

Implementasi Backpropagation Termodifikasi dengan Conjugate Gradient Powell Beale untuk Deteksi Anomali pada Intrusion Detection System (IDS)

Untari Novia Wisesty¹, Ida Parwati², Adiwijaya³

*Fakultas Informatika, Universitas Telkom
Bandung, Indonesia*

¹untarinw@telkomuniversity.ac.id

²idaaparwati@gmail.com

³adiwijaya@telkomuniversity.ac.id

Abstract

In globalization era, technology and infrastructure are developed more rapidly. With increasingly open facilities of technology, turned out to cause new problems. To maintain the security systems, need to build the limitation of information access for users to maintain data integrity. Therefore, we need security system in computer network that can detect threats rapidly and accurately. One of the system that can maintain the security of computer networks is Intrusion Detection System(IDS). This research will detect anomalies in the IDS used Modified Backpropagation with Conjugate Gradient (CG) PowellBeale. Conjugate Gradient is one of the optimization method based on the search direction of orthogonal conjugate value. The use of CG in modified backpropagation is expected to help detect anomalies on IDS faster, because epoch that is used far less and the result of performance is better. This is evident of the result performance base on parameter F-Measure, the results of 92,33% for normal class, 62,38% for probe class, 42,06% for Dos class, and 9,14% for R2L class. This is prove that the system is capable to classify better with the number of epoch slightly compares with the standard backpropagation.

Keywords: anomaly detection, backpropagation, conjugate gradient powell beale, IDS.

Abstrak

Di era globalisasi ini teknologi beserta infrastrukturnya berkembang semakin pesat. Adanya fasilitas teknologi yang semakin terbuka ternyata menimbulkan masalah-masalah baru. Untuk menjaga sistem keamanan tersebut perlu adanya batasan-batasan akses informasi bagi para pengguna untuk menjaga integritas data. Oleh sebab itu diperlukannya sistem keamanan dalam jaringan komputer yang dapat mendeteksi acaman secara cepat serta akurat. Salah satu sistem yang dapat menjaga keamanan jaringan komputer adalah *Intrusion Detection System (IDS)*. Dalam penelitian ini untuk mendeteksi anomali pada IDS akan digunakan *Backpropagation* Termodifikasi dengan *Conjugate Gradient (CG) Powell Beale*. *Conjugate Gradient* merupakan salah satu metode optimasi yang arah pencariannya didasarkan pada arah konjugasi yang nilainya orthogonal. Penggunaan CG dalam memodifikasi *Backpropagation* diharapkan dapat membantu mendeteksi anomali pada IDS menjadi lebih cepat karena *epoch* yang digunakan jauh lebih sedikit dan hasil performansi sistem yang lebih baik. Hal tersebut terlihat dari hasil performansi sistem berdasarkan parameter *F-Measure* yaitu 92.33% untuk kelas normal, 62.38% untuk kelas Probe, 42.06% untuk kelas Dos dan 9.14% untuk kelas R2L. Terbukti bahwa sistem mampu mengklasifikasikan lebih baik dengan jumlah *epoch* yang sedikit dibandingkan dengan *backpropagation* standar.

Kata Kunci: *backpropagation*, conjugate gradient powell beale, deteksi anomali, IDS.

I. PENDAHULUAN

PENELITIAN pada kasus Intrusion Detection System (IDS) dengan menggunakan dataset KDDCUP 1999, telah banyak dilakukan. Algoritma yang digunakan diantaranya Backpropagation Termodifikasi dengan *Fletcher-Reeves*, *Scaled Conjugate*, *Metode Bayesian Network* dan *Clique Partitioning*. *Intrusion Detection System (IDS)* adalah suatu proses *monitoring* kejadian pada sistem komputer atau jaringan serta memberikan analisis terhadap aktivitas yang menyimpang maupun normal [4][5][14]. Dalam mendeteksi *intrusion* pada *IDS* mengklasifikasi *intrusion* menjadi dua aspek yaitu *misuse detection* dan *anomaly detection* [6][13]. *Misuse detection* mendeteksi *intrusion* dengan menggambarkannya sebagai suatu pola [6]. Cara kerja dari model ini yaitu mencocokkan atau mengenali pola-pola yang masuk dengan pola *intrusion* yang telah tersimpan di dalam database [14]. Sedangkan *anomaly detection* mendeteksi *intrusion* pada *host* atau *network* tanpa harus mengenali polanya terlebih dahulu sehingga ketika di deteksi terjadi sedikit penyimpangan maka akan dideteksi sebagai *anomaly* [6][14]. Dalam penelitian ini digunakan pendekatan metode klasifikasi *anomaly detection* dan data yang digunakan dalam penelitian yaitu *dataset KDDCUP 1999*. *Dataset KDDCUP1999* merupakan data yang diambil dari hasil evaluasi DARPA'98 tentang *Intrusion Detection Evaluation Program* yang bekerjasama dengan MIT Licoln Labs [7]. Di dalam *dataset* telah disediakan data yang dapat digunakan untuk *training*, yaitu 494.021 dan data *testing* 311.029. Kedua data tersebut terdiri atas kelas normal dan kelas intrusi yang jumlahnya tidak seimbang (*imbalanced data*). Pada data *training* jumlah data normal terdiri atas 97.278 dan 396.744 data intrusi. Sedangkan pada data *testing* terdiri atas 60.593 data normal dan 250.436 data intrusi [7]. Data tersebut terdiri dari 41 dengan 34 data bertipe numerik dan 7 data bertipe simbolik.

