

JARINGAN SYARAF TIRUAN MENGUNAKAN ALGORITMA BACKPROPAGATION UNTUK MEMPREDIKSI PRESTASI SISWA

Fatmi Zola¹⁾, Gunadi Widi Nurcahyo²⁾, Julius santony³⁾

¹Teknik Komputer Jaringan, SMK Negeri 2 Pinggir, BTN Puri Perawang Blok C, No. 37 Tualang, fatmizola82@gmail.com

^{2,3}Jurusan Teknik Informatika, Universitas Putra Indonesia YPTK Padang.

ABSTRAK

Sekolah Menengah Kejuruan Negeri 1 Bunga Raya (SMK N 1 Bunga Raya) merupakan sekolah kejuruan dalam bidang Multi Kompetensi. Saat ini terdapat 265 orang siswa dengan 5 jurusan (bidang) diantaranya Teknik Komputer Jaringan (TKJ), Teknik Sepeda Motor (TSM), Akuntansi, Teknik Pengolahan Hasil Pertanian (TPHP), Agribisnis Tanaman Perkebunan (ATP). Untuk meningkatkan prestasi siswa, maka sekolah memberikan beasiswa berupa beasiswa prestasi akademik. Sistem Jaringan Syaraf Tiruan menggunakan metode Backpropagation yang diterapkan dalam penelitian ini untuk prediksi prestasi siswa, dan diharapkan dapat membantu memecahkan masalah pemberian beasiswa terhadap siswa yang berprestasi. Diimplementasikan menggunakan Matlab sebagai software yang mendukung. Dengan nilai ujian Sekolah sebagai data pelatihan dan pengujianya. Hasil dari pengujian dengan pola arsitektur 4-2-1, data menjadi dua bagian yaitu 20 data pelatihan dengan persentase error 95,6 %. Dan 20 data pengujian dengan persentase error 100%. Semakin kecil tingkat ketelitian error yang digunakan maka akan semakin kecil penyimpangan hasil Jaringan Syaraf Tiruan dengan target yang diinginkan.

Kata Kunci: Backpropagation, Prediksi, Jaringan Syaraf Tiruan, Software, Matlab

ABSTRACT

Vocational High School Negeri 1 Bunga Raya (SMK N 1 Bunga Raya) is a vocational school in the field of Multi Competency. Currently there are 265 students with 5 majors (field) including Computer Network Engineering (TKJ), Motorcycle Engineering (TSM), Accounting, Agricultural Processing Technique (TPHP), Plantation Agribusiness (ATP). To improve student achievement, then schools provide scholarships in the form of academic achievement scholarships. Artificial Neural Network System using Backpropagation method applied in this research to predict student achievement, and expected to help solve the problem of giving scholarship to student achievement. Implemented using Matlab as a supportive software. With the value of School examination as training data and testing. The result of testing with architectural pattern 4-2-1, the data into two parts that is 20 training data with percentage error 95,6%. And 20 test data with 100% error percentage. The smaller the level of accuracy error used will be the smaller deviation results Artificial Neural Network with the desired target.

Keywords: Backpropagation, Prediction, Artificial Neural Network, Software, Matlab

1. PENDAHULUAN

Prestasi Akademik merupakan suatu istilah yang digunakan untuk menggambarkan pencapaian tingkat keberhasilan dalam bidang akademik. Prestasi ini didapatkan dari usaha belajar yang telah optimal. Prestasi Akademik dapat juga diartikan sebagai nilai rata-rata dari siswa dalam bidang akademik (Ghaemi, and Yazdanpanah, 2014). Prestasi akademik pada seorang siswa dapat dilihat dari nilai rapornya setiap akhir semester.

Sekolah Menengah Kejuruan Negeri 1 Bunga Raya (SMK N 1 Bunga Raya) merupakan sekolah kejuruan dalam bidang Multi Kompetensi. Saat ini terdapat 265 orang siswa dengan 5 jurusan (bidang) diantaranya Teknik Komputer Jaringan (TKJ), Teknik Sepeda Motor (TSM), Akuntansi, Teknik Pengolahan Hasil Pertanian (TPHP), Agrobisnis Tanaman Perkebunan (ATP). Untuk meningkatkan prestasi siswa, maka sekolah memberikan beasiswa berupa beasiswa prestasi akademik. Faktor yang menentukan siswa untuk mendapatkan beasiswa prestasi akademik, adalah nilai pelajaran yang terdapat dalam ujian nasional di atas kriteria ketuntasan mengajar (KKM), tetapi dalam pemberian beasiswa, pihak sekolah sering menemui permasalahan salah satunya kriteria penilaian yang kurang sesuai, adanya faktor kepentingan internal bagi beberapa pihak yang bersangkutan.

Penelitian ini memusatkan kepada pembuatan Jaringan Syaraf Tiruan (*Artificial Neural Network*) dalam memprediksi prestasi siswa dan menggunakan algoritma *Backpropagation* dalam penyelesaiannya. Jika selama ini penentuan prestasi siswa masih secara manual, maka penerapan ini merupakan konsep awal suatu metode untuk memprediksi siswa menggunakan alat bantu komputer yang didukung oleh Jaringan Syaraf Tiruan. Dengan sistem pemrosesan informasi yang didesain dengan menirukan cara kerja otak manusia dalam menyelesaikan suatu masalah, dengan melakukan proses belajar melalui perubahan bobot dan mampu melakukan pengenalan kegiatan berbasis data masa lalu yang akan dipelajari oleh Jaringan Syaraf Tiruan. Sehingga mempunyai kemampuan untuk memberikan keputusan.

