

**APLIKASI JARINGAN SARAF TIRUAN *BACKPROPAGATION*
UNTUK MEMPREDIKSI PENYAKIT THT
DI RUMAH SAKIT MARDI RAHAYU KUDUS**

Arif Jumarwanto
Rudy Hartanto, Dhidik Prastiyanto

ABSTRACT

Artificial neural network (ANN) is a modern computing paradigm. That can be used for the pattern recognition and other. Backpropagation is artificial neural network which using hidden layer addition. Computation of artificial neural network through some certain step like training phase and examination. After both the step reached, so a neural network capable to recognize pattern to be entered will be found.

The purpose of this research is simulation of artificial neural network that capable to pattern recognition from output of electrocardiogram by helped of MATLAB program. Input of result electrocardiogram record, then input of data can be normalization after that data can be processed by backpropagation computing with two step (training phase and examination phase). Output of ANN is like explaining condition of patient is normal, rhinitis kronis or epistaksis.

Keyword : *rhinitis kronis, epistaksis, artificial neural network.*

I. PENDAHULUAN

1.1. Latar Belakang Masalah

Salah satu permasalahan yang ada di masyarakat adalah semakin banyaknya jenis penyakit yang bermunculan. Salah satu jenis penyakit yang sering dijumpai di masyarakat adalah penyakit THT. Hal ini dikarenakan banyak penyakit sistematis yang bermanifestasi di daerah telinga, hidung dan tenggorokan. Penelitian ini dikhususkan untuk memprediksi jenis penyakit THT pada bagian hidung, yaitu *rinitis kronis* dan *Epistaksis*.

Nafas manusia dimulai dari lubang hidung. Usaha bernafas menghantarkan udara lewat saluran pernafasan atas dan bawah kepada alveoli paru dalam volume, tekanan kelembaban, suhu dan kebersihan yang cukup untuk menjamin suatu kondisi ambil oksigen yang optimal, dan pada proses sebaliknya juga menjamin proses eliminasi karbondioksida yang optimal, yang diangkut ke alveoli lewat aliran darah.

Pada skripsi ini jaringan syaraf tiruan *backpropagation* yang akan dicoba diterapkan untuk diagnosis awal suatu penyakit yang berkembang di masyarakat yaitu THT bagian hidung yaitu *Rinitis kronis* dan *Epistaksis*

berdasarkan gejala klinis. Jaringan syaraf tiruan *backpropagation* merupakan topologi yang cukup populer dan paling banyak dipakai untuk berbagai aplikasi terutama pengenalan pola. Jaringan syaraf tiruan *backpropagation* adalah jenis *supervised learning* dimana output dari jaringan dibandingkan dengan target yang diharapkan sehingga diperoleh *error* output, kemudian *error* ini dipropagasikan balik untuk memperbaiki bobot jaringan dalam rangka meminimasi *error*. Pada sistem prediksi penyakit THT *Rinitis kronis* dan *Epistaksis* berbasis jaringan syaraf tiruan keberhasilan tergantung pada data-data yang telah diberikan pada fase pelatihan.

1.2. Rumusan Masalah

Bertitik tolak dari latar belakang masalah tersebut di atas, skripsi yang akan dibuat dapat dirumuskan sebagai berikut :

1. Bagaimana merancang arsitektur jaringan syaraf tiruan *backpropagation* sehingga dapat memprediksi penyakit THT rinitis kronis dan epistaksis ?
2. Bagaimana tingkat akurasi yang dihasilkan dari jaringan syaraf tiruan tersebut terkait dengan pengenalan pola dari

penyakit THT rinitis kronis dan epistaksis ?

1.3. Batasan Masalah

Batasan masalah dalam penulisan skripsi ini adalah:

1. Penyakit yang dibahas dalam penelitian ini dikhususkan untuk penyakit hidung yaitu *Rinitis Kronis dan epistaksis*
2. Jaringan syaraf yang digunakan adalah *backpropagation*
3. Program dibuat dengan matlab 7.04

1.4. Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah

- 1 Merancang arsitektur jaringan syaraf tiruan yang dapat memprediksi penyakit THT rinitis kronis dan epistaksis.
2. Menghitung tingkat akurasi / kehandalan jaringan syaraf tiruan yang dibuat

1.5. Manfaat Penelitian

Manfaat yang akan diperoleh dari penelitian ini adalah membantu kerja para ahli dibidangnya terutama *rinitis kronis dan epistaksis*

1.6. Metode Penelitian

Metode yang dilakukan dalam penelitian ini adalah metode pengumpulan data dengan melaksanakan kegiatan sebagai berikut:

1. Wawancara dengan pihak-pihak yang terkait dengan permasalahan yang diteliti, dalam hal ini adalah dari bagian diklit, Unit penyakit THT dirumah sakit Mardi Rahayu Kudus.
2. Observasi atau pengumpulan data pasien yang terjangkit penyakit THT terutama *rinitis kronis dan epistaksis*
3. Studi literatur dengan mencari literatur dan artikel yang menunjang penyusunan skripsi baik itu tentang penyakit THT dan tentang jaringan syaraf tiruan.

1.7. Sistematika Penulisan

Penulisan skripsi ini terbagi menjadi tiga bagian yang mencakup bagian awal, bagian utama yang terdiri atas lima bab, dan bagian akhir.

