

ANALISIS FUNGSI AKTIVASI SIGMOID BINER DAN SIGMOID BIPOLAR DALAM ALGORITMA *BACKPROPAGATION* PADA PREDIKSI KEMAMPUAN SISWA

Julpan^{1*}, Erna Budhiarti Nababan¹ & Muhammad Zarlis¹

¹Program S2 Teknik Informatika Universitas Sumatera Utara, Medan, Indonesia

*E-Mail : Julpan065@gmail.com

ABSTRAK

Metode *backpropagation* merupakan metode yang sangat baik dalam menangani masalah pengenalan pola-pola kompleks. Di dalam jaringan *backpropagation*, setiap unit yang berada di lapisan input terhubung dengan setiap unit yang ada di lapisan tersembunyi. Hal serupa berlaku pula pada lapisan tersembunyi. Setiap unit yang ada pada lapisan tersembunyi terhubung dengan setiap unit yang ada di lapisan *output*. Fungsi aktivasi yang digunakan di dalam metode *backpropagation* adalah fungsi sigmoid biner (*binary sigmoid function*), sigmoid bipolar (*bipolar sigmoid function*). Karakteristik yang harus dimiliki fungsi aktivasi tersebut adalah kontinyu dan tidak menurun secara monoton. Dari hasil penelitian maka didapatkan bahwa penggunaan fungsi aktivasi sigmoid biner memiliki tingkat rata-rata akurasi yang lebih baik dari sigmoid bipolar. Akan kecepatan perhitungannya lebih lambat dibandingkan dengan sigmoid bipolar. Fungsi aktivasi sigmoid bipolar dipilih berdasarkan asumsi *heuristik* bahwa nilai rentang [0, 1] yang dihasilkan oleh fungsi sigmoid biner (*unipolar*) kurang bagus dibandingkan dengan rentang [-1, 1] yang dihasilkan fungsi yang sigmoid bipolar. Hasil ini disebabkan karena nilai ekstrim 0 yang dihasilkan fungsi sigmoid biner kurang memberikan pengaruh pada penghitungan nilai pada *neuron*, dibandingkan dengan nilai ekstrim -1 yang dihasilkan oleh fungsi sigmoid bipolar.

Kata Kunci : Jaringan saraf tiruan (*neural network*), *backpropagation*, fungsi aktivasi sigmoid biner dan sigmoid bipolar.

PENDAHULUAN

Ketidak lulusan ketika mengikuti Ujian Nasional menjadi permasalahan rutin tiap tahun. Permasalahan ini dirasakan oleh siswa, wali siswa, orang tua, penyelenggara pendidikan, dinas pendidikan, BNSP, LPMP dan Kemendiknas. Pada tahun 2012 ada sekitar 17.269 sekolah tingkat SMA dan Madrasah Aliyah. Sekolah dengan tingkat kelulusan UN sebesar 100% berjumlah 15.024 sekolah atau sekitar 87%. Sedangkan sekolah dengan tingkat kelulusan UN 0% berjumlah 4 sekolah atau sekitar 0,02%. Kalau dibandingkan dengan tahun 2011, angka kelulusan ujian nasional (UN) tingkat SMA/ MA tahun 2012 mengalami penurunan sebesar 0,28%. Pada tahun 2011 angka kelulusan UN mencapai 99,22%, sekarang menjadi 99,50%.

Siswa pasti menginginkan mendapat nilai terbaik dan lulus ujian nasional. Kemampuan siswa untuk mendapatkan nilai terbaik dipengaruhi oleh faktor-faktor ekonomi dan psikologis. Faktor-faktor tersebut berinteraksi satu sama lain dengan cara yang sangat kompleks. Umumnya sangat sulit untuk memprediksi kemampuan siswa. Sehingga dibutuhkan suatu metode atau cara untuk memprediksi kemampuan siswa tersebut untuk membantu guru untuk menentukan nilai siswa dan mata pelajaran mana yang perlu dilakukan tindakan khusus agar siswa tersebut mendapatkan nilai terbaik agar diperoleh nilai UN tertinggi. Maka dalam penelitian ini akan digunakan suatu model Jaringan Saraf Tiruan (*Neural Network*) untuk memprediksi kemampuan siswa, selanjutnya mentransformasikan hasil yang diperoleh tersebut dengan suatu pendekatan algoritma *backpropagation*.

Hasil dari model ini digunakan untuk memprediksi kemampuan siswa dalam UN ditahun mendatang. Jika berdasarkan prediksi angka ketidaktulusan siswa dalam UN tinggi, dapat diambil langkah-langkah antisipatif sebelum pelaksanaan UN antara lain pengayaan materi pembelajaran, *try out*, penambahan jam pembelajaran, sehingga angka ketidaktulusan bisa diminimalisir.

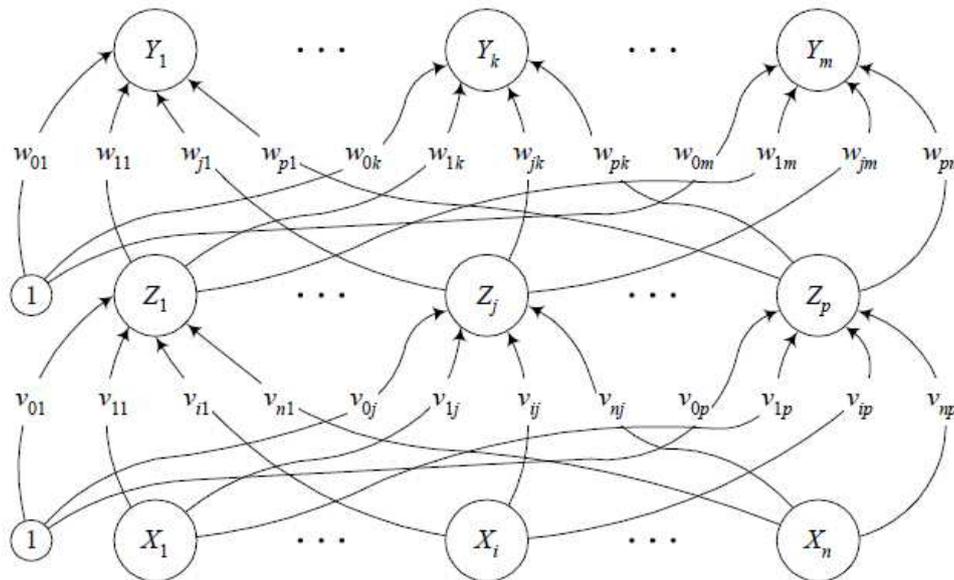
Jaringan Saraf Tiruan Algoritma Perambatan Balik (*backpropagation*)

Jaringan Saraf Tiruan memiliki kemampuan untuk dapat melakukan pembelajaran dan pendeteksian terhadap suatu pola. Proses pembelajaran merupakan suatu metoda untuk proses pengenalan suatu objek yang sifatnya kontinuitas yang selalu direspon secara berbeda dari setiap proses pembelajaran tersebut. Tujuan dari pembelajaran ini sebenarnya untuk memperkecil tingkat suatu error dalam pengenalan suatu objek.