Dataset KDDCUP 199 memiliki 4 macam kategori serangan, yaitu *Denial of Service Attack (DoS)*, *User to Root Attack (U2R)*, *Remote to Local (R2L)* dan *Probing Attack(Prob)*. DoS merupakan tipe penyerangan yang membuat komputasi atau sumber daya memori menjadi lebih sibuk yang disebabkan oleh banyaknya permintaan akses [10]. U2R merupakan tipe penyerangan dimulai dengan cara mengakses sitem melalui *user account* secara normal untuk mendapatkan password. Setelah mendapatkan hal tersebut memungkinkan penyerang mendapatkan *root access* menuju sistem [10]. R2L merupakan tipe penyerangan yang memiliki kemampuan untuk merusak lalu mendapatkan hak akses lokal sebagai pengguna [10]. Probe merupakan tipe penyerangan yang mencari informasi tentang jaringan komputer dari target penyerangan dengan tujuan untuk terhindar dari *security control* [10]. Metode klasifikasi *anomaly detection* dapat dikerjakan dengan salah satu algoritma pelatihan yaitu algoritma jaringan syaraf tiruan (JST). Jaringan Syaraf Tiruan adalah suatu algoritma yang arsitekturnya dimodelkan seperti cara kerja jaringan syaraf manusia (otak). Sebagian besar JST memiliki aturan pelatihan (*training rule*) untuk menemukan bobot-bobot koneksi berdasarkan data latih yang ada sehingga dengan kata lain JST merupakan proses *learning* [16]. Di dalam JST terdapat 2 macam model arsitektur, yaitu Single Layer Perceptron (SLP) dan Multi Layer Perceptron(MLP). Salah satu metode pembelajaran MPL yang sering digunakan yaitu *backpropagation*.

Algoritma *backpropagation* yang nantinya akan digunakan dalam mendeteksi anomali pada IDS. Algoritma *backpropagation* merupakan algoritma pembelajaran yang terawasi (*supervised learning*). Namun algoritma *backpropagation* ini memiliki kekurangan, yaitu memerlukan waktu yang lama dalam proses pelatihannya. Sehingga untuk mempercepat proses pelatihan digunakan metode *Conjugate Gradient Powell Beale* serta metode *Line Search* yang di harapkan dapat mempercepat proses pelatihan dan hasil yang diperoleh menjadi lebih akurat.

II. PENELUSURAN PUSTAKA

A. Algoritma Pelatihan Backpropagation dengan Conjugate Gradient Powell-Beale

Perhitungan yang dilakukan Algoritma *Backpropagation* membutuhkan waktu yang lama dalam proses *training*, maka digunakan *Conjugate Gradient-Powell Beale* untuk mempercepat kinerja dari Algoritma *Backpropagation*. Metode *Conjugate Gradient(CG)* merupakan salah satu metode optimasi yang arah

pencariannya di dasarkan pada arah konjugasi yang nilainya ortogonal. CG menggunakan vektor tidak nol yang ortogonal dan bebas linear. Dua vektor d_i dan d_j dikatakan ortogonal jika hasil perkalian dalam bernilai nol [1][2][9].

$$d_i^T d_j = 0 \quad (1)$$

Metode *Conjugate gradient* merupakan metode yang digunakan untuk meminimumkan fungsi turunan yang dapat dihitung dengan pendekatan w_{k+1} secara iteratif berdasarkan:

$$w_{k+1} = w_k + \alpha_k d_k \quad (2)$$

$$d_{k+1} = -g_{k+1} + \beta_k d_k \quad (3)$$

dimana nilai α dan β merupakan parameter momentum (untuk menghindari konvergensi lokal). Sehingga dapat dituliskan langkah-langkah algoritma pelatihan *backpropagation* dengan algoritma *conjugate gradient powell beale* sebagai berikut:

1. Menentukan matriks pola masukan (P) dan matriks target (T).
2. Inisialisasi arsitektur jaringan, nilai batasan MSE dan batasan jumlah *epoch* sebagai kondisi pemberhentian. Menentukan *line search*, membangkitkan bobot-bobot (w) dan bias (b) dengan bilangan acak kecil.
3. Jika kondisi pemberhentian belum terpenuhi maka lakukan langkah 3-14.

▪ Propagasi Maju

4. Tiap unit input ($X_i, i=1,2,3,..n$) menerima sinyal x_i dan meneruskannya ke semua unit pada *hidden layer* ($Z_j, j=1,2,3,..p$). Setiap unit pada *hidden layer* dijumlahkan ke setiap unit input beserta bobot dan biasnya.

$$z_{net_j} = b1_j + \sum_{i=1}^n x_i v_{ji}, \quad (4)$$

$$z_j = f(z_{net_j}) = \frac{1}{1+e^{-z_{net_j}}}. \quad (5)$$

5. Selanjutnya hitung keluaran *output layer* $y_k (k=1,2,3,..m)$

$$y_{net_k} = b2_k + \sum_{j=1}^p z_j w_{kj}, \quad (6)$$

$$y_k = f(y_{net_k}) = \frac{1}{1+e^{-y_{net_k}}}. \quad (7)$$

▪ Propagasi Mundur

6. Kemudian hitung faktor kesalahan di unit output berdasarkan perbedaan (*error*) nilai aktual dan nilai prediksi (output dari unit output)

$$\delta_k = (t_k - y_k) f'(y_{net_k}) = (t_k - y_k) y_k (1 - y_k) \quad (8)$$

7. Hitung faktor kesalahan pada unit *hidden layer* berdasarkan faktor kesalahan sebelumnya

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj}, \quad (9)$$

$$\delta_j = \delta_{net_j} f'(z_{net_j}) = \delta_{net_j} (1 - z_j) \quad (10)$$