Dengan Jaringan Syaraf Tiruan menggunakan metode *Backpropagation* komputer difungsikan sebagai alat untuk memprediksi prestasi siswa dengan tidak mempertimbangkan faktor-faktor lain penyebab kemerosotan atau peningkatan prestasi siswa, misalnya faktor lingkungan, fasilitas, motivasi belajar dan guru. Berdasarkan hal tersebutlah penulis mengangkat judul penelitian **Jaringan Syaraf Tiruan Menggunakan Algoritma *Backpropagation* Untuk Memprediksi Prestasi Siswa.**

2. TINJAUAN PUSTAKA

2.1 Gambaran Umum Penerapan Jaringan Syaraf Tiruan

Dengan berpedoman pada artikel di atas di sini penulis juga akan melakukan penelitian terhadap prestasi siswa dengan menggunakan Jaringan Syaraf Tiruan. Dan artikel di atas bisa nantinya dijadikan panduan dalam menyusun suatu penerapan Jaringan Syaraf Tiruan.

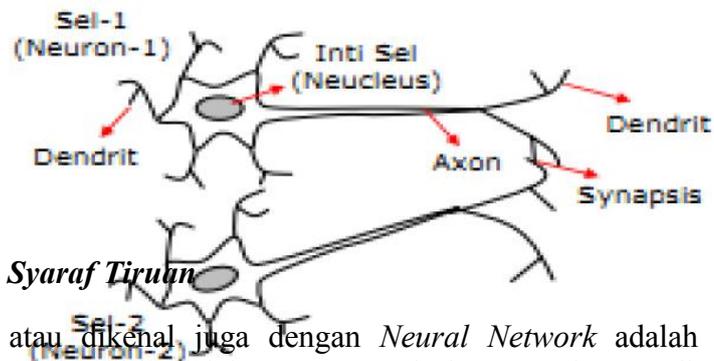
2.2 Jaringan Syaraf Biologis

Otak manusia terdiri atas 10^{11} sel syaraf yang berlaku sebagai prosesor. Sel syaraf ini berhubungan satu dengan yang lain membentuk suatu jaringan yang disebut jaringan syaraf. Proses yang terjadi yang terjadi dalam suatu sel syaraf merupakan proses elektrokimiawi.

2.3 Pengenalan Jaringan Syaraf Biologis

Jaringan Syaraf Biologis adalah kumpulan sel syaraf (*neuron*). *Neuron* mempunyai tugas mengolah informasi. Komponen utama *neuron* dapat dikelompokkan menjadi 3 bagian (Puspitaningrum D., 2006), yaitu :

1. Dendrit bertugas untuk menerima informasi
2. Badan Sel (Soma) berfungsi sebagai tempat pengolahan informasi
3. Akson (Neurit) mengirimkan impuls-impuls ke sel syaraf lainnya



2.4 Pengertian Jaringan Syaraf Tiruan

Jaringan Syaraf Tiruan atau dikenal juga dengan *Neural Network* adalah suatu sistem Jaringan Syaraf Biologis yang ditiru oleh suatu metode komputasi yang digunakan untuk pemrosesan informasi. Ide dasarnya adalah mengadopsi cara kerja otak manusia yang memiliki karakteristik pengolahan paralel, mengolah unsur dalam jumlah besar dan toleransi kesalahan (Setiabudi D., 2015)

Jaringan Syaraf Tiruan menyebutkan 3 hal:

1. Pola hubungan antar *neuron* (disebut arsitektur jaringan).
2. Metode untuk menentukan bobot pengaitan (disebut metode pelatihan/ pembelajaran / algoritma).
3. Fungsi aktivasi (Pratama R.A., and Anifah L., 2016).

1) 2.4.1 Konsep Dasar Jaringan Syaraf Tiruan

Kerangka kerja Jaringan Syaraf Tiruan bisa dilihat dari jumlah lapisan (*layer*) dan jumlah *Neuron*. *Neuron-neuron* tersebut terkumpul di dalam lapisan-lapisan yang disebut *neuron layers* (Madontang Z.A., 2013). Lapisan-lapisan penyusun Jaringan Syaraf Tiruan tersebut dapat dibagi menjadi 3, yaitu :

1. Lapisan *Input*, *node-node* di dalam lapisan *input* disebut unit-unit. Unit-unit input menerima *input* dari dunia luar. *Input* yang dimasukkan merupakan penggambaran dari suatu masalah.
2. Lapisan Tersembunyi, *node-node* dalam lapisan ini disebut unit-unit tersembunyi. Di mana keluarannya tidak dapat secara langsung diamati.

Lapisan *Output*, *node-node* pada lapisan *output* disebut unit-unit *output*. Keluaran atau *output* dari lapisan ini merupakan *output* Jaringan Syaraf Tiruan terhadap suatu permasalahan.

2) 2.4.2 *Arsitektur Jaringan Syaraf Tiruan*

Jaringan Syaraf Tiruan dirancang dengan menggunakan suatu aturan yang bersifat menyeluruh (*general rule*) di mana seluruh model jaringan memiliki konsep dasar yang sama. Arsitektur sebuah jaringan akan menentukan keberhasilan target yang akan dicapai karena tidak semua permasalahan dapat diselesaikan dengan arsitektur yang sama. Arsitektur Jaringan Syaraf Tiruan ada 3 (Wuryandari M.D., and Afrianto I., 2012), yaitu Jaringan lapis tunggal, jaringan dengan banyak lapisan, dan jaringan dengan lapisan kompetitif.

3) 2.4.3 *Mengaktifkan Jaringan Syaraf Tiruan*

Mengaktifkan Jaringan Syaraf Tiruan berarti mengaktifkan setiap *neuron* yang dipakai pada jaringan tersebut. Banyak fungsi yang dapat dipakai sebagai pengaktif, seperti fungsi-fungsi *goniometri* dan hiperboliknya, fungsi unit *step*, *seumlse*, *sigmoid*, dan lain sebagainya. tetapi yang lazim digunakan adalah fungsi *sigmoid*, karena dianggap lebih mendeteksi kinerja sinyal pada otak manusia. Fungsi *sigmoid* memiliki nilai maksimum =1. Maka untuk pola yang targetnya >1, pola masukan dan keluaran harus terlebih dahulu ditransformasikan sehingga semua polanya memiliki *range* yang sama seperti fungsi *sigmoid* yang dipakai.