1.7.1 Bagian awal, berisi : halaman sampul depan, halaman judul, halaman pengesahan, prakata, daftar isi, daftar tabel, daftar gambar, daftar lampiran, arti lambang (jika ada), singkatan, dan instisari.

1.7.2 Bagian utama skripsi :

BAB I : PENDAHULUAN

Meliputi latar belakang permasalahan yaitu latar belakang jaringan saraf tiruan, penyakit THT terutama *rinitis kronis dan epistaksis*. Berdasarkan latar belakang masalah tersebut, penulis menyusun rumusan masalah yang akan diangkat dalam tugas akhir ini. Manfaat penelitian akan dapat dirasakan apabila tujuan penelitian telah tercapai. Metodologi penelitian merupakan tahapan dalam menuntun penulis dalam mencapai tujuan skripsi

BAB II : LANDASAN TEORI

Berisi teori-teori pendekatan yang digunakan untuk menganalisis masalah dan teori yang dipakai dalam mengolah data yang digunakan dalam penelitian. Uraian pada landasan teori ini akan menunjukkan bahwa permasalahan yang diteliti memiliki dasar teori dan dapat dipecahkan melalui penelitian yang akan dilakukan penulis.

BAB III : PERANCANGAN SISTEM

Bab ini merupakan tahap dasar dari pencapaian tujuan yaitu aplikasi jaringan saraf tiruan *backpropagation* untuk memprediksi penyakit THT terutama *rinitis kronis dan epistaksis*. Bab ini berisi penjabaran perancangan system yang terdiri dari perangkat perancangan sistem, identifikasi sistem, data pelatihan, data pengujian, dan perancangan JST *Backpropagation* untuk memprediksi penyakit THT terutama *rinitis kronis dan epistaksis*.

BAB IV : HASIL DAN PEMBAHASAN

Bab Meliputi pembahasan prosedur pembuatan program sebagai wujud dari penggunaan aplikasi jaringan saraf tiruan. Program disusun mulai dari pembentukan neuron input sampai dengan proses prediksi

suatu penyakit yaitu dihasilkannya produk yang sama dengan target.

BAB V : PENUTUP

Bab ini berisi kesimpulan serta saran, kesimpulan merupakan hasil dari analisis data serta perancangan dan implementasi program. Berdasarkan tujuan awal tugas akhir kesimpulan harus dapat mencapai poin tujuan tersebut. Salah satu dari poin tujuan adalah menghasilkan metode untuk prediksi awal suatu penyakit dengan mengaplikasikan jaringan saraf tiruan. Saran dari tugas akhir ini merupakan dasar dari pengembangan penelitian selanjutnya yang belum sampai dibahas dalam tugas akhir ini.

II. LANDASAN TEORI

2.1 THT (Telinga, Hidung, Tenggorok)

Banyak penyakit sistematis yang bermanifestasi di daerah telinga, hidung dan tenggorokan. Demikian juga sebaliknya. Jenis penyakit yang menyerang telinga antara lain : *otitis, gangguan pendengaran, presbikusis, tuli, meniere, dll.* Sedangkan yang menyerang bagian hidung antara lain : *epistaksis, sinusitis, rinitis, tumor, dll.* Pada bagian tenggorokan antara lain: *esofagitis, tumor, laryngitis, dll.*

Hidung bekerja sebagai indera pencium dengan adanya *mukosa olfaktorius* pada atap rongga hidung, konka superior dan sepertiga bagian atas septum. Partikel bau dapat mencapai daerah ini dengan cara difusi dengan palat lendir atau bila menarik nafas dengan kuat.

Epistaksis adalah perdarahan dari hidung yang terjadi akibat lokal ataupun sebab sistemik. Diagnosis awalnya biasanya karena penyakit *kardiovaskular*, kelainan darah, infeksi sistemik, gangguan endokrin, ataupun kelainan *kongenital*.

Sedangkan penyakit **Rinitis Kronis** merupakan suatu penyakit infeksi hidung kronik dengan tanda adanya *atrofi progresif* tulang dan *mukosa konka*. Mukosa hidung menghasilkan secret kental dan cepat mengering, sehingga terbentuk *krusta* berbau busuk. Penyebabnya bisa berupa *spesies*

klebsiella, stafilokok, streptokok, dan pseudomonas aeruginosa. Keluhan subyektif yang sering ditemukan pada pasien biasanya napas bau, pasien menderita anosmia, ingus kental hijau, kusta hijau, gangguan penciuman, sakit kepala, dan hidung tersumbat. Gambar penyakit rinitis kronis ditunjukkan oleh gambar dibawah ini



(“Kapita selekta kedokteran”, 2001)

Gambar 1. penyakit Rinitis Kronis

2.2.1. Teori Jaringan Syaraf Tiruan

a. Pengertian Jaringan Syaraf Tiruan

Menurut Hermawan, Arief . 2006 . Jaringan Syaraf Tiruan, Teori dan Aplikasi. “Jaringan syaraf tiruan didefinisikan sebagai suatu sistem pemrosesan informasi yang mempunyai karakteristik menyerupai jaringan saraf manusia”. Jaringan saraf tiruan tercipta sebagai suatu generalisasi model matematis dari pemahaman manusia (*human cognition*) yang didasarkan atas asumsi sebagai berikut :