8. Hitung gradient di unit output berdasarkan fungsi objektif yang sudah ditentukan

$$g_{k+1} = \frac{1}{N} \sum_{n=1}^p \delta_{nk} y_{nk} \quad (11)$$

9. Hitung gradient di unit *hidden layer*

$$g_{j+1} = \frac{1}{N} \sum_{n=1}^p \delta_{nj} z_{nj} \quad (12)$$

10. Reset arah *direction* ke negatif gradient pada saat :

$$|g_{k+1} * g_k| \geq 0.2 \|g_k\|^2 \quad (13)$$

11. Hitung parameter β untuk semua neuron di unit hidden layer dan unit *output layer*. Parameter β untuk *conjugate gradient Powell-Beale* dapat dihitung dengan persamaan :

$$\beta_{k+1} = \frac{g_{k+1}^T (g_{k+1} - g_k)}{d_k^T (g_{k+1} - g_k)}, \quad (14)$$

dimana: β_{k+1} = β merupakan nilai parameter pada iterasi saat ini.
 g_{k+1} = gradient pada iterasi saat ini.
 g_k = gradient pada iterasi sebelumnya.
 d_k = *direction* pada iterasi sebelumnya

12. Hitung *direction* untuk semua *neuron* pada *hidden layer* dan *output layer*.

$$d_{k+1} = -g_{k+1} + \beta_k d_k \quad (15)$$

dimana : d_{k+1} = *direction* pada iterasi saat ini.
 g_{k+1} = *gradient* pada iterasi saat ini.
 β_k = β merupakan nilai parameter pada iterasi sebelumnya.
 d_k = *direction* pada iterasi sebelumnya.

Untuk inisial *direction*:

$$d_1 = -g_1 \quad (16)$$

13. Hitung parameter α untuk semua *neuron* pada *hidden layer* dan *output layer* α . Parameter α dapat di cari dengan teknik *line search* yang digunakan untuk meminimumkan kinerja selama pencarian [2]. Teknik *line search* tersebut, seperti :

▪ **Perubahan Bobot**

14. *Update* bobot dapat dilakukan dengan persamaan berikut

$$w_{t+1} = w_t + \alpha_{t+1} d_t + 1, \quad (17)$$

dimana

w_{t+1} = *update* bobot
 w_t = bobot sebelumnya
 α_{t+1} = nilai alfa saat ini
 d_t = *direction* pada iterasi saat ini

B. Pengukuran Performansi

▪ *Training*

Pengukuran performansi yang digunakan pada saat proses *training* yaitu menggunakan *Mean Square Error* (MSE). Persamaan tersebut dapat dituliskan sebagai berikut:

$$MSE = \frac{1}{n} \sum_{i=1}^n (x_i - f_i)^2 \quad (18)$$

dimana:

n = jumlah data
 x_i = data aktual
 f_i = data target

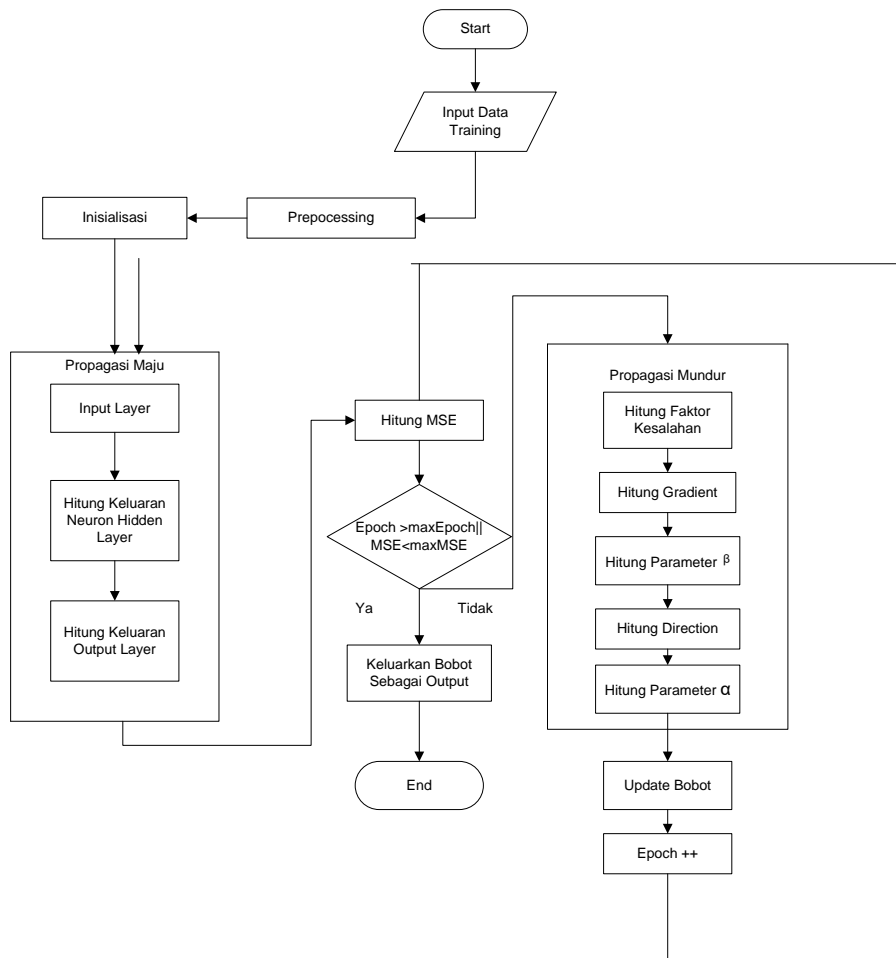
- *Testing*

Ukuran performansi sistem pada tahap pengujian digunakan *confusion matrix* [14].

III. METODE PENELITIAN

A. Perancangan Sistem

Secara umum perancangan sistem dibuat untuk mendeteksi anomali dengan pada data KDDCUP1999 tentang *Intrusion Detecton System(IDS)* dengan menggunakan algoritma *backpropagation* termodifikasi. Sistem ini terbagi menjadi beberapa tahapan proses seperti *preprocessing* data, pelatihan menggunakan *backpropagation* termodifikasi dengan *powell beale*, pengujian sistem dengan menghasilkan akurasi, *precision*, *recall* dan *f-measure* masing-masing kelas. *Preprocessing* data ini merupakan tahap awal sebelum data memasuki tahap pelatihan dan pengujian. *Preprocessing* ini dilakukan karena *dataset* KDDCUP 1999 memiliki beberapa atribut yang bertipe simbolik sehingga perlu dilakukan *preprocessing* untuk mengubahnya menjadi tipe data numerik. Setelah data selesai di *preprocessing*, data akan dilakukan normalisasi. Normalisasi ini dilakukan karena *dataset* KDDCUP 1999 memiliki *range* yang sangat beragam . Perancangan sistem pada penelitian ini terdapat 2 macam yaitu, proses pelatihan dan pengujian yang terdapat pada Gambar 1.



Gambar 1. Flowchart Tahap Pelatihan

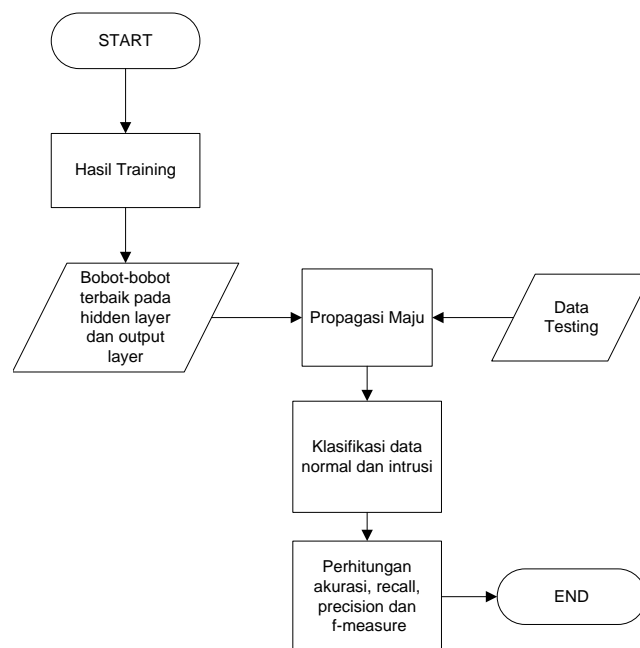
▪ Tahap Pelatihan

Tahapan-tahap pelatihan dengan menggunakan *backpropagation* termodifikasi, yaitu:

1. Menentukan inputan, target dan kriteria pemberhentian(maksimum *epoch* dan MSE).
2. Selanjutnya data input akan memasuki tahapan propagasi maju dengan memasuki 2 layer yaitu, *hidden layer* dan *output layer*. Di dalam propagasi maju ini dilakukan perhitungan yang kemudian menghasilkan *output* berupa bobot. Bobot yang dikeluarkan oleh *output layer* akan dilakukan perhitungan error, ketika error yang dikeluarkan masih bernilai tinggi maka dilakukan perhitungan ulang atau yang disebut tahapan propagasi mundur.
3. Pada tahapan propagasi mundur, perhitungan akan dimulai dari menghitung faktor kesalahan yang terjadi pada perhitungan sebelumnya. Di tahapan propagasi mundur ini *conjugate gradient powell belae* di terapkan. Metode ini akan memperbaiki faktor kesalahan sebelumnya dengan cara menghitung gradien, *learning rate* dan *direction*.
4. Setelah perhitungan akan dihasilkan arsitektur dengan bobot terbaik.

▪ Tahap Pengujian

Hasil dari tahap pelatihan yaitu berupa arsitektur dengan bobot terbaik. Arsitektur dengan bobot terbaik ini akan uji dengan data *testing*. Pada tahapan pengujian perhitungan hanya memasuki tahap propagasi maju saja. Selanjutnya sistem akan menghasilkan hasil klasifikasi dan performansi berupa akurasi, *recall*, *precision* dan *f-measure*. Diagram tahap pengujian tersaji dalam gambar 2.