4) 2.4.4 *Algoritma Umum Jaringan Syaraf Tiruan*

Algoritma pembelajaran/pelatihan Jaringan Syaraf Tiruan (Puspitaningrum D., 2006), dimasukkan contoh pelatihan ke dalam Jaringan Syaraf Tiruan. Lakukan :

1. Inialisasi bobot-bobot jaringan , set $i = 1$.
2. Masukkan contoh ke- i (dari sekumpulan contoh pembelajaran yang terdapat dalam set pelatihan) ke dalam jaringan pada lapisan *input*.
3. Cari tingkat aktivasi unit-unit *output* menggunakan algoritma aplikasi
If kinerja jaringan memenuhi standar yang ditentukan sebelumnya (memenuhi syarat bersih).
Then exit.
4. Update bobot-bobot dengan menggunakan aturan pembelajaran jaringan
5. If $i = n$, then reset $i = 1$,
Else $i = i - 1$,
Ke langkah 2.

Algoritma aplikasi inferensi Jaringan Syaraf Tiruan, dimasukkan sebuah contoh pelatihan ke dalam Jaringan Syaraf Tiruan. Lakukan :

1. Masukkan kasus ke dalam jaringan pada lapisan *input*.
2. Hitung tingkat aktifasi *node-node* jaringan.
3. Untuk jaringan koneksi umpan maju, jika tingkat aktivasi dari semua unit *outputnya* telah dikalkulasi, maka exit j jika tidak, kembali ke langkah 2. Jika jaringannya tidak stabil, maka exit dan fail.

Jadi dengan tingkat kemampuan yang sangat baik, beberapa aplikasi Jaringan Syaraf Tiruan sangaat cocok untuk diterapkan pada :

1. Klasifikasi, memilih suatu *input* data ke dalam satu kategori tertentu yang diterapkan.
2. Asosiasi, menggambarkan suatu objek secara keseluruhan hanya dengan sebuah bagian dari objek lain.

Self organizing, kemampuan untuk mengolah data-data *input* tanpa harus memiliki data sebagai target.

Optimasi, menemukan suatu jawaban atau solusi yang paling baik sehingga seringkali dengan meminimalisasikan suatu fungsi biaya (*optimizer*).

3. METODE PENELITIAN

Pada penelitian ini diuraikan tentang metodologi dipakai dalam penelitian dan kerangka kerja penelitian yang digunakan dalam menyelesaikan penulisan mulai dari mendeskripsikan masalah, menganalisa permasalahan, menentukan tujuan, mempelajari literatur, pengumpulan data, analisis teknik Jaringan Syaraf Tiruan, pengujian algoritma *Backpropagation*, pengolahan data dengan *Matlab R2011b*, dan melakukan pengujian hasil. Kerangka kerja dibuat agar penelitian ini fokus dan terarah dalam mencapai tujuan penelitian. Metode penelitian menjelaskan pendekatan, rancangan kegiatan, ruang lingkup atau objek, bahan dan alat utama, tempat, teknik pengumpulan data, definisi operasional variabel penelitian, dan teknik analisis.

4. HASIL DAN PEMBAHASAN

Adapun proses algoritma *Bacpropagation* yang akan dilakukan dalam proses pelatihan algoritma *Bacpropagation* adalah menentukan parameter-parameter jaringan *Bacpropagation* yaitu menentukan *input*, menentukan *ouput*, menentukan bobot, menentukan bias dan menentukan *threshold* atau nilai ambang (θ). Hal ini dapat di gambarkan melalui algoritma *Bacpropagation* berikut ini:

Algoritma proses *Bacpropagation* :

1. Ambil data
2. Iterasi Pertama Hingga Akhir Dari Data Pelatihan
3. Pelatihan dan Menentukan Parameter-parameter Jaringan
4. Kalkulasi *Error Output*
5. Proses Pengujian Algoritma *Bacpropagation*
6. Hasil.

Adapun data yang dijadikan objek penelitian ini adalah data Nilai Ujian Sekolah Siswa di SMK Negeri 1 Bunga Raya.

Tabel. 1 Data Input Pelatihan Prediksi Prestasi Siswa

No Res	Nilai Ujian				
	B. Ind	B. Ing	MTK	IPA	Rata-rata
1	8,60	8,60	8,50	8,50	8,55
2	9,20	9,80	9,50	9,25	9,44
3	8,40	8,60	8,50	8,40	8,48
4	9,50	8,70	8,60	8,30	8,78
5	8,50	8,60	8,80	8,60	8,63
6	8,40	8,40	7,50	8,25	8,14

7	8,60	8,20	8,25	8,25	8,33
8	9,00	8,60	7,50	8,75	8,46
9	8,50	7,80	8,00	7,80	8,03
10	8,80	9,00	7,50	8,50	8,45
11	8,60	8,60	7,75	8,00	8,24
12	8,80	8,80	7,50	8,75	8,46
13	8,80	8,40	7,75	8,50	8,36
14	7,95	7,90	7,50	7,68	7,76
15	8,39	8,52	8,23	8,29	8,36
16	8,80	8,20	7,50	8,00	8,13
17	8,60	8,25	7,90	8,05	8,20
18	8,80	8,40	7,50	8,50	8,30
19	6,55	6,70	7,50	7,80	7,14
20	8,00	8,00	8,00	7,50	7,88

Dari data sumber diatas dalam Jaringan Syaraf Tiruan dengan *Bacpropagation* ini akan dipakai fungsi aktivasi sigmoid karena pada fungsi aktivasi sigmoid nilai fungsinya terletak antara 0 dan 1 maka data tersebut ditranformasikan dulu. Data ditransformasikan ke interval yang lebih kecil. Proses normalisasi dapat ditentukan dengan menggunakan data sampel pada tabel 4.1 dapat dinormalisasikan dengan rumus *transpormasi* berikut:

$$X' = \frac{0,8(X-b)}{(a+b)} + 0,1 \dots\dots\dots (1)$$

Algoritma Propagasi balik adalah sebagai berikut:

1. Inialisasi bobot
 2. Tetapkan *epoch* maksimum, *target*, *error*, dan *leraning rate*
- Langkah 0: Inialisasi bobot dengan nilai random atau acak yang cukup kecil. *Set learning rate* α ($0 < \alpha \leq 1$)
- Langkah 1: Selama kondisi berhenti masih belum terpenuhi, laksanakan langkah sampai 9.
- Langkah 2: Untuk tiap pasangan pelatihan,kerjakan langkah 3 sampai 8.
- Feedforward*:
- Langkah 3: Setiap unit *input* X_j (dari unit ke-1 sampai unit ke- n pada lapisan *input*) pengiriman sinyal *input* kesemua unit yang ada pada lapisan atasnya kelapisan atasnya (lapisan tersembunyi) $:x_i$.
- Langkah 4 : Pada setiap unit di lapisan tersembunyi Z_j (dari unit ke-1 sampai unit ke- p ; $1=1, \dots, n$; $j=1, \dots, p$) sinyal *output* lapisan tersembunyi dihitung dengan menerapkan fungsi aktivasi terhadap penjumlahan sinyal-sinyal *input* berbobot x_i :

$$Z_j = f(V_{0j} + \sum_{i=1}^n x_i v_{ij}) \dots\dots\dots (2)$$

Di mana:

- Z_j = unit tersembunyi ke-j
- F = fungsi *nonlinear*
- V_{0j} = bias pada unit tersembunyi -j
- $\sum x_i$ = jumlah *input* ke-i
- V_{ij} = bobot ke-j

Kemudian dihitung nilai *output* dengan menggunakan fungsi aktivasi yang digunakan:

$$z_j = f(z_{in j}) \dots\dots\dots(3)$$

Di mana fungsi aktivasi yang digunakan adalah fungsi *sigmoid* biner yang mempunyai persamaan:

$$f(x) = \frac{1}{1+\delta^x} \dots\dots\dots(4)$$

Hasil fungsi tersebut dikirim ke semua *neuron* pada lapisan di atasnya.

Langkah 5: Setiap unit dilapisan *output* y_k (dari unit ke-1 sampai unit ke- m , $1, \dots, n, k=1, \dots, m$) dihitung sinyal *ouputnya* dengan menerapkan fungsi aktivasi terhadap penjumlahan sinyal-sinyal *input* berbobot z_j bagi lapisan ini :

$$Y_k = f(W_{ok} + \sum_{j=1}^p z_j w_{jk}) \dots\dots\dots (5)$$

Di mana :

- Y_k = Unit *output* ke- k
- f = Fungsi *non linier*
- W_{ok} = Unit ke-0 dan k
- $\sum z_j$ = Jumlah unit tersembunyi ke-j
- W_{jk} = Bobot awal

Kemudian dihitung koreksi nilai bobot yang kemudian akan digunakan untuk memperbaharui nilai $\Delta w_{jk} : w_{jk} = \alpha \delta_k z_j \dots\dots\dots(6)$

Hitung koreksi nilai bias yang kemudian akan adigunakan untuk memperbaharui nilai $w_{0k} : \Delta w_{0k} = \alpha \delta_k \dots\dots\dots(7)$

Dan kemudian nilai δ_k dikirim ke *neuron* pada lapisan sebelumnya

Langkah 7: Setiap unit *output* y_k (dari unit ke-1 sampai unit ke= m , $j=1, \dots, p, k=1, \dots, m$) menerima pola target t_k lalu informasi kesalahan lapisan *output* (δ_k) dihitung. δ_k dikirim ke lapisan di bawahnya dan digunakan untuk menghitung besar koreksi bobot dan bias (Δw_{ok}) antara lapisan tersembunyi dengan lapisan *outputnya* :

$$\delta_k = (t_k - y_k) (W_{ok} + \sum_{j=1}^p z_j w_{jk}) \dots\dots\dots(8)$$

- $\Delta w_{jk} = \alpha \delta_k z_j$
- $\Delta w_{ok} = \alpha \delta_k$

Di mana :

- δ_k = Informasi tentang kesalahan pada unit y_k yang disebarkan kembali ke unit tersembunyi.
- t_k = *Output* vektor target ke- k
- Y_k = Unit *output* ke- k
- f = Fungsi *non linier*
- W_{ok} = Unit *input* ke-0 dan k
- $\sum z_j$ = Jumlah unit tersembunyi ke-j
- W_{jk} = Bobot awal

ΔW_{jk} = Perubahan bobot

α = Kecepatan pembelajaran (*learning rate*)

ΔW_{ok} = Perubahan bobot awal

Untuk tiap *hidden neuron* ($Z_j, j=1,2,3,\dots,p$) :

$$\delta_{in_j} = \sum_{k=1}^m \delta_k W_{jk} \dots\dots\dots(9)$$

Kemudian nilai tersebut dikalikan dengan nilai turunan dari fungsi aktivasi untuk menghitung informasi kesalahan: $\delta_j = \delta_{in_j}$

$f'(z_{in_j})$

Hitung koreksi nilai bobot yang kemudian digunakan untuk memperbaharui v_{ij} :

$$\Delta v_{ij} = \alpha \delta_j x_i \dots\dots\dots(10)$$

Dan hitung nilai koreksi bias yang kemudian digunakan untuk memperbaharui nilai

$$v_{0j} : \Delta v_{0j} = \alpha \delta_j$$

Memperbaharui nilai bobot dan nilai bias:

Langkah 8: Tiap nilai bias dan bobot ($j=0,\dots,p$) pada *output neuron* ($Y_k, k=1,2,3,\dots,m$) diperbaharui:

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta v_{ij} \dots\dots\dots(11)$$

$$V_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \dots\dots\dots(12)$$

Langkah 9: Menguji apakah kondisi berhenti sudah terpenuhi. Kondisi berhenti ini terpenuhi jika nilai kesalahan yang dihasilkan lebih kecil dari nilai kesalahan referensi (Giawa D.I., 2017)

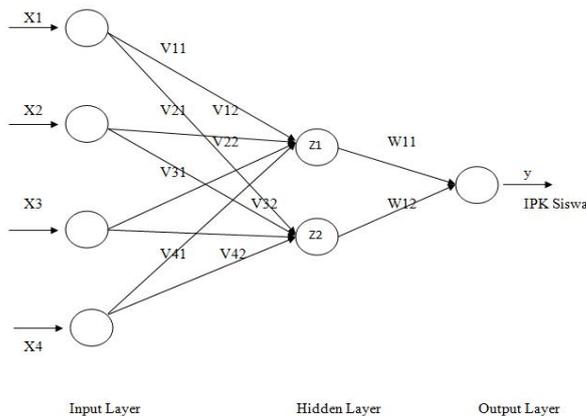
Dengan menstransformasi ini maka data terkecil akan menjadi 0.1 dan data terbesar akan menjadi 0.9. Maka data pada tabel 4.1 diatas setelah ditransformasikan hasilnya dapat ditunjukkan pada tabel 4.2 yang nantinya dipakai sebagai data pelatihan *Bacpropagation*..

Tabel. 2. Hasil Normalisasi Sampel Data *Input* dan *Output* Prediksi Prestasi Siswa

No Res	Nilai Ujian				
	B.Ind (x1)	B.Ing (x2)	MTK (x3)	IPA (x4)	Rata-rata (t1)
1	0,6559	0,5903	0,5000	0,6217	0,5904
2	0,8186	0,9000	0,9000	0,8826	0,8991
3	0,6017	0,5903	0,5000	0,5870	0,5643
4	0,9000	0,6161	0,5400	0,5522	0,6687
5	0,6288	0,5903	0,6200	0,6565	0,6165
6	0,6017	0,5387	0,1000	0,5348	0,4470
7	0,6559	0,4871	0,4000	0,5348	0,5122
8	0,7644	0,5903	0,1000	0,7087	0,5600
9	0,6288	0,3839	0,3000	0,3783	0,4078
10	0,7102	0,6935	0,1000	0,6217	0,5557

11	0,6559	0,5903	0,2000	0,4478	0,4817
12	0,7102	0,6419	0,1000	0,7087	0,5600
13	0,7102	0,5387	0,2000	0,6217	0,5252
14	0,4797	0,4097	0,1000	0,3365	0,3148
15	0,5990	0,5697	0,3920	0,5487	0,5235
16	0,7102	0,4871	0,1000	0,4478	0,4426
17	0,6559	0,5000	0,2600	0,4652	0,4687
18	0,7102	0,5387	0,1000	0,6217	0,5035
19	0,1000	0,1000	0,1000	0,3783	0,0991
20	0,4932	0,4355	0,3000	0,2739	0,3557

Arsitektur jaringan yang akan dibentuk adalah 4-2-1, di mana jumlah unit pada lapisan *input* adalah 4. Jumlah unit pada lapisan tersembunyi (*hidden layer*) adalah 2. Jumlah unit pada lapisan *output* adalah 1. Arsitektur jaringannya seperti dibawah ini :



Gambar Arsitektur JST untuk Prediksi Prestasi Siswa

Dari tabel 2 didapatkan bahwa:

Data *input* Nomor Responden 1: $X_1 = 0,6559$, $X_2 = 0,5903$, $X_3 = 0,5000$ dan $X_4 = 0,6217$ dan $T = [0,5904]$

Penyelesaian :

Langkah 0

Inisialisasi semua bobot dengan bilangan acak kecil.

1. Inisialisasi bobot pada *hidden layer*

Untuk membentuk Jaringan Syaraf Tiruan, terlebih dahulu dilakukan *inisialisasi* bobot awal. Bobot awal yang menghubungkan simpul-simpul pada lapisan *input* dan lapisan tersembunyi untuk arsitektur diatas V_{ij} dipilih secara acak. Demikian pula bobot awal yang menghubungkan simpul-simpul pada lapisan tersembunyi dan lapisan *output*

juga dipilih secara acak dari bilangan random. Berikut adalah data bobot dari *hidden layer* :

Tabel 3 Bobot Yang Diberikan dari *Input Layer* ke *Hidden Layer* dan Bias

<i>In</i>	Z_1	Z_2
X_1	4.7457	4.7786
X_2	-5.6246	4.8679
X_3	1.9958	-4.7393
X_4	-4.3897	0.7256
B1	-1.3499	0.4567

Bobot dari *hidden layer* ke *output layer* penulis ambil dari matlab. Adapun nilai bobot dapat dilihat pada tabel 4.3

Langkah 1

Jika kondisi penghentian belum terpenuhi, lakukan langkah 2 - 9

1. Bobot pada satu *output layer*

Bobot ini adalah bobot yang terdapat pada *hidden layer* menuju *output layer* (W_{jk}). Nilai diambil dari bilangan acak dan akan diperbaiki saat proses perambatan balik, bobot pada *output layer*.

