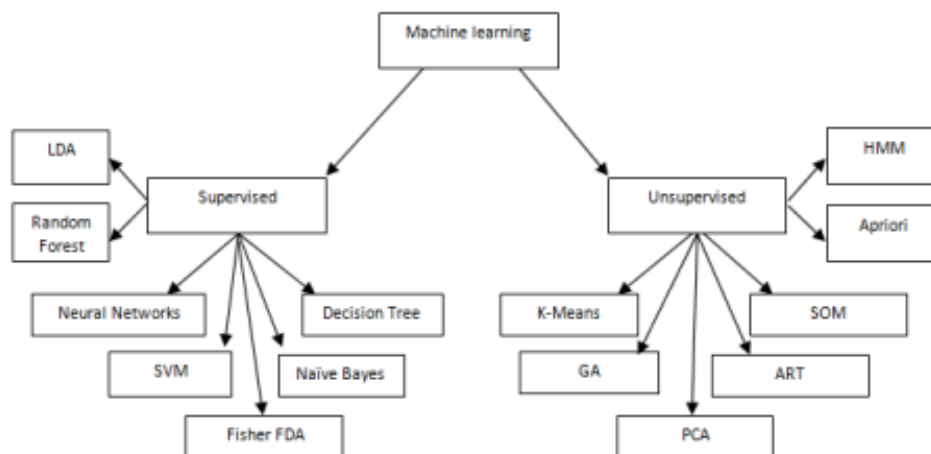
Menurut Gotama(2018) Machine Learning (ML) adalah teknik untuk melakukan inferensi terhadap data dengan pendekatan matematis. Inti machine learning adalah untuk membuat model (matematis) yang merefleksikan pola-pola data. Pada machine learning, inferensi yang dimaksud lebih menitikberatkan ranah hubungan variabel. Sementara itu, machine learning berada pada daerah representasi data/ilmu/pengetahuan dalam bentuk matematis karena keilmuan machine learning diturunkan dari matematika dan statistika. Machine learning ibarat sebuah “alat”, sama seperti rumus matematika. Bagaimana cara menggunakannya tergantung pada domain permasalahan. Tujuan machine learning minimal ada dua : memprediksi masa depan (unobserved event): dan/atau memperoleh ilmu pengetahuan (knowledge discovery/discovering unknown structure).

Machine Learning adalah subbidang komputer ilmu yang berkembang dari studi tentang pola pengenalan dan teori pembelajaran komputasi di kecerdasan buatan. Machine Learning mengeksplorasi konstruksi dan mempelajari algoritma yang dapat dipelajari, dan membuat prediksi pada data. Algoritma seperti itu beroperasi dengan membangun model dari input contoh untuk membuat prediksi berdasarkan data atau keputusan, daripada mengikuti secara ketat statis instruksi program. Model IDS adalah masalah pengklasifikasi multinomial yang dapat mengklasifikasikan peristiwa jaringan sebagai peristiwa normal atau serangan, seperti sebagai Denial of Service (DOS), Probe, U2R, dan R2L.

Tiga prasyarat untuk Machine Learning adalah:

- Data harus ada.
- Harus ada beberapa pola dalam data
- Tidak ada model matematika sederhana untuk data.

Teknik Machine Learning secara luas diklasifikasikan sebagai supervised dan unsupervised tergantung pada ada dan tidaknya label data. Gambar yang diberikan di bawah ini adalah representasi dari kemungkinan pendekatan yang telah diambil untuk merancang IDS dalam dua setengah dekade terakhir.



Gambar 1.1 Teknik Machine Learning

Supervised Learning dalam bahasa Indonesia adalah pembelajaran yang ada supervisornya. Maksud disini ada supervisornya adalah label di tiap data nya. Label maksudnya adalah tag dari data yang ditambahkan dalam machine learning model. Contohnya gambar kucing di tag “kucing” di tiap masing masing image kucing dan gambar anjing di tag “anjing” di tiap masing gambar anjing. Machine learning kategori dapat berupa classification (“anjing”, “kucing”, “beruang”, dsb) dan regression ( berat badan, tinggi badan dsb). Supervised learning banyak digunakan dalam memprediksi pola dimana pola tersebut sudah ada contoh data yang lengkap, jadi pola yang terbentuk adalah hasil pembelajaran data lengkap tersebut. Tentunya jika kita memasukan data baru, setelah kita melakukan ETL (Extract Transform Load) maka kita mendapat info feature feature dari sample baru tersebut. Kemudian dari feature feature tersebut di compare dengan pattern classification dari model yang didapat dari labeled data. Setiap label akan dicompare sampai selesai, dan yang memiliki percentage lebih banyak akan diambil sebagai prediksi akhir.

Unsupervised learning memiliki keunggulan dari unsupervised learning. Jika unsupervised learning memiliki label sebagai dasar prediksi baik serta membuat classification dan regression algorithm memungkinkan. Tetapi dalam realitanya, data real itu banyak yang tidak memiliki label. Label kebanyakan jika data sudah masuk ke ERP apapun bentuk ERPnya dan bagaimana kalau datanya berupa natural input seperti suara, gambar, dan video. Unsupervised learning tidak menggunakan label dalam memprediksi target features / variable. Melainkan menggunakan kesamaan dari atribut atribut yang dimiliki. Jika atribut dan sifat sifat dari data data feature yang diekstrak memiliki kemiripan, maka akan dikelompok kelompokkan (clustering). Sehingga hal ini akan menimbulkan kelompok kelompok (cluster). Jumlah cluster

bisa tidak terbatas. Dari kelompok kelompok itu model melabelkan, dan jika data baru mau di prediksi, maka akan dicocokkan dengan kelompok yang mirip featurenya.

Pengukuran terhadap kinerja suatu sistem klasifikasi merupakan hal yang penting. Kinerja sistem klasifikasi menggambarkan seberapa baik sistem dalam mengklasifikasikan data. *Confusion matrix* merupakan salah satu metode yang dapat digunakan untuk mengukur kinerja suatu metode klasifikasi. Pada dasarnya *confusion matrix* mengandung informasi yang membandingkan hasil klasifikasi yang dilakukan oleh sistem dengan hasil klasifikasi yang seharusnya

Terdapat 4 istilah sebagai representasi hasil proses klasifikasi pada *confusion matrix*. Keempat istilah tersebut adalah *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN). Agar lebih mudah memahaminya, bisa menggunakan contoh kasus sederhana untuk memprediksi seorang pasien menderita kanker atau tidak.

- ***True Positive* (TP)**

Merupakan data positif yang diprediksi benar. Contohnya, pasien menderita kanker (*class 1*) dan dari model yang dibuat memprediksi pasien tersebut menderita kanker (*class 1*).

- ***True Negative* (TN)**

Merupakan data negatif yang diprediksi benar. Contohnya, pasien tidak menderita kanker (*class 2*) dan dari model yang dibuat memprediksi pasien tersebut tidak menderita kanker (*class 2*).

- ***False Positive* (FP) — Type I Error**

Merupakan data negatif namun diprediksi sebagai data positif. Contohnya, pasien tidak menderita kanker (*class 2*) tetapi dari model yang telah memprediksi pasien tersebut menderita kanker (*class 1*).