Gambar 2. Tahap Pengujian

B. Dataset Training dan Testing

Pada proses pelatihan dan pengujian digunakan *dataset* dengan proporsi data yang berbeda. Pada penelitian ini terdapat 5 skenario pengujian. Pada skenario 1 dan 2 akan diterapkan metode *undersampling* dan *oversampling* untuk pemilihan data *training*. Metode *undersampling* dan *oversampling* merupakan metode yang dapat digunakan dalam untuk mengatasi masalah dalam *imbalanced* data [12]. Metode *undersampling* dilakukan dengan cara mengurangi jumlah data kelas mayor agar memiliki jumlah yang sama dengan data kelas

minornya [12]. Sedangkan *oversampling* dilakukan dengan cara menggandakan jumlah data kelas minor agar jumlahnya sama dengan data kelas mayor [12]. Proses *undersampling* dan *oversampling* ini dilakukan dengan menggunakan aplikasi WEKA dengan mengambil data sample sebanyak 2000 data.

Sedangkan untuk skenario 3 dalam pemilihan data *training* tidak menerapkan metode *undersampling* dan *oversampling*. Data *training* yang digunakan pada skenario 3 dilakukan pemilihan secara manual dan random. Pada skenario 4 akan dilakukan pelatihan dan pengujian secara *binary search* dengan untuk melihat hasil performansi masing-masing kelas jika dilakukan pelatihan dan pengujian secara terpisah. Sedangkan pada skenario 5 akan dilakukan pemilihan skenario terbaik dari skenario pengujian 1, 2 dan 3. Skenario yang terbaik tersebut selanjutnya akan dilakukan perbandingan performansi antara *backpropagation* termodifikasi dengan *backpropagation* standar. Pada masing-masing skenario akan digunakan jumlah *hidden neuron* yang dikombinasikan, yaitu , 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, dan 34 sehingga masing-masing skenario akan melewati tahap pelatihan sebanyak 17 kali. Berikut distribusi *dataset* yang akan dilakukan pada proses pelatihan dan pengujian:

TABEL I
DISTRIBUSI DATA TRAINING

Skenario	Dos	Probe	R2L	Normal	Total
1	186	75	77	906	1244
2	1000	370	91	3000	4461
3	7400	1750	800	6050	16000
4	1000	1000	1000	4000	7000

TABEL II
DISTRIBUSI DATA TESTING

	Normal	R2L	Probe	Dos	Total
Data Testing	63764	132	3062	3557	79364

IV. HASIL DAN DISKUSI

A. Analisis Pengaruh Parameter-Parameter Terhadap Hasil Pelatihan dan Pengujian

Dalam beberapa proses pelatihan yang dilakukan ternyata terdapat parameter-parameter yang berpengaruh terhadap hasil pelatihan maupun pengujian. Diantaranya yaitu proporsi data intrusi dan normal, jumlah neuron pada hidden layer. Berikut merupakan hasil pelatihan terbaik pada skenario 1 dan 2, yaitu:

TABEL III
HASIL PELATIHAN

Skenario 1			Skenario 2			Skenario 3		
Jumlah Neuron Hidden Layer	Epoch	MSE Training	Jumlah Neuron Hidden Layer	Epoch	MSE Training	Jumlah Neuron Hidden Layer	Epoch	MSE Training
16	139	1.70E-03	26	349	1.56E-04	20	381	2.50E-03

Pada Tabel 3 menunjukkan perbedaan hasil MSE *training* dan epoch untuk masing-masing skenario. Terlihat bahwa terlihat perbedaan hasil disebabkan oleh berbagai parameter seperti halnya *learning rate*, jumlah neuron *hidden layer* dan lain-lain yang bersifat *trial-error*. Berdasarkan tabel tersebut terlihat bahwa MSE training skenario 2 jauh lebih minimum dari skenario1 dan 3 yaitu, 1.56E-04 dengan jumlah neuron *hidden layer*

sebanyak 26. Selanjutnya arsitektur terbaik yang telah terbentuk akan masuk pada tahap pengujian. Berikut hasil pengujian, yaitu:

TABEL IV
 HASIL PENGUJIAN SKENARIO 1 DAN 2 DENGAN DATA TESTING 79364

Skenario	Kelas	Jumlah Terdeteksi	Recall (%)	Precision (%)	F-Measure (%)	Jumlah Benar
1	Normal	61704	91.50	96.77	94.06	67196
	R2L	57	23.75	2.85	5.09	
	Probe	2760	38.63	83.51	52.83	
	Dos	2675	59.34	25.98	36.24	
2	Normal	63182	81.31	99.09	89.32	63930
	R2L	60	13.33	3.00	4.90	
	Probe	488	73.72	14.77	24.60	
	Dos	200	36.56	1.94	3.69	
3	Normal	59600	91.21	93.47	92.33	66351
	R2L	132	14.86	6.60	9.14%	
	Probe	3062	47.02	92.65	62.38	
	Dos	3557	53.75	34.55	42.06	

B. Analisis Performansi Berdasarkan Parameter Akurasi, Recall, Precision dan F-Measure

Berdasarkan Tabel 4 jika dilihat dari segi performansi *recall*, *precision* dan *f-measure* pada masing-masing kelas di ketiga skenario memiliki hasil yang berbeda-beda. Nilai *recall* tertinggi dengan nilai mencapai 91.50% terdapat pada kelas normal pada skenario 1 dan untuk kelas intrusi yang paling tertinggi terdapat pada kelas Dos dengan nilai mencapai 59.34% pada skenario 1. Tingginya nilai *recall* yang diperoleh dapat menunjukkan tingkat keberhasilan suatu sistem dalam menemukan kembali suatu informasi yang relevan antara data diuji dengan data aktual.