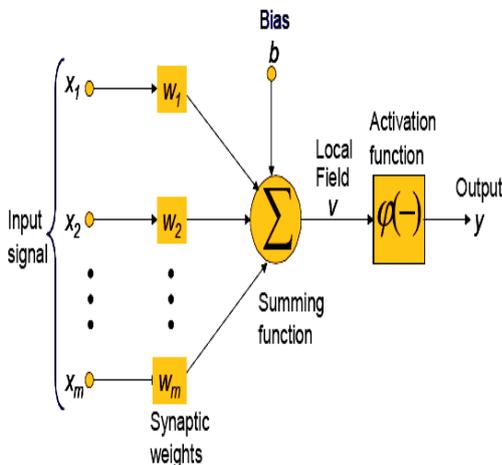
1. Pemrosesan informasi terjadi pada elemen sederhana yang disebut *neuron*.
2. Isyarat mengalir di antara sel saraf / neuron melalui suatu sambungan penghubung.
3. Setiap sambungan penghubung memiliki bobot yang bersesuaian.
4. Setiap sel saraf akan merupakan fungsi aktivasi terhadap isyarat hasil penjumlahan berbobot yang masuk kepadanya untuk menentukan isyarat keluarannya.

b. Pemodelan Jaringan Syaraf Tiruan

Jaringan syaraf terdiri atas beberapa *neuron* dan ada hubungan antara neuron-neuron tersebut. *Neuron* adalah sebuah unit pemroses informasi yang menjadi dasar pengoperasian jaringan syaraf tiruan. Syaraf adalah sebuah unit pemroses informasi dengan tiga elemen dasar yaitu :

1. Satu set link yang terhubung
2. Sebuah penjumlah untuk menghitung besarnya penambahan pada sinyal masukan
3. Sebuah fungsi aktivasi untuk membatasi banyaknya keluaran pada syaraf

Sebagian besar jaringan syaraf melakukan penyesuaian bobot-bobotnya selama menjalani pelatihan. Pelatihan dapat berupa pelatihan terbimbing (*supervised training*) di mana diperlukan pasangan masukan-sasaran untuk tiap pola yang dilatihkan. Jenis kedua adalah pelatihan tak terbimbing (*unsupervised training*). Gambar 2.2 dibawah ini menggambarkan model jaringan syaraf tiruan.



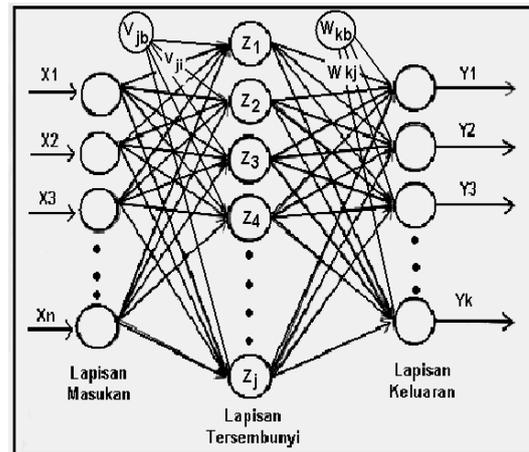
Gambar 2. Model Jaringan Syaraf Tiruan

c. Jaringan Syaraf Tiruan Backpropagation

JST dengan layar tunggal memiliki keterbatasan dalam pengenalan pola. Kelemahan ini bisa ditanggulangi dengan menambahkan satu atau beberapa layar tersembunyi diantara layar masukan dan layar keluaran. Jaringan syaraf tiruan backpropagation (JST-BP) melatih jaringan

mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan untuk memberikan respon yang benar terhadap pola masukan yang serupa dengan pola yang dipakai selama pelatihan.

d. Arsitektur JST Backpropagation



Gambar 3. Model jaringan backpropagation

Backpropagation memiliki beberapa unit yang ada dalam satu atau lebih lapis tersembunyi. Gambar 2.3 diatas adalah arsitektur *backpropagation* dengan n buah masukan ($X_1, X_2, X_3, \dots, X_n$) ditambah sebuah bias, sebuah lapis tersembunyi yang terdiri dari j unit ditambah sebuah bias, serta k buah unit keluaran.

e. Algoritma Backpropagation

Pelatihan backpropagation meliputi tiga fase.

Fase I : *Propagasi Maju*

Selama propagasi maju, sinyal masukan ($= x_i$) dipropagasikan ke lapis tersembunyi menggunakan fungsi aktivasi yang ditentukan. Keluaran dari setiap unit lapis tersembunyi ($= z_j$) tersebut selanjutnya dipropagasikan maju lagi ke lapis tersembunyi di atasnya menggunakan fungsi aktivasi yang ditentukan. Demikian seterusnya hingga menghasilkan keluaran jaringan ($= y_k$).

Berikutnya, keluaran jaringan ($= y_k$) dibandingkan dengan target yang harus dicapai ($= t_k$). Selisih $t_k - y_k$ adalah

kesalahan yang terjadi. Jika kesalahan ini lebih kecil dari batas toleransi yang ditentukan, maka iterasi dihentikan. Akan tetapi apabila kesalahan masih lebih besar dari batas toleransinya, maka bobot setiap garis dalam jaringan akan dimodifikasikan untuk mengurangi kesalahan yang terjadi.