Jaringan saraf lapis jamak (*multilayer network*) dengan pelatihan terbimbing (*supervised*) antara lain adalah jaringan perambatan-balik (*backpropagation*). Metode pelatihan perambatan-balik (*backpropagation*) secara sederhana adalah metode penurunan gradien (*gradient descent*) untuk meminimalkan total galat kuadrat keluaran. Aplikasi jaringan ini melibatkan pemetaan sekumpulan masukan terhadap sekumpulan target keluaran, jadi termasuk kategori jaringan dengan pelatihan terbimbing.

Pelatihan jaringan perambatan balik melibatkan tiga tahap yaitu umpan-maju pelatihan pola masukan, komputasi dan perambatan-balik galat, serta perubahan bobot. Setelah pelatihan, aplikasi jaringan hanya melibatkan tahap komputasi umpan-maju. Walaupun proses pelatihan jaringan berlangsung relatif lambat, namun jaringan yang telah dilatih dapat menghasilkan keluaran dengan sangat cepat.

Jaringan saraf lapis jamak (*multilayer network*) dengan lapisan tersembunyi (unit Z) diperlihatkan pada Gambar 1. Unit keluaran (unit Y) dan unit-unit tersembunyi serta bias diperlihatkan. Bias pada unit keluaran Y_k dilambangkan dengan W_{0k} , bias pada unit tersembunyi Z_j dilambangkan dengan V_{0j} . Istilah bias-bias ini berlaku sebagai bobot pada hubungan unit-unit dengan keluaran selalu satu. Hanya aliran informasi umpan-maju yang diperlihatkan pada gambar. Selama fase pelatihan perambatan-balik, sinyal dikirim pada arah yang berlawanan.



Gambar 1. Jaringan saraf tiruan perambatan-balik (*backpropagation*).

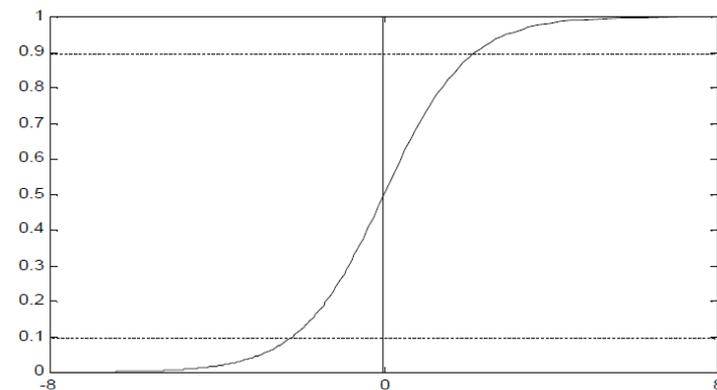
Fungsi Aktivasi

Fungsi aktivasi merupakan fungsi yang digunakan pada jaringan saraf untuk mengaktifkan atau tidak mengaktifkan *neuron*. Karakteristik yang harus dimiliki oleh fungsi aktivasi jaringan perambatan balik antara lain harus kontinu, terdiferensialkan, dan tidak menurun secara monotonis

(*monotonically non-decreasing*). Lebih lanjut, untuk efisiensi komputasi, turunan fungsi tersebut mudah didapatkan dan nilai turunannya dapat dinyatakan dengan fungsi aktivasi itu sendiri. Fungsi aktivasi yang di analisis adalah sigmoid biner dan sigmoid bipolar.

Fungsi aktivasi sigmoid biner memiliki nilai pada *range* 0 sampai 1. Ditunjukkan pada gambar 2, didefenisikan sebagai berikut :

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

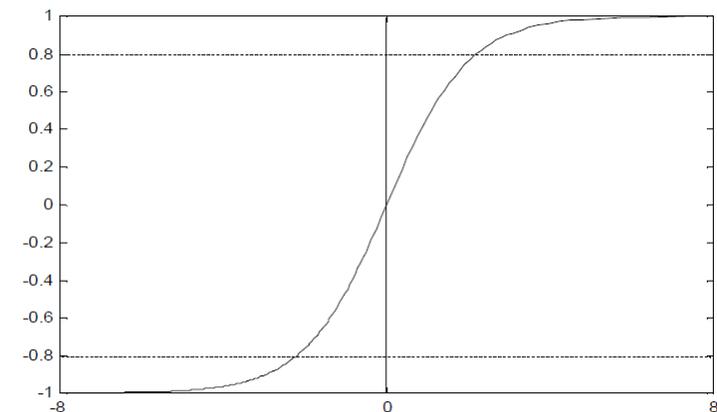


(Sumber : Fausett, L., 1994)

Gambar 2. Fungsi aktivasi *sigmoid biner*.

Fungsi aktivasi sigmoid bipolar hampir sama dengan fungsi sigmoid biner, hanya saja *output* dari fungsi ini memiliki *range* antara 1 sampai -1. Ditunjukkan pada gambar 3, didefenisikan sebagai berikut :

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



(Sumber : Fausett, L., 1994)

Gambar 3. Fungsi aktivasi *sigmoid biner*.

Algoritma *Backpropagation*

Backpropagation bekerja melalui proses secara iteratif dengan menggunakan sekumpulan contoh data (*data training*), membandingkan nilai prediksi dari jaringan dengan setiap contoh data. Dalam setiap proses, bobot relasi dalam jaringan dimodifikasi untuk

meminimalkan nilai *Mean Square Error* (MSE) antara nilai prediksi dari jaringan dengan nilai sesungguhnya. Modifikasi relasi jaringan saraf tersebut dilakukan dalam arah mundur, dari *output layer* hingga *layer* pertama dari *hidden layer* sehingga metode ini disebut *backpropagation*.