- ***False Negative* (FN) — Type II Error**

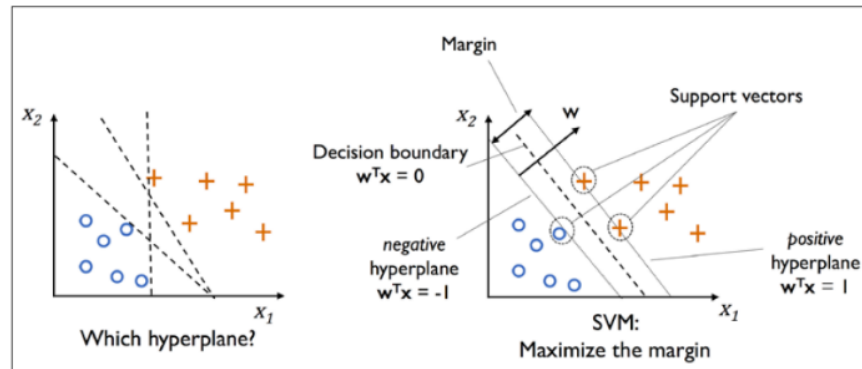
Merupakan data positif namun diprediksi sebagai data negatif. Contohnya, pasien menderita kanker (*class 1*) tetapi dari model yang dibuat memprediksi pasien tersebut tidak menderita kanker (*class 2*).

## 1.2 Pengertian SVM

Support Vector Machine (SVM) adalah salah satu dari tools yang paling populer digunakan dalam klasifikasi. SVM memiliki beberapa keunggulan seperti tidak adanya minimum lokal, kemampuan generalisasi yang tinggi, mampu beradaptasi dengan sejumlah kecil data sampel dan dimensi tinggi sampel data. Support Vector Machine merupakan salah satu metode dalam supervised learning yang biasanya digunakan untuk klasifikasi (seperti Support Vector Classification) dan regresi (Support Vector Regression). Dalam pemodelan klasifikasi, SVM memiliki konsep yang lebih matang dan lebih jelas secara matematis dibandingkan dengan teknik-teknik klasifikasi lainnya. SVM juga dapat mengatasi masalah klasifikasi dan regresi dengan linier maupun non linear.

SVM digunakan untuk mencari hyperplane terbaik dengan memaksimalkan jarak antar kelas. Hyperplane adalah sebuah fungsi yang dapat digunakan untuk pemisah antar kelas. Dalam 2D fungsi yang digunakan untuk klasifikasi antar kelas disebut sebagai line whereas, fungsi yang digunakan untuk klasifikasi antar kelas dalam 3D disebut plane similarly, pasangan

fungsi yang digunakan untuk klasifikasi di dalam ruang kelas dimensi yang lebih tinggi disebut hyperplane.



Gambar 1.2 Hyperplane yang memisahkan dua kelas positif (+1) dan negatif(-1)

*Hyperplane* yang ditemukan SVM diilustrasikan seperti Gambar 1 posisinya berada ditengah-tengah antara dua kelas, artinya jarak antara hyperplane dengan objek-objek data berbeda dengan kelas yang berdekatan (terluar) yang diberi tanda bulat kosong dan positif. Dalam SVM objek data terluar yang paling dekat dengan *hyperplane* disebut *support vector*. Objek yang disebut *support vector* paling sulit diklasifikasikan dikarenakan posisi yang hampir tumpang tindih (*overlap*) dengan kelas lain. Mengingat sifatnya yang kritis, hanya *support vector* inilah yang diperhitungkan untuk menemukan *hyperplane* yang paling optimal oleh SVM.

### 1.3 Pengertian Data Mining

Data Mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar (Turban dkk. 2005). Terdapat beberapa istilah lain yang memiliki makna sama dengan data mining, yaitu *Knowledge discovery in databases* (KDD), ekstraksi pengetahuan (*knowledge extraction*), Analisa data/pola (*data/pattern analysis*), kecerdasan bisnis (*business intelligence*) dan *data archaeology* dan *data dredging* (Larose, 2005). Kemampuan Data mining untuk mencari informasi bisnis yang berharga dari basis data yang sangat besar, dapat dianalogikan dengan penambangan logam mulia dari lahan sumbernya, teknologi ini dipakai untuk :

1. Prediksi trend dan sifat-sifat bisnis, dimana data mining mengotomatisasi proses pencarian informasi pemrediksi di dalam basis data yang besar.
2. Penemuan pola-pola yang tidak diketahui sebelumnya, dimana data mining menyapu basis data, kemudian mengidentifikasi pola-pola yang sebelumnya tersembunyi dalam satu sapuan.
3. Data mining berguna untuk membuat keputusan yang kritis, terutama dalam strategi.

Data mining mempunyai fungsi yang penting untuk membantu mendapatkan informasi yang berguna serta meningkatkan pengetahuan bagi pengguna. Pada dasarnya, data mining mempunyai empat fungsi dasar yaitu:

1. **Fungsi Prediksi (prediction).** Proses untuk menemukan pola dari data dengan menggunakan beberapa variabel untuk memprediksikan variabel lain yang tidak diketahui jenis atau nilainya.
2. **Fungsi Deskripsi (description).** Proses untuk menemukan suatu karakteristik penting dari data dalam suatu basis data.
3. **Fungsi Klasifikasi (classification).** Klasifikasi merupakan suatu proses untuk menemukan model atau fungsi untuk menggambarkan class atau konsep dari suatu data. Proses yang digunakan untuk mendeskripsikan data yang penting serta dapat meramalkan kecenderungan data pada masa depan.
4. **Fungsi Asosiasi (association).** Proses ini digunakan untuk menemukan suatu hubungan yang terdapat pada nilai atribut dari sekumpulan data.

Tahapan yang dilakukan pada proses data mining diawali dari seleksi data dari data sumber ke data target, tahap preprocessing untuk memperbaiki kualitas data, transformasi, data mining serta tahap interpretasi dan evaluasi yang menghasilkan output berupa pengetahuan baru yang diharapkan memberikan kontribusi yang lebih baik. Secara detail dijelaskan sebagai berikut (Fayyad, 1996):

1. Data selection

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam KDD dimulai. Data hasil seleksi yang digunakan untuk proses data mining, disimpan dalam suatu berkas, terpisah dari basis data operasional.

2. Pre-processing / cleaning

Sebelum proses data mining dapat dilaksanakan, perlu dilakukan proses cleaning pada data yang menjadi fokus KDD. Proses cleaning mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data.

3. Transformation

Coding adalah proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses data mining. Proses coding dalam KDD merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.

4. Data mining

Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode, atau algoritma dalam data mining sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses KDD secara keseluruhan.

5. Interpretation / evaluation

Pola informasi yang dihasilkan dari proses data mining perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap ini merupakan bagian dari proses KDD yang disebut interpretation. Tahap ini mencakup pemeriksaan apakah

pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesis yang ada sebelumnya.

#### **1.4 Tujuan Pembuatan Paper**

Dalam pembuatan paper ini bertujuan untuk membandingkan TP Rate dari class di dataset NSL-KDD seperti Probe, DoS, normal, R2I dan U2R dengan menggunakan metode normalisasi Zscore dan Log, dengan classifier SVM dan juga attribute selection InfoGainAttributeEval menggunakan software Weka. Dari hasil tersebut maka akan diketahui perbandingan TP Rate dari Zscore dan Log pada dataset NSL-KDD dan mana yang lebih baik diantara keduanya.

## **II. RELATED WORKS**

Banyak penelitian yang mengkaji pada anomaly-based intrusion detection. Beberapa diantaranya menggunakan machine learning dan data mining techniques. Neural networks, Bayesian parameter estimation, decision tree, dan clustering merupakan beberapa teknik yang dapat digunakan dalam mendeteksi kegiatan mencurigakan pada jaringan komputer. DHAK(2-1) menggunakan algoritma genetik (GA) untuk mendesain sistem deteksi intrusi yang revolusioner. Saha et al (2-2) membuat sistem deteksi yang menggunakan SVM untuk mengklasifikasi data menjadi paket serangan dan paket normal. Dengan menggunakan GA untuk meningkatkan tingkat klasifikasinya. Penelitian ini telah berhasil menemukan berbagai macam serangan baru. Neural network yang berbeda seperti PCA, MLP, dan gray neural network algorithm juga telah digunakan dalam berbagai sistem deteksi intrusi. Terutama pada penyalahgunaan deteksi(2-3,2-4).