Pada ketiga skenario yang telah diuji nilai *precision* tertinggi terdapat pada kelas normal dengan nilai 99.09% yang terdapat pada skenario 2 dan untuk kelas intrusi nilai *precision* tertinggi terdapat pada kelas probe dengan nilai 92.65% di skenario 3. Nilai *precision* yang tinggi dapat menyatakan nilai kedekatan antara hasil keluaran dari sistem yang telah terbentuk dengan nilai aktual. Nilai *precision* dan *recall* ternyata memiliki hubungan *trade-off* yang terbalik. Ketika ingin meningkatkan nilai *precision* biasanya dapat dilakukan dengan menurunkan nilai *recall*, begitu pula sebaliknya [15]. Nilai *F-Measure* tertinggi terdapat pada kelas normal di skenario 1 dengan nilai 94.06% dan untuk kelas intrusi terdapat pada kelas probe di skenario 3 dengan nilai 62.38%. Parameter *f-measure* ini menggabungkan antara nilai *precision* dan *recall* menjadi deret harmonik [15]. Berdasarkan ketiga skenario tersebut dapat disimpulkan bahwa baik atau tidaknya sistem dalam mengklasifikasikan ini tergantung pada pola atau arsitektur yang dibentuk dari hasil pelatihan. Hal tersebut terlihat dari ketiga skenario yang telah dilatih dengan jumlah data *training* yang berbeda-beda. Berdasarkan hasil pengujian dari ketiga parameter yang telah digunakan parameter *f-measure* yang akan dijadikan sebagai acuan baik atau tidaknya sistem klasifikasi yang telah terbentuk.

Hal tersebut dilakukan karena data yang digunakan untuk pelatihan merupakan data yang *imbalanced* dan salah satu parameter yang cocok yaitu *f-measure*. Secara umum nilai *f-measure* yang diperoleh dari ketiga skenario yang diuji hasil yang paling baik berada pada skenario 3 (lihat Tabel 4). Hal tersebut dapat terlihat dari nilai rata-rata nilai *f-measure* yang diperoleh skenario 3 mencapai 49.22% sedangkan skenario 1 dan 2 yang diterapkan metode *undersampling* dan *oversampling* ternyata tidak menunjukkan hasil yang begitu baik, yaitu dengan hasil 47.06 % pada skenario 1 dan 30.63% pada skenario 2 . Sehingga dapat dikatakan bahwa sistem

klasifikasi yang dibuat oleh skenario 3 merupakan sistem terbaik berdasarkan hasil pengujian yang telah dilakukan.

C. Analisis Pengujian Pada Masing-masing Kelas Intrusi

Berdasarkan hasil pengujian yang telah dilakukan pada skenario 1, 2 dan 3 memiliki perbedaan jumlah yang terdeteksi pada masing-masing kelas. Oleh karena itu pada skenario 4 akan dilakukan pengujian untuk masing-masing kelas intrusi dengan tujuan untuk melihat hasil performansi yang dihasilkan pada masing-masing kelas sudah cukup baik atau belum. Data *training* dan data *testing* yang digunakan berjumlah sama antara data intrusi dan data normal. Berikut hasil pelatihan serta pengujian yang dilakukan:

TABEL V
HASIL PELATIHAN PADA MASING-MASING KELAS INTRUSI

Skenario	Jumlah Neuron Hidden Layer	Epoch	MSE Training	Jumlah Pola	Jumlah Terdeteksi	Akurasi
Hasil Training Kelas Dos dengan Normal	14	347	4.12E-05	1800	1800	100.00%
Hasil Training Kelas Probe dan Normal	8	210	3.43E-04	1800	1800	100.00%
Hasil Training Kelas R2L dan Normal	18	163	8.50E-03	1800	1780	98.89%

Berdasarkan hasil pelatihan yang dilakukan pada masing-masing kelas intrusi dengan jumlah proporsi jumlah data normal dan intrusi yang sama, terlihat bahwa hasil yang diperoleh sudah sangat baik. MSE masing-masing kelas pun menunjukkan error yang kecil bahkan pada kelas intrusi dos dan probe akurasi *training* mencapai 100%. Selanjutnya arsitektur terbaik yang telah terbentuk oleh masing-masing kelas akan dilakukan pengujian dengan data yang jumlahnya sama dengan data *training*-nya. Berikut hasil pengujianya:

TABEL VI
HASIL PENGUJIAN MASING-MASING KELAS

		Jumlah	Recall	Precision	F-Measure	Jumlah Benar	Jumlah Pola	Akurasi Testing
Hasil Testing	Normal	898	74.58%	99.78%	85.36%	1492	1800	82.89%
	DOS	881	87.49%	97.89%	92.40%			
Hasil Testing	Normal	881	87.49%	97.89%	92.40%	1655	1800	91.94%
	Probe	774	97.60%	86.00%	91.44%			
Hasil Testing	Normal	647	98.48%	71.89%	83.11%	1537	1800	85.39%
	R2L	890	77.87%	98.89%	87.13%			

Pada tabel hasil pengujian yang ditunjukkan oleh Tabel 6, terlihat bahwa akurasi yang diperoleh masing-masing kelas mencapai nilai lebih dari 80%. Hal tersebut dapat menyatakan bahwa sistem yang telah terbentuk sudah cukup baik dalam melakukan klasifikasi. Hanya saja pengaruh banyaknya proporsi jumlah data dan intrusi sangat lah berpengaruh. Selain itu pembagian kelas klasifikasi yang banyak juga dapat mempengaruhi hasil akurasi.