Langkah-langkah yang dilakukan pada prosedur pelatihan menggunakan *backpropagation* adalah:

- Langkah 0** : Inisialisasi bobot keterhubungan antara *neuron* dengan menggunakan bilangan acak kecil (-0.5 sampai +0.5).
- Langkah 1** : Jika kondisi henti tidak terpenuhi, lakukan langkah 2 - 9
- Langkah 2** : Untuk setiap pola input, lakukan step 3-8

Feedforward :

Pembelajaran pada layer 1

- Langkah 3** : Setiap *neuron* masukan (X_i $i= 1...n$) menerima sinyal masukan x_i dan menyebarkannya ke semua *neuron* pada lapis tersembunyi.
- Langkah 4** : Setiap *neuron* pada lapis tersembunyi (Z_j , $j=1...p$) menjumlahkan bobot-bobot sinyal masukan,

$$z_in_j = \sum_{i=1}^n x_i v_{ij}$$

dan menerapkannya pada fungsi aktivasi untuk menghitung sinyal keluarannya,
 $z_j = f(z_in_j)$

- Langkah 5** : Setiap *neuron* keluaran (Y_k , $k=1...m$) menjumlahkan bobot sinyal yang masuk,

$$y_in_k = \sum_{j=1}^p z_j w_{jk}$$

dan mengaplikasikan fungsi aktivasinya untuk menghitung sinyal yang akan dikeluarkannya

$$y_k = f(y_in_k)$$

Backpropagasi error :

- Langkah 6** : Setiap *neuron* keluaran (Y_k , $k=1...m$) menerima sebuah pola target yang berhubungan dengan pola masukan pelatihan dan menghitung kesalahan informasi dengan mengalikan dengan turunan fungsi aktivasinya

$$\delta_k = (t_k - y_k) f'(y_in_k)$$

menghitung koreksi bobot (yang akan digunakan untuk memperbaiki W_{jk} nanti),

$$\Delta w_{jk} = (\alpha \delta_k z_j)$$

- Langkah 7** : Setiap *neuron* tersembunyi (Z_j , $j=1...p$) menjumlahkan bobot setiap *neuron* yang telah dikali dengan kesalahan informasinya,

$$\delta_in_j = \sum_{k=1}^m \delta_k w_{jk}$$

- Langkah 7 (lanjutan)** : mengalikan dengan turunan fungsi aktivasinya untuk menghitung kesalahan informasinya,

$$\delta_j = \delta_{in_j} f'(z_{in_j})$$

menghitung koreksi bobotnya (yang akan digunakan untuk memperbaiki V_{ij} nanti)

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

Perbaiki bobot untuk proses selanjutnya

Langkah 8 : Setiap *neuron* keluaran ($Y_k, k=1 \dots m$) memperbaiki bobotnya ($j=0 \dots p$)

$$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk}$$

Setiap *neuron* tersembunyi ($Z_j, j=1 \dots p$) memperbaiki bobotnya

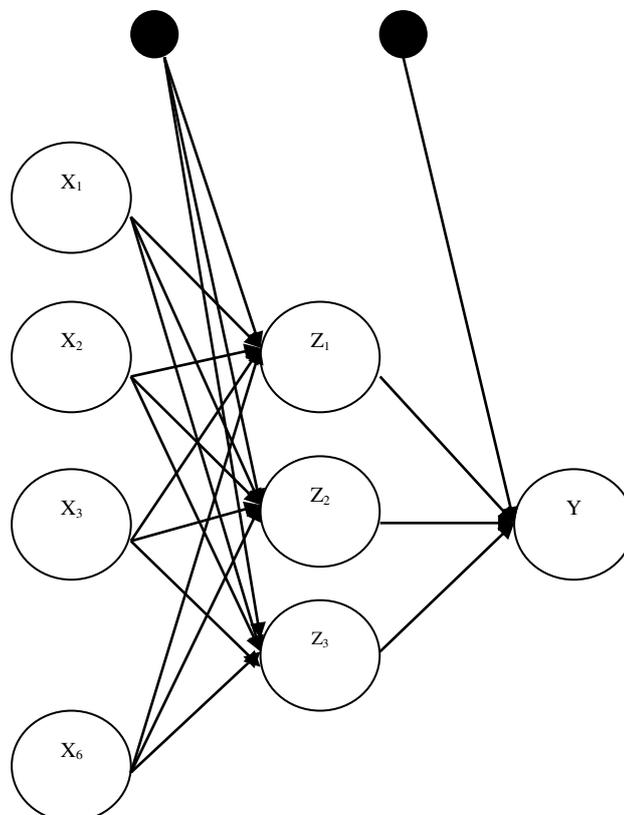
$$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij}$$

Langkah 9 : Lakukan pengujian kondisi henti.

METODE PENELITIAN

Arsitektur Jaringan

Perancangan arsitektur sistem adalah merancang jumlah *input node*, *output node*, *hidden node* serta fungsi aktivasi yang digunakan. Dalam penelitian ini jumlah *hidden layer* ditentukan dengan pengertian pembelajaran yang tercepat dan terbaik itulah yang akan menentukan jumlah *hidden layer* tersebut.



Gambar 4. Arsitektur jaringan *backpropagation*

Data Yang Digunakan

Data yang diperlukan dalam prediksi kemampuan peserta didik menggunakan jaringan saraf tiruan ini adalah data hasil penelitian murni dengan mengambil variable *input* berupa nilai raport per mata pelajaran, yang meliputi : Bahasa Indonesia disimpan pada variable X1, Bahasa Inggris disimpan pada variable X2, Matematika disimpan pada variable X3, Kimia/Sosiologi disimpan pada variable X4, Fisika/Ekonomi disimpan pada variable X5 dan Biologi/Geografi disimpan pada variable X6, sedangkan *output* (target) nilai UN.

Dalam *backpropagation* jika data yang digunakan terlalu besar maka sistem akan membutuhkan waktu yang sangat lama mencapai *konvergensi* atau menemukan pola untuk mencapai target. Sebaliknya jika data terlalu sedikit maka akan menyebabkan sistem mungkin tidak dapat mempelajari atau mengenali taburan data dengan baik sehingga sulit untuk mencapai target.

Pembagian data yang dilakukan dalam penelitian ini adalah :

1. Data untuk pelatihan (2007-2012)
2. Data untuk pengujian (2009-2012)

Normalisasi Data

Normalisasi dilakukan untuk menyesuaikan nilai data dengan *range* fungsi aktivasi yang digunakan dalam jaringan. Tujuannya adalah agar diperoleh interval data yang sesuai dengan fungsi aktivasi yang digunakan yaitu fungsi aktivasi *sigmoid biner* dan *sigmoid bipolar*.