Fase II : *Propagasi Mundur*

Berdasarkan kesalahan $t_k - y_k$, dihitung faktor δ_k ($k=1, 2, \dots, m$) yang dipakai untuk mendistribusikan kesalahan di unit y_k ke semua unit tersembunyi yang terhubung langsung dengan y_k . δ_k juga dipakai untuk mengubah bobot garis yang menghubungkan langsung dengan unit keluaran.

Dengan cara yang sama, dihitung δ_j di setiap unit di lapis tersembunyi sebagai dasar perubahan bobot semua garis yang berasal dari unit tersembunyi di lapis di bawahnya. Demikian seterusnya hingga faktor δ di unit tersembunyi yang berhubungan langsung dengan unit masukan dihitung.

Fase III : *Perubahan Bobot*

Setelah semua faktor δ dihitung, bobot semua garis dimodifikasi bersamaan. Perubahan bobot suatu garis didasarkan atas faktor δ neuron di lapis atasnya. Sebagai contoh, perubahan bobot garis yang menuju ke lapis keluaran didasarkan atas dasar δ_k yang ada di unit keluaran.

Ketiga fase tersebut diulang-ulang terus hingga kondisi penghentian dipenuhi. Umumnya kondisi penghentian yang sering dipakai adalah jumlah iterasi atau kesalahan. Iterasi akan dihentikan jika jumlah iterasi yang dilakukan sudah melebihi jumlah maksimum iterasi yang ditetapkan, atau jika kesalahan yang terjadi sudah lebih kecil dari batas toleransi yang diijinkan.

f. Optimalitas jaringan backpropagation

1) Pemilihan bobot dan bias awal

Bobot awal akan mempengaruhi apakah jaringan mencapai titik minimum lokal atau global, dan seberapa cepat konvergensinya. Bobot yang menghasilkan nilai turunan aktivasi yang kecil sedapat mungkin dihindari karena akan menyebabkan perubahan bobotnya menjadi sangat kecil. Demikian pula nilai bobot awal tidak boleh terlalu besar karena nilai turunan fungsi aktivasinya menjadi sangat kecil juga.

2) Jumlah unit tersembunyi

Hasil teoritis yang didapat menunjukkan bahwa jaringan dengan sebuah lapis tersembunyi sudah cukup bagi *backpropagation* untuk mengenali sembarang perkawanan antara masukan dan target dengan tingkat ketelitian yang ditentukan. Akan tetapi penambahan jumlah lapis tersembunyi kadangkala membuat pelatihan lebih mudah dan bisa jadi pelatihan dapat lebih mencapai target.

3) Jumlah pola pelatihan

Tidak ada kepastian tentang berapa banyak pola yang diperlukan agar jaringan dapat dilatih dengan sempurna. Jumlah pola yang dibutuhkan dipengaruhi oleh banyaknya bobot dalam jaringan serta tingkat akurasi yang diharapkan.

4) Lama iterasi

Tujuan utama penggunaan *backpropagation* adalah mendapatkan keseimbangan antara penganalan pola pelatihan secara benar dan respon yang baik untuk pola lain yang sejenis. Jaringan dapat dilatih terus menerus hingga semua pola pelatihan dikenali dengan benar. Akan tetapi hal itu tidak menjamin jaringan akan mampu mengenali pola pengujian dengan tepat. Jadi tidaklah bermanfaat untuk meneruskan iterasi hingga semua kesalahan pola pelatihan sama dengan 0 (nol).

5) Parameter laju pelatihan

Parameter laju pelatihan (*learning rate*) sangat berpengaruh pada proses pelatihan. Begitu pula terhadap efektivitas dan kecepatan mencapai konvergensi dari pelatihan. Nilai optimum dari *learning rate* tergantung

permasalahan yang diselesaikan, prinsipnya dipilih sedemikian rupa sehingga tercapai konvergensi yang optimal dalam proses pelatihan.

Nilai *learning rate* yang cukup kecil menjamin penurunan *gradient* terlaksana dengan baik, namun ini berakibat bertambahnya jumlah iterasi.

g. Variasi Backpropagation

Variasi ini biasa digunakan untuk keperluan khusus, atau teknik modifikasi bobot untuk mempercepat pelatihan dalam kasus tertentu. Beberapa variasi backpropagation diantaranya adalah :

1) Momentum

Penambahan momentum dimaksudkan untuk menghindari perubahan bobot yang mencolok akibat adanya data yang sangat berbeda dengan yang lain. Modifikasi yang dapat dilakukan adalah melakukan perubahan bobot yang didasarkan atas arah gradien pola terakhir dan pola sebelumnya yang dimasukkan.