D. Analisis Performansi Berdasarkan Parameter Akurasi, Recall, Precision dan F-Measure

Untuk melihat perbandingan performansi antara *backpropagation* yang termodifikasi *conjugate gradient* powell belae dengan *backpropagation* standar maka akan dilakukan pelatihan dan pengujian. Berikut perbandingan hasil pelatihannya:

TABEL VII
 PERBANDINGAN HASIL PELATIHAN BACKPROPAGATION TERMODIFIKASI DENGAN BACKPROPAGATION STANDAR

Jumlah Neuron Hidden Layer	Backpropagation Conjugate Gradient		Jumlah Neuron Hidden Layer	Backpropagation Standar	
	Epoch	MSE		Epoch	MSE
20	381	2.50E-03	34	1001	3.93E-02

Pada Tabel 6 terlihat perbedaan *epoch* yang cukup signifikan dalam mencapai solusi yang optimum. Pelatihan menggunakan *backpropagation* standar memerlukan usaha yang lebih keras. Hal tersebut dapat dilihat dari *epoch* yang sangat besar atau mencapai batas maksimum *epoch* yang telah ditentukan. Berbeda dengan pelatihan *backpropagation* dengan *conjugate gradient* powell beale yang hanya memerlukan *epoch* yang sedikit untuk mencapai solusi yang optimum. Sedangkan jika dilihat dari nilai MSE *training* terdapat pada *backpropagation* termodifikasi dengan MSE 2.50E-03, jumlah neuron *hidden layer* sebanyak 20 serta *epoch* 381. Berikut tabel perbandingan waktu pada saat pelatihan antara *backpropagation* standar dan *backpropagation* termodifikasi. Pada tabel dibawah ini akan memperlihatkan perbandingan antara *backpropagation* standar dengan *backpropagation* termodifikasi dengan *conjugate gradient* powell beale dalam mendeteksi atau mengklasifikasi serangan, yaitu sebagai berikut:

TABEL VIII
 PERBANDINGAN HASIL PENGUJIAN CG POWELL BEALE DENGAN BACKPROPAGATIONN STANDAR PADA DATA TESTING 79364

Skenario	Kelas	Jumlah	Recall (%)	Precision (%)	F-Measure (%)	Jumlah Benar
Backpropagation Standar	Normal	61299	96.13%	90.10%	89.98%	62897
	R2L	55	4.31%	2.75%	3.36%	
	Probe	746	15.50%	22.57%	18.38%	
	Dos	797	81.83%	7.74%	14.15%	
Backpropagation Termodifikasi	Normal	59600	91.21%	93.47%	92.33%	66351
	R2L	132	14.86%	6.60%	9.14%	
	Probe	3062	47.02%	92.65%	62.38%	
	Dos	3557	53.75%	34.55%	42.06%	

Berdasarkan Tabel 8 hasil pengujian dengan menggunakan *backpropagation* termodifikasi memperoleh hasil yang lebih baik dan akurat dibandingkan dengan *backpropagation* standar. Hal tersebut dapat terlihat dari jumlah data yang dapat dideteksi secara benar oleh *backpropagation* termodifikasi yaitu dengan jumlah 66351 sedangkan *backpropagation* standar berjumlah 62897. Selain itu juga hasil pengujian berdasarkan parameter *f-measure*, hampir semua nilai tertinggi untuk semua kelas terdapat pada *backpropagation* termodifikasi. Sehingga dapat disimpulkan bahwa dengan menggunakan algoritma *backpropagation* termodifikasi dapat meningkatkan hasil performansi.

Perbedaan yang lain terlihat pada jumlah *epoch* yang diperlukan pada saat pelatihan. *Backpropagation* termodifikasi memerlukan *epoch* yang lebih kecil untuk mencapai solusi terbaik atau optimum berbeda jauh jika dibandingkan dengan *epoch* *backpropagation* standar. Maka dapat disimpulkan bahwa pelatihan menggunakan algoritma *backpropagation* termodifikasi dengan *conjugate gradient* mampu mempercepat dalam tahap pelatihan untuk pencarian solusi yang konvergen. Hal ini disebabkan penambahan metode *conjugate gradient* yang berada pada tahap propagasi mundur. Propagasi mundur ini yang akan menghitung α (*learning rate*), β (parameter saat ini) dan arah pencarian serta lebar langkah.

Nilai-nilai tersebut yang akan berpengaruh pada *update* bobot dalam mendapatkan solusi yang optimum, sehingga *epoch* yang diperlukan menjadi lebih sedikit. Sedangkan pada *backpropagation* standar terdapat parameter *learning rate* yang sangat berpengaruh terhadap tahap pembelajaran. *Learning rate* yang digunakan pada *backpropagation* standar ini nilainya bersifat tetap pada setiap iterasi sesuai nilai yang telah diset di awal. Semakin besar nilai *learning rate* ini akan menyebabkan pembelajaran menjadi tidak stabil namun jika diset terlalu kecil waktu pembelajaran akan semakin lama [8]. Sehingga penentuan *learning rate* ini menjadi hal yang paling penting pada *backpropagation* standar.