Tabel 1. Data nilai raport sebelum dinormalisasi

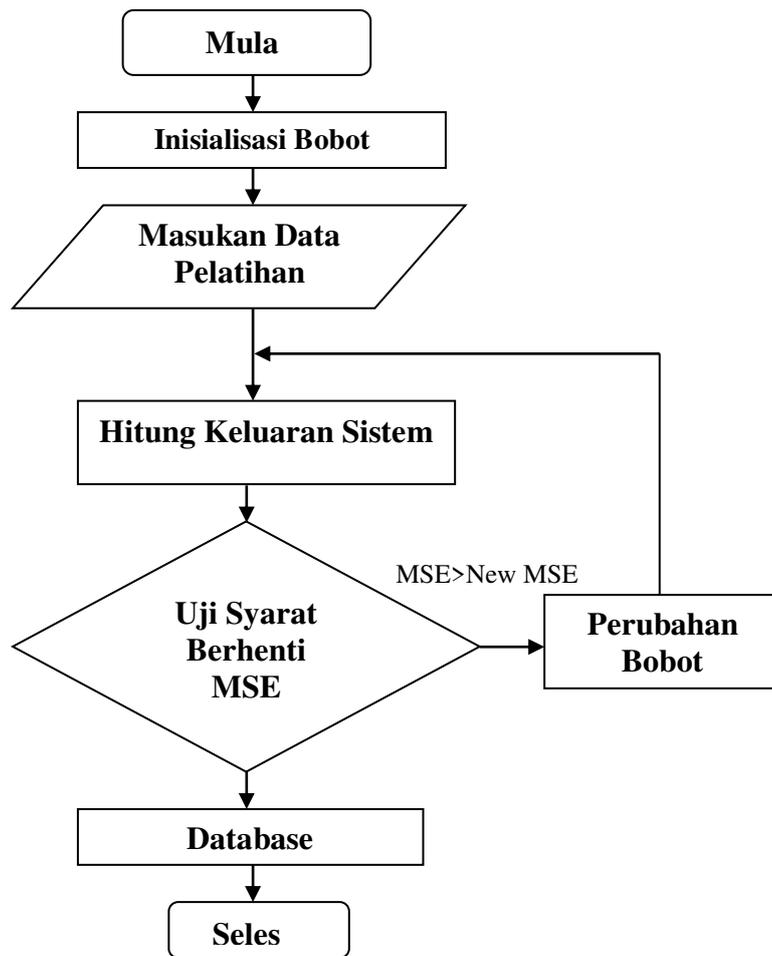
Nilai Raport						Sem
B. Ind	B. Ing	MM	Fis	Kim	Bio	
67.00	74.00	71.00	72.00	70.00	73.00	I
78.00	85.00	82.00	71.00	77.00	82.00	II
75.00	69.00	76.00	63.00	70.00	73.00	III
83.00	86.00	85.00	78.00	85.00	84.00	IV
97.00	93.00	83.00	87.00	85.00	87.00	V
92.00	88.00	82.00	80.00	82.00	94.00	VI

Tabel 2. Data nilai raport setelah dinormalisasi

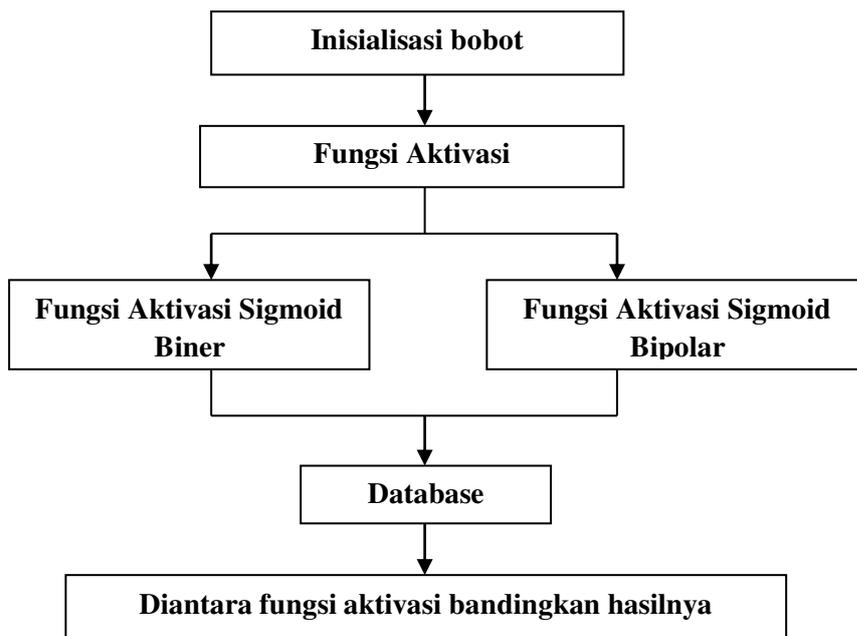
Nilai Raport						Sem
B. Ind	B. Ing	MM	Fis	Kim	Bio	
0.01	0.39	0.38	0.40	0.34	0.38	I
0.33	0.63	0.67	0.38	0.51	0.51	II
0.24	0.29	0.51	0.16	0.34	0.38	III
0.47	0.65	0.75	0.57	0.71	0.54	IV
0.87	0.79	0.69	0.82	0.71	0.59	V
0.73	0.69	0.67	0.62	0.63	0.69	VI

Perancangan Sistem

Perancangan sistem digunakan untuk menggambarkan langkah kerja dari sistem yang akan dibuat. perancangan sistem penelitian dapat dilihat pada gambar 5.