Tabel 4 Bobot Yang Diberikan dari *Hidden Layer* ke *Output Layer*

	Y
Z_1	0.9150
Z_2	0.9298
B2	-0.6848

Diketahui :

$$\begin{aligned}
 X_1 &= 0.6559 & X_2 &= 0.5903 & X_3 &= 0.5000 & X_4 &= 0.6217 \\
 V_{11} &= 4.7457 & V_{12} &= 4.7786 & V_{21} &= -5.6246 & V_{22} &= 4.8679 & V_{31} &= 1.9958 \\
 V_{32} &= -4.7393 & V_{41} &= -4.3897 & V_{42} &= 0.7256 & W_{11} &= 0.9150 & W_{12} &= 0.9298 \\
 \theta_3 &= -1.3499 & \theta_4 &= 0.4567 & \theta_5 &= -0.6848 & \alpha &= 0.2 & Y_d &= 0
 \end{aligned}$$

Langkah 2

Untuk setiap pasang data pelatihan, lakukan langkah 3 - 8

Fase I: Propagasi Maju

Langkah 3

Tiap unit masukkan menerima sinyal dan meneruskan ke unit tersembunyi

Langkah 4

Hitung semua keluaran di unit tersembunyi (Z_j):

Pada tahapan perambatan maju ini dilakukan perhitungan bobot-pada masing-masing layer menggunakan fungsi aktivasi *sigmoid*, langkah-langkahnya sebagai berikut:

- a. Setiap unit *input* ($X_i, i = 1, 2, 3, \dots, p$) menerima sinyal X_i dan meneruskan sinyal tersebut ke semua unit pada *hidden layer*.
- b. Setiap unit tersembunyi ($X_i, i = 1, 2, 3, \dots, p$) menjumlahkan bobot sinyal dengan *input* dengan persamaan rumus (2) berikut.

$$Z_{net\ 1} = V_{j11} + (X_1 * Z_{11}) + (X_2 * Z_{12}) + (X_3 * Z_{13}) + (X_4 * Z_{14})$$

$$= -1,3499 + (0,6559 * 4,7457) + (0,5903 * -5,6246) + (0,5000 * 1,9958) + (0,6217 * -4,3897)$$

$$= -3,2887$$

$$Z_{net\ 2} = V_{j12} + (X_1 * Z_{21}) + (X_2 * Z_{22}) + (X_3 * Z_{23}) + (X_4 * Z_{24})$$

$$= 0,4567 + (0,6559 * 4,7789) + (0,5903 * 4,8679) + (0,5000 * -4,7393) + (0,6217 * 0,7256)$$

$$= 4,5463$$

$$Z_j = f'(z_{netj}) = \frac{1}{1 + e^{-z_{netj}}}$$

$$Z_1 = \frac{1}{1 + e^{-(-3,2887)}} = 0,0360$$

$$Z_2 = \frac{1}{1 + e^{-(4,5463)}} = 0,9895$$

Langkah 5

Hitung keluaran unit *output* (y_k) dengan rumus (5):

Jaringan memiliki 1 unit *output* y , maka:

$$y_{net\ 1} = \sum_{j=1}^1 z_j w_{kj}$$

$$= 0,6848 + (0,0360 * 0,9150) + (0,9895 * 0,9298)$$

$$= (0,2681)$$

$$y = f(y_{net}) = \frac{1}{1 + e^{-y_{net}}}$$

$$y_1 = f(y_{net}) = \frac{1}{1 + e^{-(0,2681)}} = 0,5666$$

Tahap II : Umpan Mundur (*Backward Propagation*)

Langkah 6

Hitung faktor δ di unit *output* (y_k) dengan rumus (8):

$$\delta_k = (t_k - y_k) f'(y_{net\ k}) = (t_k - y_k) y_k (1 - y_k)$$

Karena jaringan hanya memiliki 1 unit *output*, maka:

$$\delta_1 = (t - y) y (1 - y) = (0,5904 - 0,5666) (0,5666) (1 - 0,5666) = 0,0058$$

Kesimpulan : Pada langkah ke 6 sudah didapatkan *output* aktual yang diharapkan, jika dilihat dengan fungsi *threshold* yakni $f(net) = \begin{cases} 1 & \text{jika } net > \theta \\ 0 & \text{jika } net \leq \theta \end{cases}$, maka data Nilai responden 1 sudah dapat dikenali. Karena nilai δ_1 menghasilkan nilai yaitu [0,0058].

Selanjutnya kita lakukan perubahan bobot $w_{jk} = w_{jk}$ dengan ($\alpha = 0.5$) maka dapat dihitung dengan persamaan sebagai berikut:

$$\Delta W_{11} = 0,5(0,0058) (0,0360) = 0,0001$$

$$\Delta W_{12} = 0,5(0,0058) (0,9895) = 0,00029$$

Langkah 7

Hitung penjumlahan kesalahan dari unit tersembunyi (δ) dengan rumus (8):

Jaringan memiliki 1 unit *output*, maka:

$$\delta_{net_1} = (0,0058) + (0,9150) = 0,0053$$

$$\delta_{net_2} = (0,0058) + (0,9298) = 0,0054$$

Untuk menghitung informasi *error*, kalikan nilai ini dengan turunan dari fungsi aktivasinyal dengan rumus :

$$\delta_j = \delta_{net_j} f'(z_{net,j}) = \delta_{net,j} z_j (1 - z_j)$$

$$\delta_1 = 0,0053 (0,0360)(1 - 0,0360) = 0,0002$$

$$\delta_2 = 0,0054 (0,9895)(1 - 0,9895) = 0,0001$$

Kemudian hitung koreksi bobot dengan rumus (10)

	$\Delta V_{ji} = \alpha \delta_j x_i$	
$\Delta V_{11} = \alpha \delta_j x_i$	$= 0,5 * 0,0002 * 0,6559$	$= 0,0001$
$\Delta V_{12} = \alpha \delta_j x_i$	$= 0,5 * 0,0001 * 0,6559$	$= 0,0000$
$\Delta V_{21} = \alpha \delta_j x_i$	$= 0,5 * 0,0002 * 0,5903$	$= 0,0001$
$\Delta V_{22} = \alpha \delta_j x_i$	$= 0,5 * 0,0001 * 0,5903$	$= 0,0000$
$\Delta V_{31} = \alpha \delta_j x_i$	$= 0,5 * 0,0002 * 0,5000$	$= 0,0001$
$\Delta V_{32} = \alpha \delta_j x_i$	$= 0,5 * 0,0001 * 0,5000$	$= 0,0000$
$\Delta V_{41} = \alpha \delta_j x_i$	$= 0,5 * 0,0002 * 0,6217$	$= 0,0001$
$\Delta V_{42} = \alpha \delta_j x_i$	$= 0,5 * 0,0001 * 0,6217$	$= 0,0000$