Meskipun ini metode memiliki kinerja yang dapat diterima dalam mendeteksi serangan yang diketahui, mereka perlu diperbarui dengan serangan-serangan baru dan karenanya tidak praktis. Jaringan kekebalan buatan adalah pendekatan pengelompokan bio-terinspirasi lain yang digunakan dalam [2-5] merancang IDS berbasis anomali. Sebuah IDS hybrid multi-level menggunakan SVM dan Extreme Machine Learning diusulkan oleh Al-Yaseen et al. [2-6]. Mereka menggunakan k-mean yang dimodifikasi untuk meningkatkan kualitas pelatihan dari dataset. Pendekatan ini juga dapat mendeteksi semua kelas serangan. Akurasi klasifikasi kelas keseluruhannya lebih tinggi.

### III. KAJIAN PUSTAKA

#### 3.1 Ukuran evaluasi atribut

Pengukuran berdasarkan konten informasi banyak digunakan dalam pembelajaran mesin. Jumlah informasi dari hasil didefinisikan sebagai logaritma negatif dari probabilitasnya sebagai berikut:

$$I(X_j) = -\log_2 P(X_j) \quad (1)$$

Jumlah rata-rata informasi disebut entropi dari suatu hasil. Jika percobaan ini memiliki hasil  $m$  yang terputus putus dan hasil  $X_j$  dimana  $j = 1.. m$  dan  $\sum_j P(X_j) = 1$  maka entropi dihasilkan sebagai:

$$H(X) = - \sum P(X_j) \log_2 P(X_j) \quad (2)$$

InfoGainAttributeEval mengevaluasi nilai atribut dengan mengukur perolehan informasi sehubungan dengan kelas. InfoGainAttributeEval memiliki perhitungan sebagai berikut :

$$InfoGain(A) = HC - HC|A \quad (3)$$

Dimana di mana H adalah entropi informasi, C adalah Class dan A adalah attribute.

#### 3.2 Normalisasi

Pada bagian ini menjelaskan metodologi normalisasi untuk mengevaluasi dua skema normalisasi atribut untuk mengklasifikasi deteksi intrusi, yaitu: z-score dan normalisasi log.

Z-score merupakan pengukuran numerik yang menggambarkan hubungan nilai dengan mean dari sekelompok nilai. Z-score diukur dalam hal standar deviasi dari mean. Jika Z-score adalah 0, ini menunjukkan bahwa skor titik data identik dengan skor rata-rata Z-score menggunakan perhitungan rumus sebagai berikut:

$$s' = (s - \mu) / \sigma \quad (4)$$

Setelah itu ada Log Normalisasi yang merupakan suatu proses pemetaan dari peristiwa log menjadi taksonomi. Log Normalisasi memiliki rumus sebagai berikut:

$$x' = \log(1 + x) \quad (5)$$

di mana  $x$  adalah nilai fitur sebelum normalisasi dan  $x'$  adalah nilai setelah normalisasi.

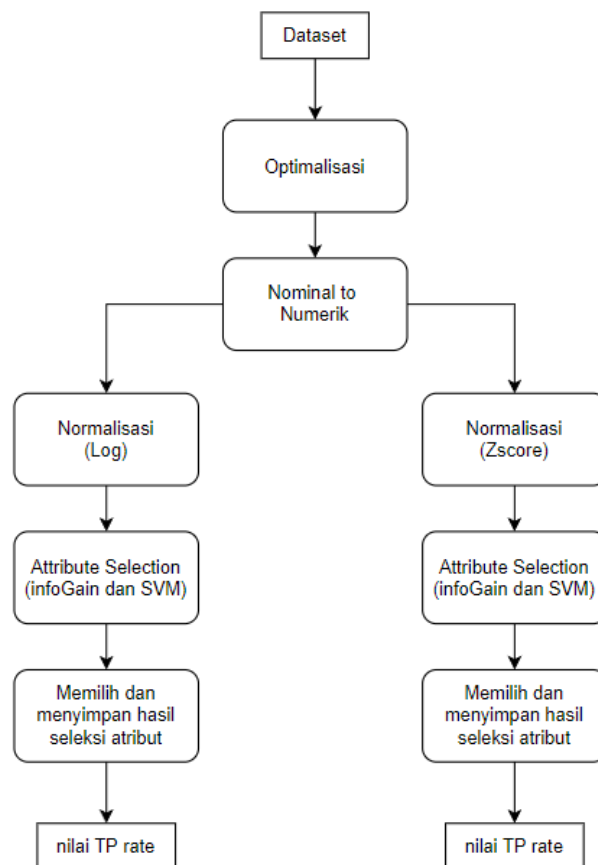


### 3.3 Seleksi Fitur

Metode seleksi fitur pada paper ini menggunakan metode search “Ranker”. Metode “Ranker” memberi peringkat atribut berdasarkan evaluasi individualnya. Gunakan bersama dengan evaluator atribut (ReliefF, Gain Ratio, Entropy, dll.) dengan parameter yang menghasilkan peringkat (benar atau salah). Metode ranker umumnya melakukan pemeringkatan atribut mana yang harus mendapatkan peringkat tinggi atau rendah sesuai dengan atribut yang dipilih dalam kumpulan data yang diberikan. Ranker memberikan peringkat atribut, secara berurutan berdasarkan skornya kepada evaluator

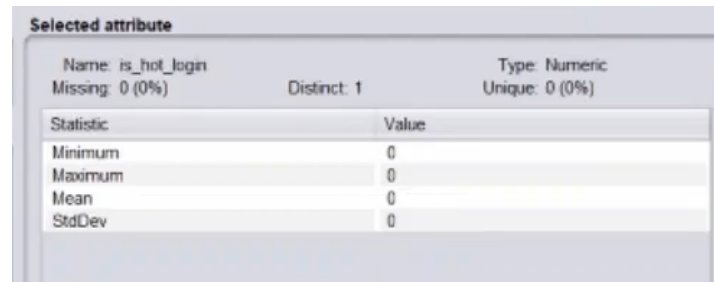
## IV. METODOLOGI

Untuk melakukan percobaan yang telah kami jelaskan pada latar belakang, pertama-tama disini kami akan menggunakan contoh dataset NSL-KDD. Setelah itu, dataset akan melalui 5 tahapan yang kami proses pada aplikasi Weka, agar nantinya kami bisa membandingkan tingkat akurasi atau True Positive rate dari 2 metode normalisasi. Adapun 5 tahapan tersebut kami buat dalam bentuk diagram yang disajikan pada gambar 4.1.



Gambar 4.1 Diagram tahapan metodologi

Pada tahap optimalisasi cukup sederhana, yaitu kami menghilangkan atribut yang tidak berguna (*useless*). Tahapan ini berguna untuk mempercepat proses dengan cara menghilangkan atribut apapun yang tidak akan mempengaruhi percobaan kami. Adapun yang dimaksud atribut yang tidak berguna disini dapat dilihat pada gambar 4.2. Pada statistik minimum, maximum, mean, dan stdDev, semuanya bernilai 0. Yang berarti, atribut tersebut tidak akan menghasilkan nilai apapun walau kita jalankan.