Gambar 5. Perancangan sistem penelitian



Gambar 6. Perbandingan fungsi aktivasi sigmoid biner dan sigmoid bipolar

HASIL DAN PEMBAHASAN

Pembobotan Awal

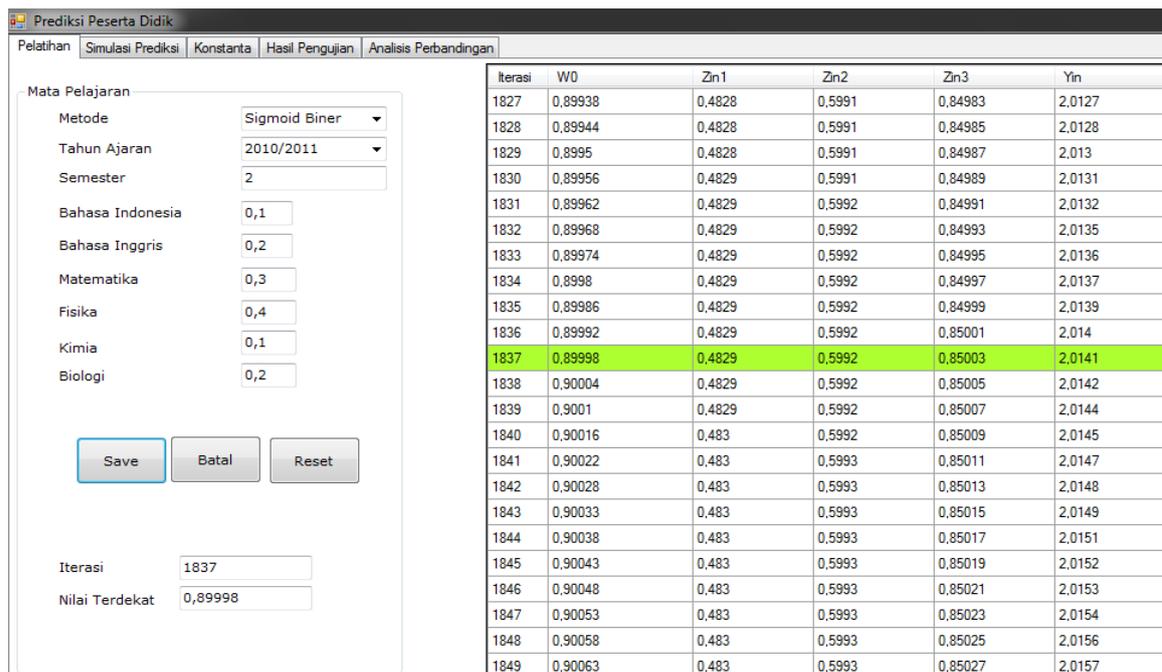
Untuk inialisasi bobot awal sistem akan melakukan pembobotan secara random. Untuk menghentikan program maka terdapat 2 cara yang dapat digunakan yaitu dengan menentukan *Epoch* dan bobot *error* telah tercapai. Pembobotan awal dengan cara random dilakukan langsung oleh program. Nilai yang digunakan berada dalam interval -0,5 sampai 0,5. Sebelum pengisian data maka pembobotan dan pengisian konstanta dilakukan terlebih dahulu. Pada tahap pembobotan akan ditentukan bobot dan bias yang akan digunakan untuk pelatihan.



Gambar 7. Tampilan program saat dilakukan pembobotan

Pelatihan (Training) Data

Pada pelatihan *backpropagation* ini, inialisasi untuk input $X_1, X_2 .. X_6$ bobot awal yang menghubungkan *node* pada lapisan *input* ke *hidden layer* ($V_{11}, V_{12}, V_{13}, ..V_{63}$) dan bobot bias V_{01}, V_{02}, V_{03} menggunakan metode random. Untuk bobot awal yang menghubungkan *node* pada *hidden layer* ke *output layer* (W_1, W_2, W_3) dan bobot bias W_0 ditentukan secara random.



Gambar 8. Tampilan program pelatihan menggunakan sigmoid biner.

The screenshot shows the 'Prediksi Peserta Didik' application. On the left, there is a configuration panel for training parameters. The 'Mata Pelajaran' (Subject) is set to 'Sigmoid Bipolar'. Other parameters include 'Tahun Ajaran' (2010/2011), 'Semester' (2), and various subject weights (Bahasa Indonesia: 0,1; Bahasa Inggris: 0,2; Matematika: 0,3; Fisika: 0,4; Kimia: 0,1; Biologi: 0,2). Below these are 'Save', 'Batal', and 'Reset' buttons. At the bottom left, 'Iterasi' is set to 130 and 'Nilai Terdekat' is 0,89986.

The main area displays a table of iteration results:

Iterasi	W0	Zin1	Zin2	Zin3	Yin
120	0,87561	0,4616	0,5782	0,82134	1,2754
121	0,87808	0,4617	0,5786	0,82207	1,2789
122	0,88054	0,4618	0,579	0,82279	1,2823
123	0,88299	0,4619	0,5794	0,82351	1,2858
124	0,88543	0,462	0,5798	0,82423	1,2892
125	0,88786	0,4621	0,5802	0,82495	1,2926
126	0,89028	0,4622	0,5806	0,82567	1,296
127	0,89269	0,4623	0,5809	0,82639	1,2994
128	0,89509	0,4624	0,5813	0,82711	1,3027
129	0,89748	0,4625	0,5817	0,82783	1,3061
130	0,89986	0,4626	0,5821	0,82855	1,3095
131	0,90223	0,4627	0,5825	0,82926	1,3128
132	0,90459	0,4628	0,5829	0,82997	1,3161
133	0,90694	0,4629	0,5833	0,83068	1,3194
134	0,90929	0,463	0,5836	0,83139	1,3227
135	0,91163	0,4631	0,584	0,8321	1,326
136	0,91396	0,4632	0,5844	0,83281	1,3293
137	0,91628	0,4633	0,5848	0,83352	1,3326
138	0,91859	0,4633	0,5852	0,83423	1,3359
139	0,92089	0,4635	0,5855	0,83494	1,3391
140	0,92318	0,4636	0,5859	0,83564	1,3423
141	0,92546	0,4637	0,5863	0,83634	1,3456
142	0,92773	0,4638	0,5867	0,83704	1,3488

Gambar 9. Tampilan program pelatihan menggunakan sigmoid bipolar

Pengujian Data

Setelah data dilatih maka akan diketahui bias dari *hidden layer* menuju *output layer*, dan nilai *learning rate* yang tepat. Langkah selanjutnya setelah di training adalah dilakukan pengujian dengan menginput nilai bias dan *learning rate* yang sama pada saat *training*. Proses saat dilakukan pengujian dapat dilihat pada gambar 10.

The screenshot shows the 'Hasil Pengujian' (Test Results) tab. It displays two tables comparing predicted values against target values for two different years (2010/2011 and 2011/2012) across six semesters. The 'Skala (%)' (Scale %) indicates the accuracy of the predictions.