Tabel 5 Perubahan Ke Unit Tersembunyi

Var	Z1	Z2
X1	0,0001	0,0000
X2	0,0001	0,0000
X3	0,0001	0,0000
X4	0,0001	0,0000

Tahap III : Peng-*update*-an bobot

Langkah 8

Hitung semua perubahan bobot

Untuk perubahan bobot unit *output* dilakukan dengan rumus (11):

$$w_{jk} = w_{kj} w_{jk} (baru) = w_{jk} (lama) + \Delta w_{jk}$$

$$w_{l1} = w_{kj} w_{jk} (baru) = w_{jk} (lama) + \Delta w_{jk}$$

$$\begin{aligned}
 &= 0,9150 + 0,0002 \\
 &= 0,9152 \\
 w_2 &= w_{kj} w_{jk} \text{ (baru)} &= w_{jk} \text{ (lama)} + \Delta w_{jk} \\
 &= 0,9298 + 0,0001 &= 0,9299
 \end{aligned}$$

Tabel 6 Tabel Bobot Baru

W1₁	0.9152
W1₂	0.9299

Untuk menghitung perubahan bobot baru pada *hidden layer* maka digunakan formula rumus (12) sebagai berikut:

$$vij \text{ (baru)} = vij \text{ (lama)} + \Delta vij$$

$$\begin{aligned}
 V11 \text{ (baru)} &= vij \text{ (lama)} + \Delta vij \\
 &= 4,7457 + 0,0001 = 4,7458 \\
 V12 \text{ (baru)} &= vij \text{ (lama)} + \Delta vij \\
 &= 4,7786 + 0,0000 = 4,7786 \\
 V21 \text{ (baru)} &= vij \text{ (lama)} + \Delta vij \\
 &= -5,6246 + 0,0001 = -5,6245 \\
 V22 \text{ (baru)} &= vij \text{ (lama)} + \Delta vij = 4,8679 + 0,0000 = 4,8679 \\
 V31 \text{ (baru)} &= vij \text{ (lama)} + \Delta vij = 1,9958 + 0,0001 = 1,9959 \\
 V32 \text{ (baru)} &= vij \text{ (lama)} + \Delta vij = -4,7393 + 0,0000 = -4,7393 \\
 V41 \text{ (baru)} &= vij \text{ (lama)} + \Delta vij = -4,3897 + 0,0001 = -4,3896 \\
 V42 \text{ (baru)} &= vij \text{ (lama)} + \Delta vij = 0,7254 + 0,0001 = 0,7255
 \end{aligned}$$

Tabel 7 Tabel Bobot Baru Pada Hidden Layer

	Z1	Z2
X1	4,7458	4,7786
X2	-5,6245	4,8679
X3	1,9959	-4,7393
X4	-4,3896	0,7255

Tahapan ini dikatakan *epoch* I pada data Responden 1, di mana proses algoritma *Bacpropagation* ini belum mencapai *error* yang diinginkan, oleh sebab itu pelatihan jaringan harus dilakukan perulangan dimulai dari langkah 2 sampai *error* yang dihasilkan sesuai dengan yang diharapkan. Hasil sementara perubahan dari perubahan bobot yang terjadi pada data Responden 1 dapat dilihat pada tabel 4.11.

Tabel 8 Tabel Persentase Target Dengan Output Aktual

No Res	X1	X2	X3	X4	T	Y	%
1	0,655	0,590	0,500	0,621	0,590	0,006	86,8

Karena hasil sementara ini masih sangat jauh dari target yang diharapkan, maka, untuk pelatihan jaringan *Bacpropagation* dilakukan perulangan sampai *output* yang dihasilkan sesuai dengan Target yang diharapkan.

Dari hasil pengolahan dan perancangan menggunakan Jaringan Syaraf Tiruan dengan metode *Bacpropagation*, maka dapat dilihat hubungan bahwa Jaringan Syaraf Tiruan bisa digunakan untuk melakukan peramalan prestasi siswa data *input* pola 40 data transformasi dibagi yang dibagi menjadi data pelatihan sebanyak 20 data dan data pengujian sebanyak 20 data. Hasil dari proses pengolahan data di atas didapatkan nilai update bobot antara input ke *hidden layer* yang digunakan untuk iterasi data selanjutnya sehingga diperoleh *output actual* yang diharapkan.

Hasil.

Berdasarkan tabel 8 diperoleh hasil dari pengolahan data responden 1 adalah Bahasa Indonesia (x1) sebesar 0,6559, Bahasa Inggris (x2) 0,5903, Matematika (x3) 0,5000, IPA (x4) 0,6217, Target (t) 0,5904 dan *outputnya* 0,0058. Maka persentase yang diperoleh oleh responden 1 adalah sebesar 86,82%. Dengan persentase yang diperoleh tersebut maka responden 1 dinyatakan berprestasi. Bagian ini menyajikan hasil penelitian.

5. SIMPULAN

Setelah melakukan pelatihan dan pengujian serta implementasi dengan menggunakan *software Matlab R2011b* terhadap data nilai siswa SMKN 1 Bunga Raya, maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Dengan menggunakan Jaringan Syaraf Tiruan dengan algoritma *Backpropagation* proses prediksi lebih cepat, akurat, meminimalisir kesalahan dan bisa menggunakan teknologi komputer. Serta mudah dalam pengembangannya.
2. Dapat menganalisis faktor menyebabkan kemerosotan prestasi siswa
3. Aplikasi Jaringan Syaraf Tiruan dapat memprediksi prestasi siswa pada mata pelajaran, Bahasa Indonesia, Bahasa Inggris, Matematika dan IPA dengan algoritma *Backpropagation* dan penggunaan *tools Matlab R2011b*, dengan tingkat akurasi mendekati 100%.