The screenshot shows the 'Selected attribute' dialog box in Weka. It displays the attribute 'is\_hot\_login' with a type of 'Numeric'. The statistics table shows that all values (Minimum, Maximum, Mean, StdDev) are 0, indicating that the attribute is constant and therefore useless.

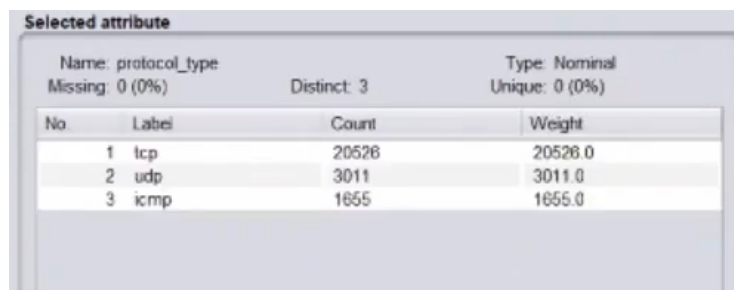
Selected attribute	
Name: is_hot_login	Type: Numeric
Missing: 0 (0%)	Distinct: 1
Unique: 0 (0%)	
Statistic	Value
Minimum	0
Maximum	0
Mean	0
StdDev	0

Gambar 4.2 Contoh atribut yang tidak berguna

Dan perintah yang dijalankan untuk melakukan optimalisasi adalah sebagai berikut :

```
"weka.filters.unsupervised.attribute.RemoveUseless"
```

Tahapan kedua yaitu merubah atribut nominal ke numerik. Berdasarkan dataset yang sudah dioptimalisasi tadi, apabila kita coba lihat satu persatu akan ada beberapa atribut yang memiliki nilai nominal. Sedangkan, untuk melakukan perbandingan kami membutuhkan nilai numerik. Sehingga, atribut-atribut yang ada dalam dataset yang memiliki nilai nominal harus kita ubah menjadi nilai numerik. Adapun contoh perbandingan dari atribut dengan nilai nominal sebelum dan sesudah diubah menjadi nilai numerik dapat dilihat pada gambar 4.3 dan gambar 4.4.



The screenshot shows the 'Selected attribute' dialog box in Weka for the attribute 'protocol\_type'. It has a type of 'Nominal'. The table below shows the distribution of values: 'tcp' (20526), 'udp' (3011), and 'icmp' (1655).

Selected attribute			
Name: protocol_type		Type: Nominal	
Missing: 0 (0%)		Distinct: 3	
Unique: 0 (0%)			
No.	Label	Count	Weight
1	tcp	20526	20526.0
2	udp	3011	3011.0
3	icmp	1655	1655.0

Gambar 4.3 Contoh atribut dengan nilai nominal (sebelum)

Selected attribute		
Name: protocol_type	Distinct: 3	Type: Numeric
Missing: 0 (0%)		Unique: 0 (0%)
Statistic	Value	
Minimum	0	
Maximum	2	
Mean	0.251	
StdDev	0.565	

Gambar 4.4 Contoh atribut dengan nilai nominal (sesudah)

Sedangkan perintah yang digunakan adalah sebagai berikut :

```
"weka.filters.unsupervised.attribute.OrdinalToNumeric"
```

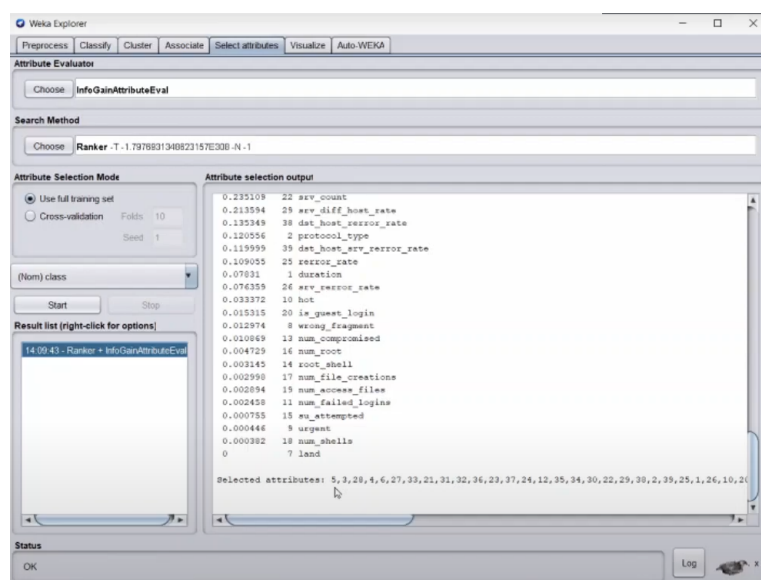
Tahapan selanjutnya yaitu melakukan normalisasi. Normalisasi disini dimaksudkan untuk menyamakan rentang nilai antar atribut, yaitu nilai minimum 0 dan maksimumnya 1. Adapun metode normalisasi yang akan kami gunakan disini adalah metode log dan Zscore. Dalam Weka, normalisasi ini dapat dilakukan dengan perintah :

```
"weka.filters.unsupervised.attribute.MathExpression-Elog(1+A)-Rlast"
```

Adapun untuk metode Zscore, dapat dilakukan dengan perintah berikut :

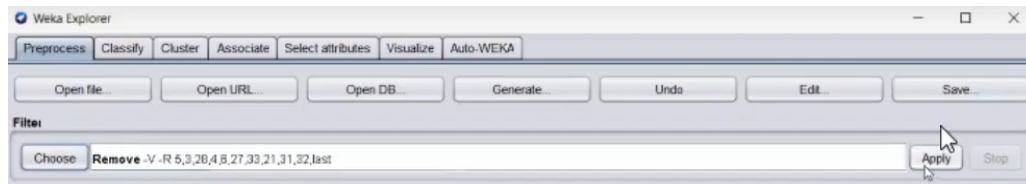
```
"weka.filters.unsupervised.attribute.MathExpression-E(A-MEAN)/S"
```

Lalu yang keempat disini kami akan melakukan seleksi fitur yang paling berguna (Attribute selection). Dapat disebut juga bahwa pada tahap ini kami mengurutkan atribut yang memiliki hubungan paling banyak dengan attribute classifier dengan metode search "ranker" dan attribute evaluator "SVM" dan "infoGain". Lebih lengkapnya dapat dilihat pada gambar 4.5



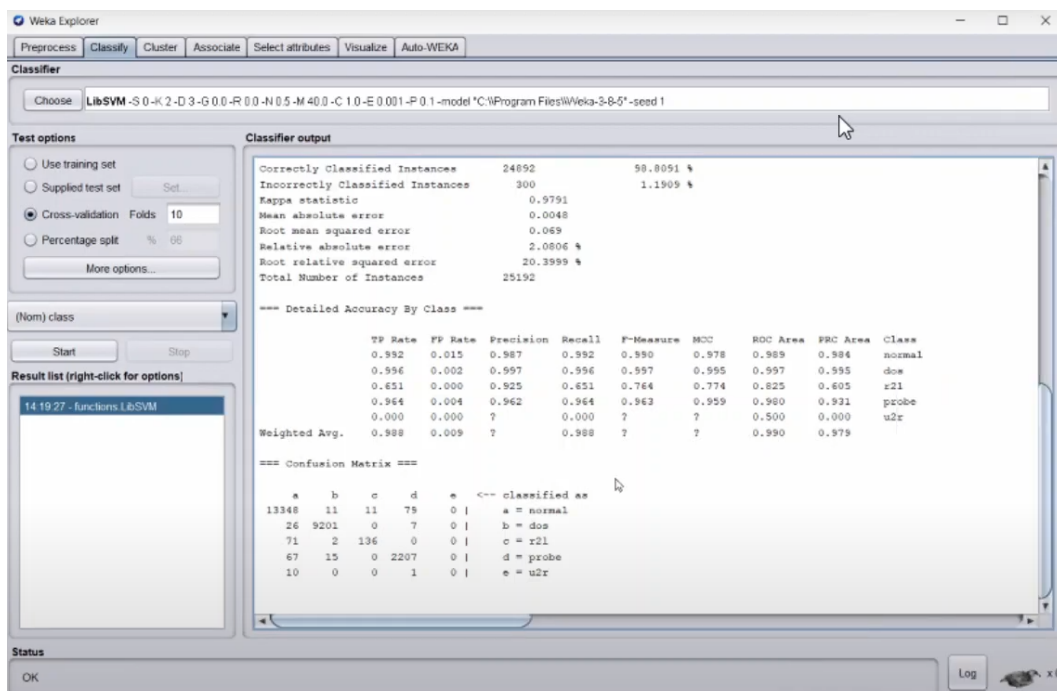
Gambar 4.5 Contoh seleksi fitur menggunakan infoGain