Metode : Sigmoid

Tahun Ajaran	Semester 1	Semester 2	Semester 3	Semester 4	Semester 5	Semester 6	Target
2010/2011	0,868	0,84128	0,89997	0,89997	0,89999	0,84277	0,9
Skala (%)	96,44%	93,48%	100%	100%	100%	93,64%	100%
2011/2012	0,89999	0,89999	0,88798	0,89999	0,8932	0,89485	0,9
Skala (%)	100%	100%	98,66%	100%	99,24%	99,43%	100%

Metode : Sigmoid Bipolar

Tahun Ajaran	Semester 1	Semester 2	Semester 3	Semester 4	Semester 5	Semester 6	Target
2010/2011	0,8999	0,89978	0,89923	0,89936	0,89932	0,89979	0,9
Skala (%)	99,99%	99,98%	99,91%	99,93%	99,92%	99,98%	100%
2011/2012	0,89951	0,89935	0,89965	0,89976	0,89963	0,89946	0,9
Skala (%)	99,95%	99,93%	99,96%	99,97%	99,96%	99,94%	100%

Buttons: Tampil Data, Reset

Gambar 10. Tampilan program saat dilakukan pengujian

Dalam sistem ini pengujian dilakukan sebanyak 2 tahap. Untuk prediksi data tahun 2012 maka data *input* yang digunakan adalah tahun 2009 sampai dengan 2011. Hasil keakuratan pada tahap pengujian akan mempengaruhi keakuratan pada saat membuat prediksi untuk tahun-tahun berikutnya. Sehingga pada proses pengujian harus dicari nilai keakuratan yang tertinggi.

Hasil Pelatihan Menggunakan Fungsi Aktivasi Sigmoid biner

Hasil pelatihan disusun dalam sebuah tabel sehingga dapat dibandingkan untuk menentukan kondisi *epoch* tercepat. Adapun hasil perubahan dari *epoch* dan nilai *learning rate* seperti terlihat pada tabel 3.

Tabel 3. Hasil pelatihan menggunakan fungsi aktivasi *sigmoid biner*

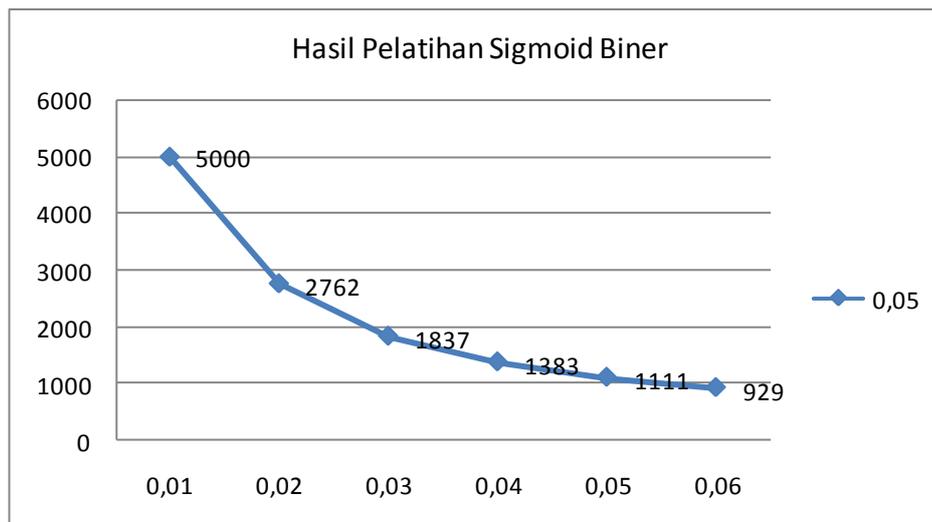
Bias (W_0)	Learning Rate						Maksimum Epoch
	0,01	0,02	0,03	0,04	0,05	0,06	
0,1	TK	TK	TK	TK	TK	TK	1000
0,3	TK	TK	TK	TK	TK	TK	
0,5	TK	TK	TK	TK	TK	TK	
0,1	TK	TK	TK	TK	TK	TK	5000
0,3	TK	TK	TK	TK	TK	TK	
0,5	TK	2762	1837	1383	1111	929	
0,1	TK	TK	TK	TK	TK	TK	10000
0,3	TK	TK	TK	TK	TK	TK	
0,5	5566	2762	1837	1883	1111	729	

Keterangan :

- LR : *Learning Rate*
- TK : Tidak *Konvergen*

Nilai *node* pada *input layer*, nilai bobot sangat mempengaruhi tingkat kecepatan dan keakuratan pelatihan. Jika data yang digunakan terlalu besar maka sistem akan membutuhkan waktu yang sangat lama mencapai *konvergensi* atau menemukan pola untuk mencapai target. Sebaliknya jika data terlalu sedikit maka akan menyebabkan sistem mungkin tidak dapat mempelajari atau mengenali pola data dengan baik sehingga sulit untuk mencapai target.

Bias (W_0), *learning rate* (α) juga berpengaruh sangat besar dalam meningkatkan atau menurunkan *input net* dari fungsi aktivasi, tetapi tergantung nilai dari bias apakah positif atau negative, jika positif akan meningkatkan, dan jika negatif akan menurunkan. Semakin tinggi nilai α dan bias maka sistem akan semakin cepat mencapai *konvergensi* atau menemukan pola untuk mencapai target. Hasil pelatihan dengan menggunakan fungsi aktivasi *sigmoid biner*, jaringan mengalami *konvergensi* tercepat berada pada *epoch* ke 929. Adapun grafik pelatihan fungsi aktivasi *sigmoid biner* seperti terlihat pada gambar 11.



Gambar 11. Grafik pelatihan fungsi aktivasi *sigmoid biner*

Dari gambar grafik dapat dilihat hasil pelatihan sebanyak 150 data dengan menggunakan fungsi aktivasi *sigmoid biner* yang dilatihkan oleh jaringan mencapai hasil 100%. Hal tersebut dikarenakan hasil keluaran yang ditampilkan sesuai dengan target yang sudah ditentukan (0,9).

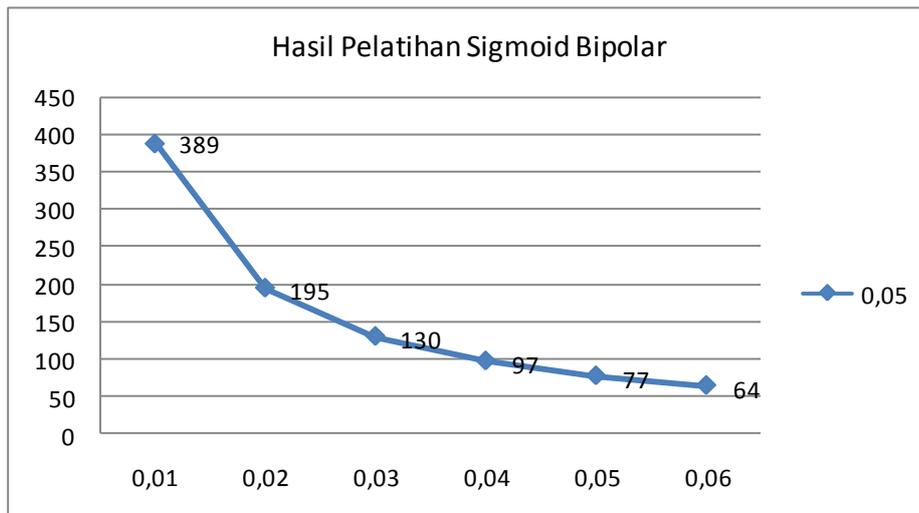
Hasil Pelatihan Menggunakan Fungsi Aktivasi Sigmoid Bipolar

Hasil pelatihan disusun dalam sebuah tabel sehingga dapat dibandingkan untuk menentukan kondisi *epoch* tercepat. Adapun hasil perubahan dari *epoch* dan nilai *learning rate* seperti terlihat pada tabel 4.