Dengan penambahan momentum, bobot baru pada waktu ke- $(t + 1)$ didasarkan atas bobot pada waktu t dan $t-1$. Jika μ adalah konstanta yang menyatakan momentum, maka bobot baru dihitung berdasarkan persamaan (JJ Siang, 2005: 113) :

$$w_{kj}(t+1) = w_{kj}(t) + \alpha \delta_k z_k + \mu (w_{kj}(t) - w_{kj}(t-1))$$

dan

$$v_{ji}(t+1) = v_{ji}(t) + \alpha \delta_j x_i + \mu (v_{ji}(t) - v_{ji}(t-1))$$

2) Delta - Bar - Delta

Laju pemahaman (α) merupakan suatu konstanta yang dipakai dalam seluruh iterasinya. Perubahan dapat dilakukan dengan memberikan laju pemahaman yang berbeda-beda untuk setiap bobotnya, atau tiap iterasinya. Perubahan bobot dalam aturan delta-bar-delta adalah sebagai berikut :

$$w_{kj}(t+1) = w_{kj}(t) + \alpha_{kj}(t+1) \delta_k z_j$$

3) Perubahan Bobot Berkelompok

Variasi lain yang dapat dilakukan pada standar backpropagation adalah merubah bobotnya sekaligus setelah semua pola dimasukkan. Prosedur ini memberikan efek yang lebih halus dalam perubahan bobot. Dalam beberapa kasus, variasi perubahan ini akan meningkatkan kemungkinan konvergensi ke titik minimum lokal.

h. Sum Square Error dan Root Mean Square Error

Kesalahan pada keluaran jaringan merupakan selisih antara keluaran sebenarnya dengan keluaran yang diinginkan. Selisih yang dihasilkan antara keduanya biasanya ditentukan dengan cara dihitung menggunakan suatu persamaan.

Sum Square Error (SSE) dihitung sebagai berikut :

1. Hitung keluaran jaringan syaraf untuk masukan pertama.
2. Hitung selisih antara nilai keluaran jaringan syaraf dan nilai target / yang diinginkan untuk setiap keluaran.
3. Kuadratkan setiap keluaran kemudian hitung seluruhnya.

Adapun rumusnya adalah:

$$SSE = \sum_p \sum_j (T_{jp} - X_{jp})^2$$

Dengan :

T_{jp} = nilai keluaran jaringan syaraf

X_{jp} = nilai target / yang diinginkan untuk setiap keluaran

Root Mean Square Error (RMS Error):

1. Hitung SSE
2. Hasilnya dibagi dengan perkalian antara banyaknya data pada pelatihan dan banyaknya keluaran, kemudian diakarkan.

Rumus :

$$RMS\ Error = \sqrt{\frac{\sum_p \sum_j (T_{jp} - X_{jp})^2}{n_p n_o}}$$

Dengan :

T_{jp} = nilai keluaran jaringan syaraf

X_{jp} = nilai target / yang diinginkan untuk setiap keluaran

n_p = jumlah seluruh pola

n_o = jumlah keluaran

III. PERANCANGAN SISTEM

3.1. Perangkat yang dipakai

Perangkat Keras (Hardware)

PC (Personal Computer) dengan spesifikasi :

CPU : Intel Pentium 4 Celeron 1.7 Ghz

RAM : 256 MB

HRD : 80 GB

VGA : On Board share 128 MB

Monitor : 15" SVGA

Printer : Canon ip1000

Perangkat Lunak (Software)

OS Windows XP Service Pack 1

MATLAB 7.04

3.2. Identifikasi sistem

3.2.1 Data Masukan

Data ini dapat berupa suatu masukan suatu device ataupun data statistik dari suatu percobaan. Nilai data adalah bebas artinya nilai dapat diisi dengan sembarangan bilangan. Untuk perancangan sistem ini menggunakan data masukan berupa numeric dari gejala penyakit THT *rinitis kronis dan epistaksis* itu sendiri meliputi Hidung Tersumbat, pilek, bersin, keluar sekret, hidung gatal, batuk, alergi, nyeri kepala, demam, keluar darah, lemas

- a. Hidung tersumbat (X_1)
 - Diberi nilai 0 jika 'tidak'
 - Diberi nilai 1 jika 'ya'
- b. Pilek (X_2)
 - Diberi nilai 0 jika 'tidak'
 - Diberi nilai 0,25 jika 'ya' tetapi lama pilek kurang dari 3 minggu
 - Diberi nilai 0,50 jika 'ya' tetapi lama pilek antara 3 sampai 8 minggu
 - Diberi nilai 1 jika 'ya' dengan lama pilek lebih dari 8 minggu
- c. Bersin (X_3)
 - Diberi nilai 0 jika 'tidak'
 - Diberi nilai 0,25 jika 'ya' tetapi kadang – kadang dan periode bersinnya tidak tetap.
 - Diberi nilai 0,50 jika 'ya' tetapi dengan serangan kurang dari 5 kali per periode.
 - Diberi nilai 1 jika 'ya' dengan serangan lebih besar dari 5 kali per periode.
- d. Keluar sekret (X_4)