Selanjutnya tahap yang terakhir yaitu memilih dan menyimpan hasil seleksi atribut, yaitu dengan menggunakan perintah “remove” berdasarkan pada setiap urutan rank atribut seleksi dari 0 hingga 39. Sebagai contoh pada 10 atribut tertinggi, maka perintahnya dapat dilihat seperti gambar 4.6.



Gambar 4.6 Contoh perintah pada 10 atribut tertinggi

Setelah menjalankan perintah tersebut, selanjutnya pada menu Classify akan muncul nilai sebagaimana pada gambar 4.7. Disana didapatkan nilai TP rate dari setiap class yang mana nantinya akan kita bandingkan antara metode normalisasi.



Gambar 4.7 Perolehan nilai

## V. HASIL DAN PEMBAHASAN

### 5.1 Percobaan yang Dilakukan

Kami melakukan percobaan berdasarkan dari Jurnal yang berjudul “Increasing Accuracy and Completeness of Intrusion Detection Model Using Fusion of Normalization, Feature Selection Method and Support Vector Machine” hasil karya dari Bambang Setiawan, Tohari Ahmad, dan Supeno Djanali [5.1].

Kami melakukan percobaan dengan menggunakan bahasa pemrograman Java dan menggunakan Library Weka 3.8.3 [5.2]. OAO Multi-Class SVM dengan kernel RBF diimplementasikan menggunakan paket LibSVM yang sudah terintegrasi dengan Weka [5.3]. TP (True Positive) dilambangkan sebagai jumlah sampel positif yang diprediksi dengan benar sebagai positif.

Dataset NSL-KDD [5.4] digunakan dalam percobaan ini. Dataset ini memiliki lima kelas yaitu Normal, Probe, DoS, U2R, dan R2L. Eksperimen menggunakan seluruh Dataset Training NSLK-DD, yang berisi 125.973 data. Dengan komposisi kelas serangan sebagai berikut: Normal memiliki 67343 data, DoS memiliki 45927 data, Probe memiliki 11656 data, R2L memiliki 995 data, dan U2R memiliki 52 data. Kami mengujinya menggunakan metode validasi silang 10 kali lipat Informasi tentang dataset NSL-KDD dan serangannya dapat ditemukan di [5.5].

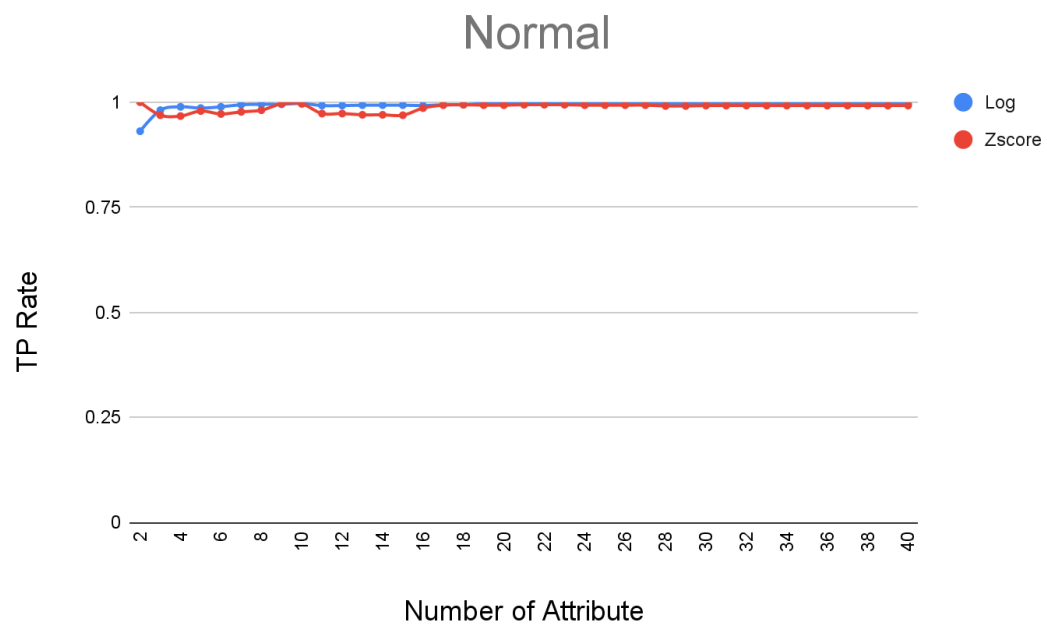
Tiga atribut non-numerik dalam dataset, yaitu flag, service, dan protocol\_type, telah diubah menjadi numerik dengan mengkategorikannya ke dalam bilangan bulat yang sesuai. Sebelum memulai percobaan ini, telah dilakukan proses normalisasi Log atau proses normalisasi Zscore yang disesuaikan dengan proses normalisasi yang ingin didapatkan datanya. Proses ini kemudian dilanjutkan dengan proses pemilihan fitur menggunakan ranker menggunakan InfoGainAttributeEval sebagai jenis Attribute Selection yang digunakan. 39 atribut teratas yang diperoleh dari proses seleksi fitur untuk normalisasi log maupun normalisasi Zscore akan ditunjukkan pada tabel berikut.

Rank	Attributes	Description
1	src_bytes	Jumlah data per byte yang di transfer dari source ke destination
2	service	Layanan yang digunakan oleh destination network
3	diff_srv_rate	Persentase koneksi yang menuju layanan yang berbeda, di antara koneksi yang dikumpulkan dalam hitungan diantara koneksi yang dikumpulkan dalam hitungan
4	flag	Status dari connection (error atau normal)
5	dst_bytes	Jumlah data per byte yang di transfer dari destination ke source
6	same_srv_rate	Persentase koneksi yang menuju layanan yang sama, di antara koneksi dikumpulkan dalam hitungan