Tabel 4. Hasil pelatihan menggunakan fungsi aktivasi *sigmoid bipolar*

Bias (W ₀)	Learning Rate						Maksimum Epoch
	0,01	0,02	0,03	0,04	0,05	0,06	
0,1	TK	TK	340	TK	TK	TK	1000
0,3	702	TK	233	TK	TK	TK	
0,5	454	TK	150	TK	TK	TK	
0,1	1027	TK	340	TK	TK	TK	5000
0,3	702	TK	233	TK	TK	TK	
0,5	454	TK	150	TK	TK	TK	
0,1	786	395	263	187	158	131	10000
0,3	575	192	144	115	96	289	
0,5	389	195	130	97	77	64	

Proses pelatihan sama seperti menggunakan fungsi aktivasi *sigmoid biner*. Nilai *node* pada *input layer*, nilai bobot, bias (b_k), *learning rate* (α) sangat mempengaruhi untuk mencapai *konvergensi* atau menemukan pola untuk mencapai target. Berdasarkan tabel 6 Hasil pelatihan dengan menggunakan fungsi aktivasi *sigmoid bipolar*, jaringan mengalami *konvergensi* tercepat berada pada *epoch* ke 64. Adapun grafik pelatihan fungsi aktivasi *sigmoid bipolar* seperti terlihat pada gambar 12.



Gambar 12. Grafik pelatihan fungsi aktivasi *sigmoid bipolar*

Pembahasan dalam penelitian ini akan dilakukan dengan membandingkan pengaruh bobot, bias, maksimum *epoch* kedalam sistem untuk pengujian terhadap prediksi kemampuan siswa. Untuk meningkatkan sinyal keluaran dari suatu lapisan maka salah satu cara adalah dengan menambah bias. Penambahan bias dari lapisan input menuju *hidden layer* akan menambah nilai bobot pada *hidden layer*.

Pengujian Terhadap Nilai Raport

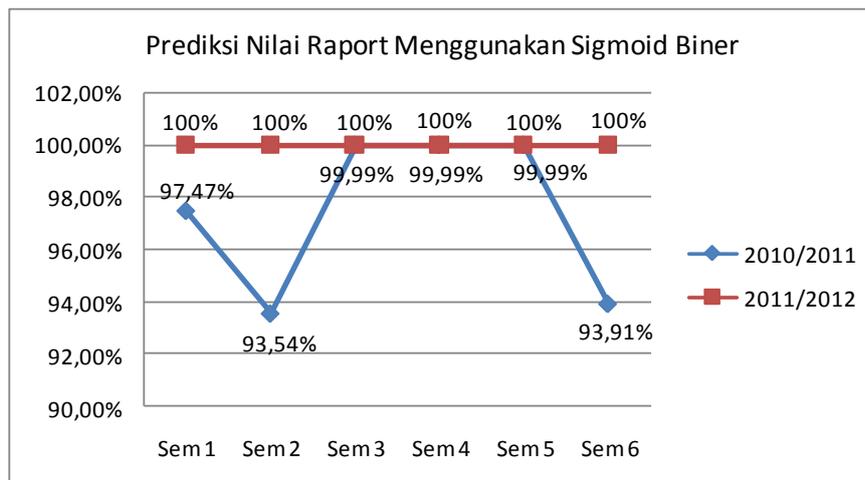
Setelah tahap pelatihan selesai dilakukan maka program akan diuji. Pengujian ini berfungsi untuk mengetahui apakah sistem mampu mengenali data yang digunakan sebagai target. Semakin akurat sistem mengenali target pada tahap pengujian maka hasil prediksi yang diperoleh juga semakin akurat. Dalam penelitian ini akan dilakukan pengujian terhadap data dengan mengubah nilai *learning rate*, bias dan *epoch* untuk melihat keakuratan prediksi.

Pengujian Terhadap Nilai Raport

Hasil pengujian terhadap nilai raport mampu membuat prediksi dengan tingkat keakuratan sampai dengan 100%. Pengujian terhadap nilai raport dengan mengubah nilai *learning rate*, bias dan pembobotan untuk melihat keakuratan hasil prediksi. Maksimum *epoch* yang digunakan adalah 9000 *epoch*. Hal ini didasarkan karena pada *epoch* ke-9000 batas *error* yang digunakan tercapai, yaitu 0,01. Dalam tabel 5 dapat dilihat pengujian terhadap nilai raport yang telah dilakukan pada sistem dengan pembobotan secara random.

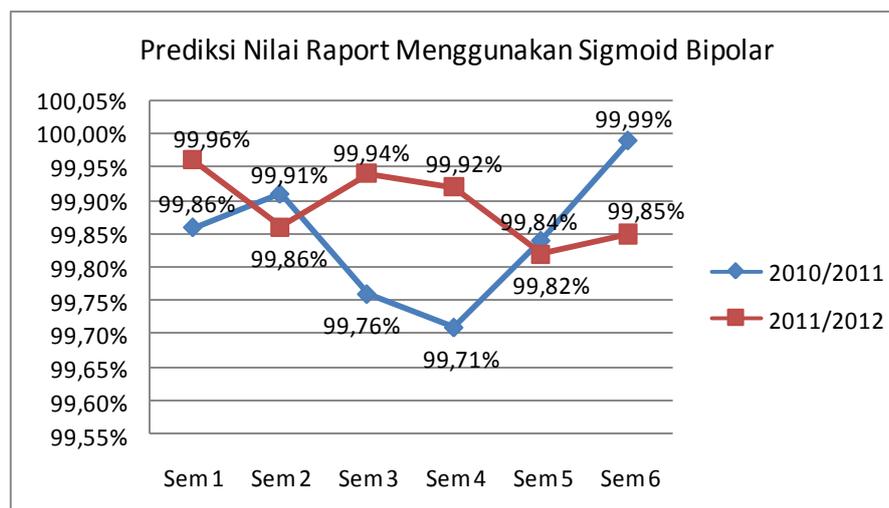
Tabel 5. Pengujian terhadap nilai raport yang telah dilakukan dengan pembobotan secara random