- Diberi nilai 0 jika 'tidak'
 - Diberi nilai 1 jika 'ya'
- e. Hidung gatal (X_5)
 - Diberi nilai 0 jika 'tidak'
 - Diberi nilai 1 jika 'ya'
 - f. Batuk (X_6)
 - Diberi nilai 0 jika 'tidak'
 - Diberi nilai 0,50 jika 'ya' tetapi batuk ringan
 - Diberi nilai 1 jika 'ya' dengan batuk berat atau berdahak
 - g. Alergi (X_7)
 - Diberi nilai 0 jika 'tidak'
 - Diberi nilai 1 jika 'ya'
 - h. Nyeri kepala (X_8)
 - Diberi nilai 0 jika 'tidak'
 - Diberi nilai 1 jika 'ya'
 - i. Demam (X_9)
 - Diberi nilai 0 jika 'tidak'
 - Diberi nilai 1 jika 'ya'
 - j. Keluar darah (X_{10})
 - Diberi nilai 0 jika 'tidak'
 - Diberi nilai 0,50 jika 'ya' tetapi kurang dari 5 ml
 - Diberi nilai 1 jika 'ya' dan melebihi 5 ml
 - k. Lemas (X_{11})
 - Diberi nilai 0 jika 'tidak'
 - Diberi nilai 1 jika 'ya'

3.3 Normalisasi

Sebelum digunakan untuk proses pelatihan, perlu dilakukan penskalaan terhadap nilai-nilai masukan dan target sedemikian hingga data-data masukan dan target tersebut masuk dalam suatu *range* tertentu yang disebut *preprocessing* atau normalisasi data. Tujuan utama normalisasi adalah agar terjadi sinkronisasi data, disamping itu juga untuk memudahkan dalam proses komputasi. Pada MATLAB ada beberapa *tools* untuk *preprocessing* diantaranya *premnmx* dan *prestd*

3.4 Perancangan Sistem

3.4.1 Perancangan Jaringan Syaraf Tiruan

a) Penentuan arsitektur jaringan

Penentuan arsitektur jaringan pada JST tidak memiliki rumusan khusus, untuk itu diperlukan adanya percobaan-percobaan. Percobaan pertama dengan 1 *hidden layer*. Dengan unit hidden layer juga dirandom mulai dari 11 unit sampai 40 unit. Percobaan yang kedua dengan menambahkan 1 hidden layer lagi sehingga terdapat 2 hidden layer. Untuk menentukan unit hidden layer yang kedua juga dilakukan dengan mengurutkan nilai mulai dari 6 unit sampai 25 unit. Untuk lapisan keluaran ada 3 unit yang menjelaskan mengenai kondisi seseorang, yaitu normal, rinitis kronis atau epistaksis. pada Matlab menggunakan instruksi newff.

$$\text{Net} = \text{newff}(\text{PR}, [\text{S1 S2 S3}], \{\text{TF1 TF2 TF3}\}, \text{BTF})$$

b) Penentuan fungsi aktivasi

Ada beberapa model fungsi aktivasi yang disediakan diantaranya yaitu, logsig, tansig, purelin. Pada penelitian ini fungsi aktivasi yang digunakan adalah logsig. Nilai logsig digunakan karena nilai ini mudah untuk komputasi dan mudah diturunkan (*determinan*).

c) Penentuan variasi jaringan

variasi jaringan yang digunakan pada penelitian ini berguna untuk membatasi eksekusi dari program diantaranya adalah menentukan jumlah epoch maksimum, menentukan nilai momentum dan pesat belajar, dan menentukan target yang ingin dicapai.

3.4.2 Perancangan Pelatihan Jaringan

Pelatihan jaringan digunakan untuk melatih set data yang telah dibuat, yaitu data input berupa nilai dari data-data hasil perekaman alat EKG. Sedangkan data targetnya adalah kondisi seseorang tersebut apakah denyut jantungnya normal atau abnormal. Pelatihan dapat dilakukan secara standar dengan ditambahkan berbagai variasi untuk optimasi target.

Setiap variasi parameter diamati dengan menghitung MSE dan lama iterasinya. Setelah nilai yang optimal didapatkan maka nilai tersebut nantinya digunakan untuk melakukan pengujian jaringan.

3.4.3 Perancangan Pengujian Jaringan

Pengujian mutlak dilakukan setelah langkah pelatihan selesai dan didapatkan nilai-nilai bobot dan bias yang optimal. Jaringan yang telah jadi tersebut kemudian diuji dengan dua pengujian yaitu pengujian dengan set-data pelatihan dan pengujian dengan set-data pengujian. Set-data pengujian adalah set-data yang belum pernah dilatihkan sebelumnya (yang tidak digunakan untuk proses pelatihan).

3.4.4 Perancangan Analisis Hasil Pengujian

Kinerja dari suatu jaringan syaraf tiruan setelah dilakukan pelatihan dapat diukur dengan melihat *error* hasil pelatihan, dan pengujian terhadap sekumpulan data masukan baru. Hasil pelatihan dan pengujian dapat dianalisis dengan mengamati ketepatan atau akurasi antara target dengan keluaran jaringan, yang dirumuskan :

$$PK(\%) = \left| \frac{\text{JumlahData Uji} - \text{JumlahKesalahanKeluaranJST}}{\text{JumlahData Uji}} \right| \times 100\%$$

Keluaran jaringan akan dibandingkan dengan hasil diagnosa dokter. Dari hasil perbandingan ini akan diketahui keunggulan ataupun kelemahan jaringan syaraf tiruan dibandingkan dengan diagnosa dokter ataupun ahli tersebut.