7	dst_host_diff_srv_rate	Persentase koneksi yang menuju layanan yang berbeda di antara koneksi dikumpulkan dalam jumlah host pertama
8	count	Jumlah koneksi yang sama dengan destinasi host
9	dst_host_srv_count	Jumlah koneksi yang memiliki port number yang sama
10	dst_host_same_srv_rate	Persentase koneksi yang menuju layanan yang sama di antara koneksi dikumpulkan dalam jumlah host pertama
11	dst_host_serror_rate	Persentase koneksi yang telah mengaktifkan flag s0, s1, s2 atau s3, di antara koneksi yang dikumpulkan dalam jumlah host pertama
12	serror_rate	Persentase koneksi yang telah mengaktifkan flag s0, s1, s2 atau s3, di antara koneksi yang dikumpulkan dalam hitungan
13	dst_host_srv_serror_rate	Persentase koneksi yang telah mengaktifkan flag s0, s1, s2 atau s3, di antara koneksi yang dikumpulkan dalam jumlah srv host pertama
14	srv_serror_rate	Persentase koneksi yang telah mengaktifkan flag s0, s1, s2 atau s3, di antara koneksi yang dikumpulkan dalam jumlah srv host pertama
15	logged_in	Menandakan login status (1 jika sukses, 0 jika gagal)
16	dst_host_srv_diff_host_rate	Persentase koneksi yang menuju mesin tujuan berbeda di antara koneksi yang dikumpulkan dalam jumlah srv host dst
17	dst_host_same_src_port_rate	Persentase koneksi yang menuju port sumber yang sama di antara koneksi yang dikumpulkan dalam jumlah srv host pertama
18	dst_host_count	Jumlah koneksi yang memiliki destinasi host ip address yang sama
19	srv_count	Jumlah koneksi yang sama dengan layanan (sesuai dengan jenis port number yang diakses)
20	srv_diff_host_rate	Persentase koneksi yang menuju mesin tujuan berbeda di antara koneksi yang dikumpulkan dalam hitungan srv
21	dst_host_rerror_rate	Persentase koneksi yang telah mengaktifkan flag (#4) rej, di antara koneksi yang dikumpulkan dalam jumlah host pertama
22	protocol_type	Protokol yang digunakan
23	dst_host_srv_rerror_rate	Persentase koneksi yang telah mengaktifkan flag (#4) rej, di antara koneksi yang dikumpulkan dalam jumlah srv host pertama
24	rerror_rate	Persentase koneksi yang telah mengaktifkan flag (#4) rej, di antara koneksi yang dikumpulkan dalam hitungan
25	duration	Panjang durasi waktu pada koneksi
26	srv_rerror_rate	Persentase koneksi yang telah mengaktifkan flag (#4) s0, s1, s2 atau s3, di antara koneksi yang dikumpulkan dalam jumlah srv
27	hot	Jumlah yang mengindikasikan serangan yang memasuki sistem direktori
28	is_guest_login	Jika login sebagai guest maka akan bernilai 1, jika tidak maka 0
29	wrong_fragment	Jumlah angka yang salah pada koneksi
30	num_compromised	Jumlah compromised condition

31	num_root	Jumlah user yang beroperasi sebagai root
32	root_shell	Menunjukkan status dari root shells (1 jika root shell obtained, 0 jika root shell not obtained)
33	num_file_creations	Jumlah file yang sedang dioperasikan
34	num_access_files	Jumlah operation yang diakses oleh control file
35	num_failed_logins	Jumlah user yang failed saat login
36	su_attempted	Menunjukkan penggunaan perintah “su root” (1 jika iya, 0 jika tidak)
37	urgent	Jumlah paket urgent
38	num_shells	Jumlah shell prompts di dalam koneksi
39	land	Jika ip address sumber dan tujuannya sama, maka akan bernilai 1, jika tidak maka 0

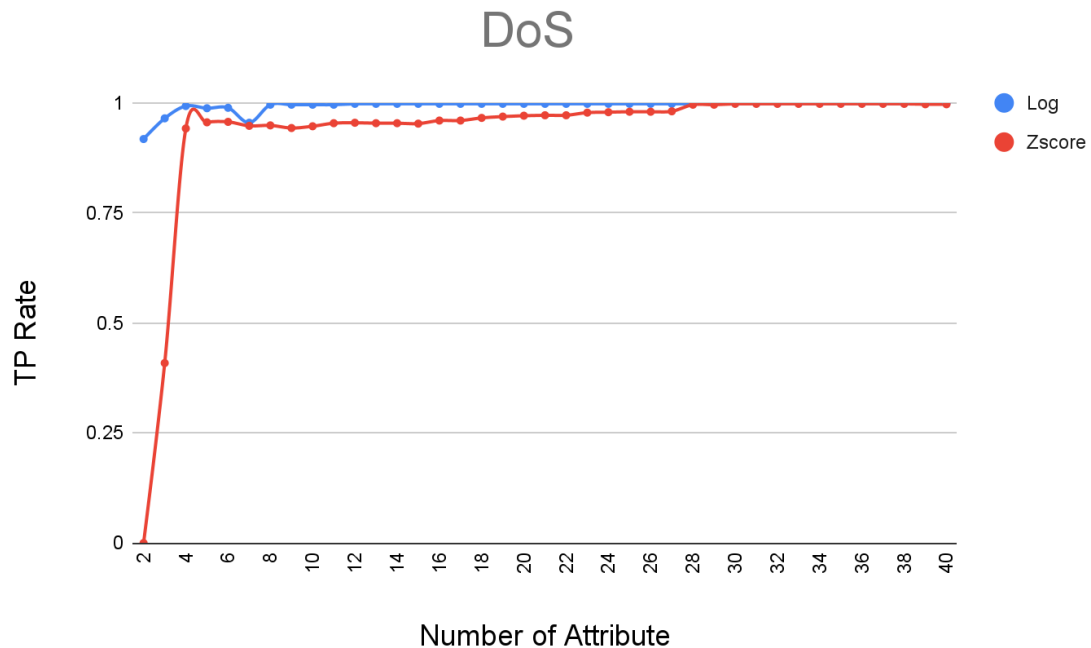
## 5.2 Hasil yang Didapatkan



Gambar 5.1

Perbandingan TP Rate Normalisasi Log dan Normalisasi Zscore Dalam Kelas Normal

Bisa dilihat pada grafik Perbandingan TP Rate diatas bahwa pada kelas Normal, didapati bahwa hasil TP Rate dari kedua metode normalisasi, baik untuk Normalisasi Log dan juga Normalisasi Zscore, sudah cukup tinggi dan hampir mendapatkan nilai 1. Namun untuk beberapa jumlah atribut awal, Normalisasi Log memiliki nilai yang relatif lebih tinggi dari Normalisasi Zscore meskipun tidak jauh berbeda. Kita harus melihat kelas lainnya pada Dataset NSL-KDD untuk bisa lebih melihat perbandingan antara kedua metode normalisasi ini.