NR	Sigmoid Biner			Epoch
	2010/2011		2011/2012	
Sem 1	0,87722	97,47%	0,89998	100%
Sem 2	0,84185	93,54%	0,89998	100%
Sem 3	0,89992	99,99%	0,89999	100%
Sem 4	0,89993	99,99%	0,89998	100%
Sem 5	0,89995	99,99%	0,89999	100%
Sem 6	0,84523	93,91%	0,89999	100%



Gambar 13. Hasil prediksi nilai raport menggunakan fungsi aktivasi *sigmoid biner*

NR	Sigmoid Bipolar				Epoch
	2010/2011		2011/2012		
Sem 1	0,89874	99,86%	0,89962	99,96%	9000
Sem 2	0,89916	99,91%	0,89878	99,86%	
Sem 3	0,8978	99,76%	0,89945	99,94%	
Sem 4	0,89741	99,71%	0,89932	99,92%	
Sem 5	0,89858	99,84%	0,89834	99,82%	
Sem 6	0,89987	99,99%	0,89866	99,85%	



Gambar 14. Hasil prediksi menggunakan fungsi aktivasi *sigmoid bipolar*

Berdasarkan tabel 5 diatas maka pengujian tertinggi keakuratannya adalah 0,89999 (100%), dan berhenti pada epoch ke-4431. Dan pengujian yang mengalami konvergensi tercepat berada pada *epoch* ke 368 (99,76%). Maka berdasarkan hasil pengujian dengan tingkat keakuratan paling tinggi prediksi kemampuan siswa berdasarkan nilai raport diperoleh dengan menggunakan fungsi aktivasi *sigmoid biner*. Sedangkan hasil pengujian dengan tingkat kecepatan (*konvergen*) paling tinggi

prediksi kemampuan siswa berdasarkan nilai raport diperoleh dengan menggunakan fungsi aktivasi *sigmoid bipolar*.

Fungsi aktivasi *sigmoid bipolar* dipilih berdasarkan asumsi *heuristik* bahwa nilai rentang [0, 1] yang dihasilkan oleh fungsi *sigmoid biner (unipolar)* kurang bagus dibandingkan dengan rentang [-1, 1] yang dihasilkan fungsi yang *sigmoid bipolar*. Hasil ini disebabkan nilai ekstrim 0 yang dihasilkan fungsi *sigmoid biner* kurang memberikan pengaruh pada penghitungan nilai pada *neuron* berikutnya dibandingkan dengan nilai ekstrim -1 yang dihasilkan oleh fungsi *sigmoid bipolar*.

KESIMPULAN

Hasil pengujian dengan tingkat kecepatan (*konvergen*) paling tinggi prediksi kemampuan siswa berdasarkan nilai raport diperoleh dengan menggunakan fungsi aktivasi *sigmoid bipolar*. Disebabkan karena nilai ekstrim 0 yang dihasilkan fungsi *sigmoid biner* kurang memberikan pengaruh pada penghitungan nilai pada *neuron* berikutnya dibandingkan dengan nilai ekstrim -1 yang dihasilkan oleh fungsi *bipolar*. Apabila nilai *learning rate*, bias terlalu kecil atau terlalu besar proses pembelajaran berlangsung kurang efektif dan cenderung tidak *konvergen*. Keakuratan data sangat tergantung pada nilai *error* yang dihasilkan pada saat pelatihan. Semakin kecil nilai *error* yang dihasilkan maka solusi yang dihasilkan akan semakin akurat dan untuk mendapatkan nilai *error* yang semakin kecil diperlukan jumlah iterasi yang banyak pula.

DAFTAR PUSTAKA

- Agustin, M. 2012. *Penggunaan jaringan saraf tiruan backpropagation untuk seleksi penerimaan mahasiswa baru pada jurusan teknik komputer di politeknik negeri sriwijaya*. Program Pascasarjana Universitas Diponegoro Semarang.
- Arifin & Tumanan, O. 2011. Pengenalan Pola Sidik Jari Menggunakan Jaringan Syaraf Tiruan Dengan Metode Pembelajaran Backpropagation. *Jurnal Aplikasi Fisika Volume 7 No. 1 Februari 2011*.
- Fausett, L. 1994. *Fundamentals of Neural Networks : Architectures, Algorithms, and Applications*, Prentice Hall, New Jersey.
- Hermawan, A. 2006. *Jaringan Saraf Tiruan, Teori dan Aplikasinya*, Yogyakarta.
- Hidayatno. 2008. Identifikasi Tanda Tangan Menggunakan Jaringan Saraf Tiruan Perambatan Balik (Backpropagation), *Jurnal Teknologi*.
- Hsieh, & William, W. 1998. *Applying Neural Network Models to Prediction and Data Analysis in Meteorology and Oceanography*. Bulletin of the American Meteorological Society Vol. 79, No. 9, September 1998.
- Jang, JSR, Sun, CT, dan Mizutani, E. 1997. *Neuro Fuzzy and Soft Computing*. London. Prentice-Hall.
- Kurniawan. & Iwan, T. 1999. Metode peramalan memakai jaringan saraf buatan dengan cara *backpropagation*, *Jurnal teknologi industri vol iii no.2*.
- Lubis. 2005. Prediksi Harga Saham Dengan algoritma Hybrid Neural Network. *Seminar Nasional Aplikasi Teknologi Informasi 2005 (SNATI 2005) ISBN: 979-756-061-6 Yogyakarta, 18 Juni 2005*.
- Septiarini, A & Sya'baniah, N. 2012. Sistem Peramalan Jumlah Produksi Air PDAM Samarinda Menggunakan Jaringan Syaraf Tiruan Backpropagation. *Jurnal EKSPONENSIAL Volume 3, Nomor 1, Mei 2012*.
- Siang, J.J. 2005. *Jaringan Saraf Tiruan dan pemrograman Menggunakan Matlab*, Penerbit Andi, Yogyakarta.
- Suhardi, I. 2007. Evaluasi Pengaruh Fungsi Aktivasi Dan Parameter Kemiringannya Terhadap Unjuk Kerja Pengenalan Jaringan Syaraf Tiruan, *Jurnal Ilmiah Semesta Teknika, Vol. 10, No. 1, 2007: 53 – 68*.