V. KESIMPULAN

Algoritma *Backpropagation* Termodifikasi dengan *Conjugate Gradient Powell Beale* terbukti dapat diimplementasikan untuk mendeteksi anomali pada IDS dan mampu mempercepat proses pelatihan dalam mencari bobot terbaik. Hal tersebut di tunjukan pada hasil pelatihan pada *backpropagation* termodifikasi dengan nilai MSE yang lebih minimum yaitu $2.50E-03$ dan *epoch* sebanyak 381. Sedangkan *backpropagation* standar memperoleh nilai MSE $3.93E-02$ dan *epoch* yang mencapai batas maksimum yaitu 1001. Dalam menganalisis hasil performansi untuk Algoritma *Backpropagation* Termodifikasi dengan *Conjugate Gradient Powell Beale* digunakan beberapa parameter yaitu *recall*, *precision* dan *f-measure*. Namun karena data yang digunakan merupakan *imbalanced data* sehingga untuk melihat hasil performansi terbaik dapat dilihat dari nilai *f-measure*. Berdasarkan hasil pengujian, sistem klasifikasi yang terbentuk cukup baik, dimana nilai *F-Measure* tertinggi 62.38% pada kelas intrusi Probe dan 92.33% pada kelas Normal. Berdasarkan skenario pelatihan dan pengujian yang telah dilakukan terlihat bahwa pemilihan proporsi data intrusi dengan normal sangat mempengaruhi terhadap hasil *training*. Selain itu faktor yang dapat mempengaruhi pada pelatihan *backpropagation* termodifikasi yaitu jumlah neuron pada *hidden layer* dan fungsi line search.

DAFTAR PUSTAKA

- [1] Adiwijaya, T.A.B. Wirayuda, U.N. Wisesty, Z.K.A. Baizal, U. Haryoko. "An improvement of Backpropagation Performance by Using Conjugate Gradient on Forecasting of Air Temperature and Humidity in Indonesia".(2013). Far East Journal of Mathematical Sciences (FJMS), (Part I), 57-67.
- [2] Adiwijaya, U.N. Wisesty, F. Nhita, Some Line Search Techniques on the Modified Backpropagation for Forecasting of Weather Data in Indonesia. (2014). Far East Journal of Mathematical Sciences 86:2 pp. 391 -396.
- [3] Akbar, Shaik. Rao, K. Nageswara. Chandulal, J.A. 2010. "Intrusion Detection System Methodologies Based on Data Analysis". International Journal of Computer Applications Volume 5.
- [4] Beale, Jay, Foster, James C, and Posluns, Jeffrey.2003. "Snort 2.0 Intrusion Detection". Rockland, MA, USA. Syngress Publishing.
- [5] Debar,H, Dacier Marc, and Wespi Andrea.(1999)."Towards a taxonomy of intrusion-detection systems". Computer Networks, 31(8):805-822.
- [6] Han, Jiawai, Micheline Kamber. 2006. "Data mining : Concepts and Technique Second Edition" .University of Illinois at Urbana-Champaign :Micheline Kamber.
- [7] KDD Cup 1999. Available on: <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>
- [8] Kusumadewi,Sri,2004."Membangun Jaringan Syaraf Tiruan(Menggunakan MATLAB & Excel Link"Yogyakarta, Graha Ilmu.
- [9] M.Shidiq,Azis, Adiwijaya, B.Munajat.(2014)."Deteksi Anomaly pada Intrusion Detection System Menggunakan Metode Backpropagation Termodifikasi", Seminar Nasional Ilmu Komputasi dan Teknik Informatika.
- [10] M.Tavallae, E.Bagheri, W.Lui, and A.A.Ghorbani.2009."A Detailed Analysis of The KDD CUP 2009 Data Set". Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2009).
- [11] MATLAB Toolbox R2013a, TRAINCGB Conjugate Gradient Backpropagation with Powell-Beale Restart.
- [12] Melissa, Ira, Oetama ,S ,Raymond.(2013)." Analisis Data Pembayaran Kredit Nasabah Bank Menggunakan Metode Data Mining". ULTIMA InfoSys, Vol. IV, No. 1 Juni2013.
- [13] Nastaiinullah,Adiwijaya, Kurniati P A.(2014)." Anomaly Detection on Intrusion Detection System Using CLIQUE Partitioning". International Conference on Onformation and Communication Trchnology(ICoICT).
- [14] Oktavia A.M, Adiwijaya, Angelina P.K.2012. "Anomaly Intrusion Detection Sytem menggunakan metode bayes". IT Telkom Bandung.
- [15] Prasetyo,Eko.2014."DataMining, Mengolah Data Menjadi Informasi Menggunakan MATLAB". Yogyakarta. Andipublisher.
- [16] Suyanto, 2008, "Soft Computing: Membangun Mesin Ber-IQ Tinggi" ,Bandung.Informatika.
- [17] Tan, Pang -Ning, Steinbach, Michael, Kumar, Vipin."Introduction To Data Mining ".
- [18] Tsai, Jeffrey J. P. 2011.Intrusion Detection : A Machine Learning Approach. River Edge, NJ, USA: World Scientific & Imperial College Press.