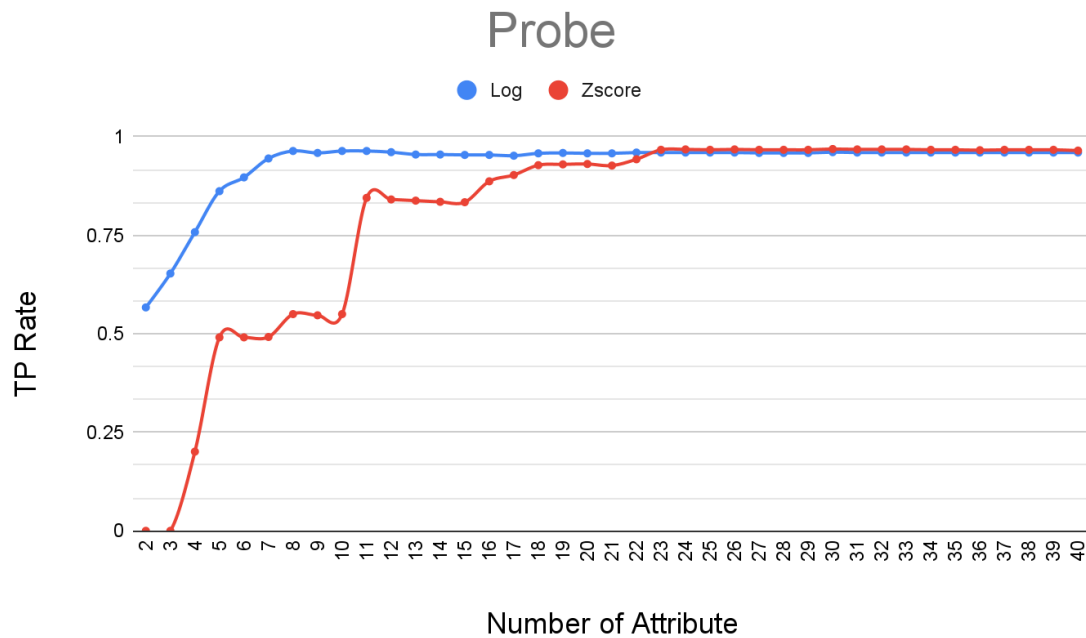


Gambar 5.2

Perbandingan TP Rate Normalisasi Log dan Normalisasi Zscore Dalam Kelas DoS

Bisa dilihat pada grafik Perbandingan TP Rate antara normalisasi Log dan juga normalisasi Zscore pada kelas DoS di atas, bahwa normalisasi Log sudah bisa menghasilkan TP Rate yang cukup tinggi pada atribut ke-2. Sedangkan untuk normalisasi Zscore didapati bahwa untuk bisa mendapatkan TP Rate yang tinggi atau mendekati TP Rate milik normalisasi Log, dibutuhkan minimal 4 atribut. Pada atribut ke-4 dan seterusnya hingga atribut ke-27 hasil dari Normalisasi Log terus menerus berada di atas hasil Normalisasi Zscore hingga pada atribut ke-28 hasil Normalisasi Zscore bisa menyamai TP Rate dari Normalisasi Log.

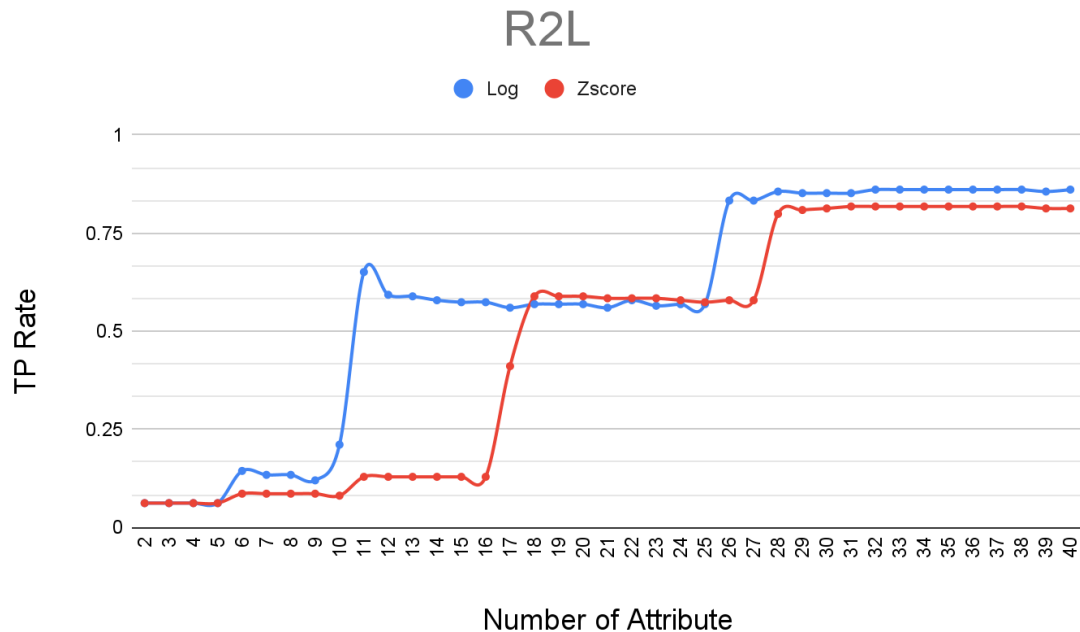




Gambar 5.3

Perbandingan TP Rate Normalisasi Log dan Normalisasi Zscore Dalam Kelas Probe

Bisa dilihat pada grafik Perbandingan TP Rate antara normalisasi Log dan juga normalisasi Zscore pada kelas Probe di atas, bahwa normalisasi Log bisa menghasilkan TP Rate yang cukup tinggi pada atribut ke-8. Sedangkan untuk normalisasi Zscore didapati bahwa untuk bisa mendapatkan TP Rate yang tinggi atau mendekati TP Rate milik normalisasi Log, dibutuhkan minimal 18 atribut. TP Rate dari Normalisasi Log terus menerus meningkat mulai dari atribut ke-2 hingga akhirnya mencapai maksimumnya di atribut ke-8. TP Rate dari Normalisasi Log ini juga secara konsisten terus berada di atas TP Rate dari Normalisasi Zscore pada 22 atribut pertama hingga disamai oleh Normalisasi Zscore pada atribut ke-23 dimana keduanya menghasilkan nilai yang sama hingga atribut ke-40.

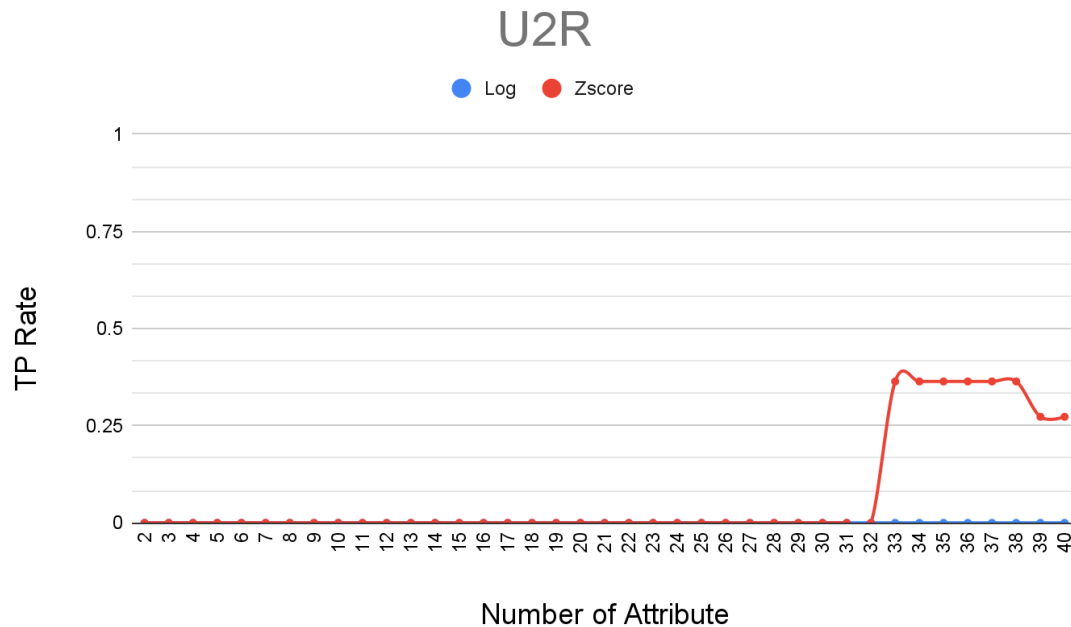


Gambar 5.4

Perbandingan TP Rate Normalisasi Log dan Normalisasi Zscore Dalam Kelas R2L

Untuk kelas R2L terjadi hal yang cukup menarik dimana terjadi 3 tahap peningkatan TP Rate untuk Normalisasi Log dan 4 tahap peningkatan TP Rate untuk Normalisasi Zscore. Peningkatan pertama untuk Normalisasi Log terjadi pada atribut ke-6 dimana sampai atribut ke-9 nilainya relatif stagnan. Peningkatan kedua untuk Normalisasi Log terjadi pada atribut ke-11 dimana nilainya turun sedikit pada atribut ke-12. Peningkatan kedua ini menunjukkan nilai TP Rate yang cukup stagnan hingga pada atribut ke-26 terjadi peningkatan yang ketiga dan diikuti dengan nilai TP Rate yang stagnan hingga atribut ke-40.