IV. HASIL DAN PEMBAHASAN

4.1 Analisis pertama

4.1.1 Analisis Layer tersembunyi

Hasil pelatihan memiliki kesamaan dan perbedaan. Kesamaannya terletak pada performanya. Keduanya menunjukkan bahwa *goal* (kinerja tujuan) yang ingin dicapai hasilnya tidak tercapai semuanya

(pelatihan pertama dan kedua), meskipun epoch yang diinginkan tercapai. Ini menunjukkan bahwa jaringan masih belum mampu mengenali pola masukan yang diberikan dengan benar. Sebenarnya ini bukan suatu masalah karena grafik yang dihasilkan selama pelatihan menunjukkan adanya penurunan MSE.

Ada beberapa masalah yang mengakibatkan goal tidak tercapai pada kedua pelatihan diatas. Pertama epoch yang disediakan 10000 epoch, kalau menginginkan kinerja tujuan tercapai maka perlu ditambah epoch lagi sampai takberhingga hingga kinerja tujuan tercapai. Ini menjadi kendala sebab akan membutuhkan banyak waktu. Padahal yang dibutuhkan waktu yang relatif singkat. Kedua kecilnya laju belajar. Laju belajar yang digunakan pada pelatihan pertama dan kedua adalah 0.1, ini membuat jaringan sulit untuk mengenali pola. Akan lebih baik jika laju belajar ditambahkan, ini akan dibahas pada sub-bab berikutnya. Ketiga momentum yang kecil. Momentum yang kecil akan berakibat pada penurunan gradien. Idealnya momentum yang digunakan berkisar antara 0.4 sampai dengan 0.8.

Perbedaan yang terjadi antara penggunaan satu hidden layer dengan dua hidden layer tersebut ada pada MSE-nya. Memang MSE yang diinginkan tidak tercapai, tetapi dari hasil pelatihan MSE yang dihasilkan relatif kecil. Hasil-hasil pelatihan diatas tetap digunakan dengan melihat MSE yang paling kecil dari tabel diatas. MSE terkecil pada pelatihan pertama adalah **0.00275369**, sedangkan pada pelatihan kedua MSE yang terkecilnya adalah **0.00170939**. Dari perbedaan MSE ini menjelaskan bahwa penggunaan layer tersembunyi dapat meningkatkan performa dari jaringan yang dibuat.

Dari hasil pelatihan pertama dan kedua maka dapat disimpulkan bahwa arsitektur jaringan yang optimal berdasar

pertimbangan diatas adalah jaringan dengan satu hidden layer sudah cukup yaitu 11-29-3, karena dengan 2 hidden layer perbedaan MSE sangat tipis dari satu hidden layer, jadi untuk mempercepat proses iterasi digunakan satu hidden layer

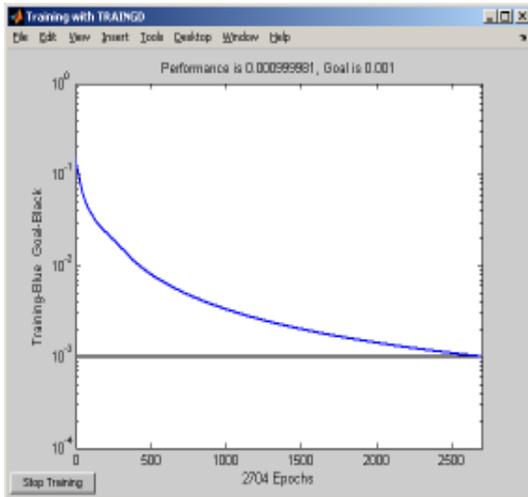
4.1.2 Analisis Laju belajar dan moentum

Analisis kedua bertujuan untuk menganalisis pelatihan ketiga. Dari hasil pengamatan bahwa untuk nilai laju belajar yang cukup kecil menjamin penurunan gradient terlaksana dengan baik, namun ini berakibat bertambahnya jumlah iterasi sehingga untuk mencapai konvergensi berlangsung lebih lama. Jika konstanta laju belajar bertambah besar maka konvergensi akan berlangsung lebih cepat. Kemudian penggunaan konstanta momentum berfungsi untuk mempercepat konvergensi dan juga untuk mencegah terjebaknya pelatihan ke dalam minimum lokal, jika momentum semakin besar maka konvergensi akan cepat tercapai. Akan tetapi penggunaan konstanta momentum jangan terlalu tinggi, sebab kemungkinan gelombang yang dihasilkan akan terjadi noise. Gambar grafik pelatihan ketiga pada sub-bab 3.4 menjelaskan bahwa penggunaan pesat belajar dan momentum yang terus bertambah mengakibatkan jaringan akan mencapai target yang diinginkan

Nilai 0,8 untuk pesat belajar (LR) dan 0,2 untuk momentum dipilih dengan mempertimbangkan beberapa aspek. Pertama aspek prosentase data uji. Ada beberapa pasangan LR dan momentum yang mencapai target yang diinginkan, namun ketika diberi data uji pasangan tersebut belum mampu menghasilkan prosentase yang sempurna (100 %).