Model Jaringan Syaraf Tiruan dapat memprediksi prestasi siswa pada Mata Pelajaran Ujian Sekolah.

UCAPAN TERIMAKASIH

1. Bapak **H. Herman Nawas** selaku Ketua Yayasan Perguruan Tinggi Komputer (YPTK) Padang yang telah memberikan dukungan baik secara moril dan spritual dalam menyelesaikan studi Program Magister Ilmu Komputer pada Universitas Putra Indonesia "YPTK" Padang.
2. Bapak **Prof. Dr. Sarjon Defit, S.Kom, M.Sc.** selaku Rektor Universitas Putra Indonesia "YPTK" Padang yang telah memberikan dukungan baik secara moril dan spritual dalam menyelesaikan studi Program Magister Ilmu Komputer pada Universitas Putra Indonesia "YPTK" Padang.
3. Bapak **Dr. Julius Santony, S.Kom, M.Kom.** selaku Dekan Fakultas Ilmu Komputer Universitas Putra Indonesia "YPTK" Sekaligus pembimbing II yang telah banyak membantu dan memberi arahan dalam menyelesaikan Jurnal ini.
4. Bapak **Dr. Ir. Gunadi Widi Nurcahyo, M.Sc.** selaku Ketua Program Studi Magister Ilmu Komputer Universitas Putra Indonesia "YPTK" sekaligus pembimbing I yang telah banyak membantu dan memberi arahan dalam menyelesaikan Tesis ini.

5. Seluruh Dosen Pascasarjana Magister Ilmu Komputer yang telah berbagi ilmu selama pendidikan di Universitas Putra Indonesia “YPTK” Padang.
6. Keluarga Besar SMK Negeri 1 Bunga Raya telah memberikan dukungan berupa moril dalam menyelesaikan studi Program Magister Ilmu Komputer pada Universitas Putra Indonesia “YPTK” Padang.
7. Keluarga Besar SMK Negeri 2 Pinggir yang telah memberi dukungan dan motivasi dalam menyelesaikan studi Program Magister Ilmu Komputer pada Universitas Putra Indonesia “YPTK” Padang.
8. Orang Tua Tercinta Ayah (Zalmi Fakri), Ibu (Asnadiar) yang senantiasa mendoa’kan kepada ananda, Suami (Irrizon, S.Hi) dan anak- anak tercinta (Indana Halwah Raqiqah, Rameyzha Elya Raqiqah dan Fayrel Atharizzon Calief)
9. Teman – teman terutama Gang Mayat yang senantiasa memberikan semangat dan berbagi ilmu serta pengalamannya.
10. Rekan-rekan M.Kom angkatan XXVIII-B yang selalu memberikan semangat dan motivasi dalam menyelesaikan tesis ini.

DAFTAR PUSTAKA

- Anike M., Suyoto., and Ernawati . (2012).** “Pengembangan Sistem Jaringan Syaraf Tiruan dalam Memprediksi Jumlah Dokter Keluarga menggunakan *Backpropagation*”.
- Anwar B. (2011)** “Penerapan Algoritma Jaringan Syaraf Tiruan *Backpropagation* dalam Memprediksi Tingkat Suku Bunga Bank”.
- Berutu R. (2017).** “Implementasi Jaringan Syaraf Tiruan untuk Memprediksi Penjualan Kosmetik *Decorative* dengan Metode *Backpropagation*”.
- Gupta A., and Shreevastava M. (2011).** “*Medical Diagnisis using Backpropagation Algorithma*” :*Internationla Journal, ISSN:2250-2459.*
- Kusmaryanto S. (2014).** “Jaringan Syaraf Tiruan *Backpropagation* untuk Pengenalan Wajah Metode Ekstraksi Fitur Berbasis Histogram”.
- Madontang Z.A. (2013).** “ Jaringan Syaraf Tiruan dengan Algoritma *Backpropagation* untuk menentukan kelulusan Sidang Skripsi”.
- Malik B.A., Naqash A., and Bhat. (2016).** “*Backpropagation Artificial Neural Network for Determination of Glocose Concentration from Near-Infrared Spectra*”: *India ICACCI, September 21-24.*
- Mustaqim J.R., Ratianingsih R., and Lusiyantri. (2017).** “Prediksi Kualitas Air Bersih PDAM Kota Palu Menggunakan Metode *Backpropagation*”.
- Pratama R.A., and Anifah L. (2011).** “Peramalan Beban Listrik Jangka Panjang Provinsi D.I Yogyakarta menggunakan *Neural Network Bacpropagation*”.
- Setiabudi D. (2015).** “Sistem Informasi Peramalan Beban Listrik Jangka Panjang di Kabupaten Jember Menggunakan Jaringan Syaraf Tiruan *Backpropagation*”: *ISSN:2476-9754.*
- Siang J.J. (2004).** “Jaringan Syaraf Tiruan & Pemogramannya Menggunakan Matlab”: Andi.
- Tanjung D.H. (2015).** “Jaringan Syaraf Tiruan dengan *Backpropagation* untuk Memprediksi Penyakit Asma” : *ISSN: 2354-5771.*
- Tisan A., and Chin J. (2016).** “*An End User Platform For FPGA-Based Design and Rapid Prototyping of Feed Forward Artificial Neural Networks with on-Chip Backpropagation Learning*”.
- Wuryandari M.D., and Afrianto I. (2012).** “Perbandingan Metode Jaringan Syaraf Tiruan *Backpropagation* dan *Learning Vector Quantization* pada Pengenalan Wajah”.
- Yani E. (2005).** “Pengantar Jaringan Syaraf Tiruan ”