Sedangkan untuk Normalisasi Zscore, peningkatan pertama terjadi pada atribut ke-6 dan peningkatan kedua terjadi pada atribut ke-11. Meskipun peningkatan pertama dan kedua ini hampir sama dengan milik Normalisasi Log, tapi Normalisasi Zscore masih belum dapat memberikan TP Rate yang tinggi seperti Normalisasi Log. Namun pada atribut ke-18, Normalisasi Zscore mengalami peningkatan yang ke-3 dimana dia bisa menyamai TP Rate milik Normalisasi Log meskipun terlambat 6 atribut. Dan pada atribut ke-28, Normalisasi Zscore mengalami peningkatan terakhirnya yang mana nilainya stagnan dan konsisten terus berada di bawah Normalisasi Log hingga atribut ke-40.



Gambar 5.5  
Perbandingan TP Rate Normalisasi Log dan Normalisasi Zscore Dalam Kelas U2R

Untuk kelas U2R, baik Normalisasi Log maupun Normalisasi Zscore sama-sama kompak menunjukkan TP Rate 0 mulai dari atribut ke-2 hingga atribut ke-33, dimana untuk Normalisasi Zscore mulai menghasilkan TP Rate. TP Rate yang dihasilkan oleh Normalisasi Zscore ini nilainya tetap stagnan hingga pada atribut ke-39 terjadi penurunan. Meskipun TP Rate yang dihasilkan oleh Zscore ini nilainya dibawah 50%, tetapi masih lebih baik jika dibandingkan dengan TP Rate milik Normalisasi Log yang bernilai 0 mulai dari atribut ke-2 hingga atribut ke-40.

## **VI. KESIMPULAN**

Setelah dilakukan metode penelitian perbandingan TP Rate Antara Normalisasi Log dan Normalisasi Zscore Pada Dataset NSL-KDD (Class Probe, Normal, DoS, R2L, U2R) Menggunakan Metode Search Ranker dan Attribute Selection InfoGainAttributeEval, diketahui bahwa Log lebih baik daripada Z-score, karena pada Log lebih cepat mendapat TP Rate yang tinggi dibanding Z-score yang membutuhkan beberapa atribut untuk mencapai TP Rate yang tinggi, sehingga Log dapat lebih cepat mendeteksi adanya intrusi dibanding Z-score. Hanya U2R yang berbeda, dimana Z-score mencapai TP Rate terlebih dahulu daripada Log.

### **6.1 Dokumentasi Laporan**

Untuk melihat dokumentasi rekaman presentasi dari kelompok satu dalam bentuk video bisa mengunjungi laman berikut <https://youtu.be/JWEZh5ahbZk> sedangkan untuk melihat laporan excel dari TP Rate secara mendetail yang didapatkan dari hasil Data Mining pada Weka selama proses pembuatan laporan ini bisa mengunjungi pada laman berikut <https://docs.google.com/spreadsheets/d/1zMsZdUssLOFZCL-kcfWbiCVQXSJhzfkfFxfE9-sCO8M/edit#gid=262433001>

## Daftar Pustaka

- 1-1 B.Setiawan, T. Ahmad, S. Djanali, Increasing Accuracy and Completeness of Intrusion Detection Model Using Fusion of Normalization, Feature Selection Method and Support Vector Machine, 2019
- 1-2 <https://medium.com/@samsudiney/apa-itu-machine-learning-a6a3fc28162a>, 2019
- 1-3 Yasir Hamid, Sugumaran Muthukumarasamy, Balasaraswathi Ranganathan, IDS Using Machine Learning -Current State of Art and Future Directions, 2016
- 1-4  
<https://ksnugroho.medium.com/confusion-matrix-untuk-evaluasi-model-pada-unsupervised-machine-learning-bc4b1ae9ae3f>, 2019
- 1-5  
<https://medium.com/@samsudiney/penjelasan-sederhana-tentang-apa-itu-svm-149fec72bd02>, 2019
- 1-6 <https://medium.com/@16611061/data-mining-d48b2389b61>, 2019
- 2-1. Dhak BS, Lade S (2012) An evolutionary approach to intrusion detection systems using genetic algorithms. *Int J Emerg Technol Adv Eng* 2(12):632–637
- 2-2. Saha S, Sairam A, Ekbal A (2012) Genetic algorithm combined with support vector machine for building an intrusion detection system. In: *International conference on advances in computing, communications and informatics*
- 2-3. Kachurka P, Golovko V (2011) Neural network approach to real time network intrusion detection and recognition. In: *Proceedings of the 6th IEEE international conference on intelligent data acquisition and advanced computing systems*, pp 15–17
- 2-4. Lakhinaet S, Joseph S, Verma B (2010) Feature reduction using principal component analysis for effective anomaly—based intrusion detection on NSL-KDD. *Int J Eng Sci Technol* 2:1790–1799
- 2-5. Rassam MA, Maarof MA (2014) Artificial immune network clustering approach for anomaly intrusion detection. *J Adv Inf Technol* 3(3):147–154
- 2-6. W. L. Al-Yaseen, Z. A. Othman, and M. Z. A. Nazri, “Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system”, *Expert Systems with Applications*, Vol. 67, pp. 296–303, 2017.

- 3-1 B.Setiawan, T. Ahmad, S. Djanali, Increasing Accuracy and Completeness of Intrusion Detection Model Using Fusion of Normalization, Feature Selection Method and Support Vector Machine, 2019
- 3-2 Z-Score: Definition, Formula, and Calculation, <https://www.statisticshowto.com/probability-and-statistics/z-score/>
- 3-3 S. Dinakaran, Dr. P. Ranjit Jeba Thangaiah, Role of Attribute Selection in Classification Algorithms, International Journal of Scientific & Engineering Research, Volume 4, Issue 6, June-2013
- 5-1 B.Setiawan, T. Ahmad, S. Djanali, Increasing Accuracy and Completeness of Intrusion Detection Model Using Fusion of Normalization, Feature Selection Method and Support Vector Machine, 2019
- 5-2 I. H. Witten, E. Frank, and M. A. Hall, Data Mining: Practical Machine Learning Tools and Techniques, Fourth Edi. Morgan Kaufmann, 2016.
- 5-3 C. Chang and C. Lin, "LIBSVM : A Library for Support Vector Machines", ACM Transactions on Intelligent Systems and Technology, Vol. 2, No. 3, p. 27, 2011.
- 5-4 Canadian-Institute, "NSL-KDD dataset," 2009. [Online]. Available: <https://www.unb.ca/cic/datasets/nsl.html>.
- 5-5 M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set", In Proc. of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, 2009.
- 6-1 Canadian-Institute, "NSL-KDD dataset," 2009. [Online]. Available: <https://www.unb.ca/cic/datasets/nsl.html>
- 6-2. Lakhinaet S, Joseph S, Verma B (2010) Feature reduction using principal component analysis for effective anomaly—based intrusion detection on NSL-KDD. Int J Eng Sci Technol 2:1790–1799