Aspek kedua adalah aspek grafik. Grafik yang dihasilkan dari berbagai pasangan berbeda-beda meski pencapaian targetnya sama. Jika fungsi alih yang digunakan adalah logsig artinya logaritmik, maka grafik yang dihasilkan harusnya berbentuk logaritmik pula. Akan lebih baik jika grafik yang dihasilkan tidak terdapat noise

Hasil terbaik terjadi pada laju belajar **0,8** dan konstanta momentum **0,2** yang menghasilkan MSE sebesar **0.000999981**. Hasil terbaik tersebut disajikan pada Gambar 4



Gambar 4. pelatihan terbaik

4.1.3 Analisis variasi jaringan

Analisis ini bertujuan menganalisis pelatihan keempat yakni pelatihan untuk menentukan variasi jaringan. Tujuan dari pelatihan ini adalah mencari variasi jaringan yang mampu menghasilkan kesalahan terkecil dengan waktu komputasi yang cepat. Pelatihan dilakukan dengan hasil pelatihan terbaik sebelumnya yaitu arsitektur jaringan 11-29-3, laju belajar 0,8; momentum 0,2; target error (MSE) 0,001 dan maksimum epochnya 10000 epoch.

Ada beberapa variasi jaringan yang dapat digunakan dalam JST backpropagation. Setiap variasi yang dilatihkan memiliki karakteristik masing-masing. Untuk menentukan variasi yang digunakan perlu melihat MSE, jumlah epoch dan prosentase hasil pengujian. Ada beberapa variasi yang MSE-nya tercapai namun prosentasenya tidak tercapai, ada yang MSE tidak tercapai namun prosentasenya tercapai. Selain itu, perlu juga melihat grafik yang dihasilkan

Dari berbagai pertimbangan diatas maka variasi yang paling baik adalah menggunakan *traingdx*. Selain menghasilkan waktu komputasi yang singkat, grafiknya juga baik.

4.1. Analisis kedua

analisis kedua adalah Proses pengujian, pengujian dilakukan dengan dua data yaitu set-data pelatihan dan set-data pengujian. Hasil yang didapatkan yaitu keduanya masing-masing memiliki kemampuan 100 %, ini artinya jaringan mampu untuk mengenali pola yang diberikan.

V. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan pada bab sebelumnya dapat diambil kesimpulan, sebagai berikut :

1. Berdasar dari hasil MSE, jumlah epoch yang diinginkan dan gambar grafik yang dihasilkan maka dapat dijabarkan bahwa Arsitektur jaringan syaraf tiruan yang dibuat memiliki satu lapisan tersembunyi dengan arsitektur jaringan 11-29-3, dengan pesat belajar 0,8 dan nilai momentum 0,2. Fungsi aktivasi yang digunakan logsig dan algoritma pelatihan yang digunakan adalah traingdx.
2. Jaringan saraf tiruan yang dihasilkan pada penelitian ini memiliki tingkat akurasi 100 %, baik pada tahap pelatihan maupun tahap pengujian. Artinya jaringan syaraf turuan yang dibuat dapat mengenali pola data THT dengan sempurna (benar).

5.2. Saran

1. Sistem jaringan yang telah dibuat menunjukkan jaringan dapat mengenali pola dengan sempurna, ini karena data yang diujikan adalah data yang sempurna (data lengkap). Untuk mengetahui keandalan jaringan dapat diuji dengan data yang tidak sempurna.
2. Jaringan syaraf ada banyak jenisnya, salah satunya backpropagation seperti yang digunakan pada penelitian ini. Untuk

mengetahui kinerja jaringan syaraf tiruan yang lebih optimal, perlu digunakan algoritma pelatihan selain *backpropagation*.

3. Untuk lebih memudahkan pengguna, maka perlu dibuat suatu interface seperti GUI (graphical user interface)

DAFTAR PUSTAKA

- Aston, Richard. 1991. **Principles Of Biomedical Instrumentation And Measurement**. Maxwell Macmillan International Editions
- Hermawan, Arief. 2006. **Jaringan Saraf Tiruan Teori dan Aplikasi**. Yogyakarta: ANDI OFFSET.
- Kusumadewi. Sri 2003. **Artificial Intelligence Teknik dan Aplikasi**. Yogyakarta: Graha Ilmu.
- Puspitaningrum, Diah. 2006. **Pengantar Jaringan Saraf Tiruan**. Yogyakarta: ANDI OFFSET.
- Setiawan, Sandi. 1993. **Artificial Intelligence**. Yogyakarta: ANDI OFFSET.
- Siang, JJ 2005. **Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan Matlab**. Yogyakarta: ANDI.

www.elekroindonesia.com/jst diupload tanggal 28 Agustus 2007 pukul 14.40 WIB

http://telkomnika.elektrouad.net/?Volume_2%2C_No_2:Desain_Dan_Penggunaan_%22E2glite_Expert_System_Shell%22_Untuk_Diagnosis_Penyakit_THT diupload tanggal 28 Agustus 2007 pukul 14.23 WIB

BIOGRAFI

Arif Jumarwanto, Pendidikan terakhir S1 Teknik Elektro Unnes

Rudy Hartanto, dosen Teknik Elektro UGM, menekuni bidang konsentrasi Sistem Informasi dan Teknologi Komputer.

Dhidik Prastiyanto, dosen Teknik Elektro Unnes, menekuni bidang konsentrasi Sistem Informasi dan Teknologi Komputer