

**PENERAPAN NEURAL NETWORK BERBASIS PARTICLE SWARM
OPTIMIZATION UNTUK SELEKSI ATRIBUT PENENTUAN MAHASISWA
DROP OUT**

Laila Septiana

Manajemen Informatika, AMIK BSI Jakarta
Jalan RS Fatmawati No. 24 Pondok Labu 12450, Telp.(021)7500282, 7500680
laila.lsp@bsi.ac.id

ABSTRACT

The drop-out student is a student whose their student status is revoked, in case of determined orders by the university. The height of drop-out students number will have the impact in scoring of accreditation in an university. therefore, it is necessary to know what factors the students have the drop-out status are. Most of algorithm comparison has been done in previous research. Otherwise, this research uses Neural Network with Particle Swarm Optimization based model to level up optimization value in selection attribute influencing prediction the students drop out.

Keywords : Drop Out Student, Neural Network, Particle Swarm Optimization

I. PENDAHULUAN

Berdasarkan hasil penelitian yang dilakukan oleh Hastuti (2012) menunjukkan bahwa perguruan tinggi menerima rata-rata 868.050 mahasiswa baru dan meluluskan rata-rata 451.168 mahasiswa setiap tahunnya. Jumlah lulusan perguruan tinggi ternyata hanya mencapai 51.97% dari jumlah mahasiswa baru setiap tahun. Artinya, terdapat 48,03% mahasiswa yang tidak diketahui statusnya. Ketidakjelasan status tersebut bisa jadi karena mahasiswa menempuh studi tidak tepat waktu, memiliki status non-aktif atau bahkan drop out. Berdasarkan matriks penilaian instrumen akreditasi program studi Badan Akreditasi Nasional Perguruan Tinggi, salah satu faktor penilaian untuk akreditasi sebuah perguruan tinggi adalah ketepatan waktu penyelesaian studi, proporsi mahasiswa yang menyelesaikan studi dalam batas masa studi. Dan untuk menentukan harkat serta peringkat sangat baik, presentase mahasiswa yang Drop Out atau mengundurkan diri harus $\leq 6\%$ dari jumlah keseluruhan mahasiswa.

II. KAJIAN LITERATUR

Metode prediksi mahasiswa non aktif pernah dilakukan oleh Hastuti dengan mengkomparasi beberapa algoritma klasifikasi data mining yaitu Logistic Regression, Decision Tree, naïve Bayes dan Neural Network. Dengan hasil penelitian akhir menyatakan bahwa Decision Tree merupakan

algoritma yang paling akurat, namun dalam hasil T-Test tidak dominan terhadap algoritma lain. Kotsiantis (2009), menyatakan bahwa sangat penting untuk mendeteksi mahasiswa yang cenderung drop out sebelum mereka memasuki pertengahan masa studi. Beberapa algoritma klasifikasi data mining digunakan yaitu decision tree, neural network, naïve bayes, instancebased learning, logistic regression dan support vector machine. Sedangkan Dekker (2009) menyebutkan bahwa.

Monitoring dan dukungan terhadap mahasiswa pada tahun pertama sangat penting dilakukan. Kurikulum yang berat, nilai, prestasi, kepribadian, latar belakang sosial juga dianggap mempengaruhi dalam kesuksesan akademik mahasiswa.

Decision tree mempunyai kelebihan dalam hal prediksi, karena struktur algoritmanya mudah dimengerti dan tingkat kesalahannya cukup kecil sedangkan kelemahan algoritma decision tree adalah keandalan cabang yang lebih rendah menjadi lebih buruk dari cabang atasnya, pohon keputusan yang dihasilkan tidak optimal dan tidak bisa menggunakan sample yang lebih besar (Sug, 2009), oleh karena itu tidak mudah untuk memahami pohon keputusan besar dan masalah overfitting data bisa terjadi dengan target data terbatas.

Particle Swarm Optimization (PSO) merupakan algoritma optimasi yang lebih efektif yang dapat memecahkan masalah yang ada pada algoritma neural network, yang pada

umumnya menggunakan algoritma backpropagation (Park, Lee & Choi, 2009).

III. METODE PENELITIAN

3.1 Data Mining

Menurut Witten(2005) Data mining adalah mengekstraksi suatu data untuk mendapatkan informasi penting yang bersifat implisit dan tidak diketahui sebelumnya.

McDonald dalam Liao (1999) “ Data Mining as one of the new tools that’s have accelerated the pace of yield improvement in IC (Integrated Circuit) manufacturing”.

Informasi dan pengetahuan yang diperoleh dapat digunakan untuk aplikasi mulai dari pasar analisis, deteksi penipuan dan retensi pelanggan, untuk pengendalian produksi dan ilmu pengetahuan eksplorasi (Han& Kamber, 2007)

Sedangkan Gartner Group mendefinisikan data mining sebagai serangkaian proses untuk mendapatkan korelasi, pola-pola serta trend baru dengan cara memilah-milah sejumlah data ukuran besar yang terdapat pada repositories, dengan menggunakan teknologi pengenalan pola serta statis dan matematika (Larose, 2005:2).

Berdasarkan definisi-definisi diatas maka dapat kita simpulkan bahwa data mining adalah kegiatan untuk mendapatkan informasi tersembunyi yang sebelumnya tidak diketahui. Peran utama data mining adalah klustering, klasifikasi, prediksi, analisis pola berurutan dan asosiasi. Masing-masing tugas mempunyai tool yang berbeda, klustering memerlukan kmeans, hirarchical clustering atau fuzzy Cmeans. Klasifikasi memerlukan analisa diskriminan, decision tree, neural network atau support vector machines.

3.2 Neural Network

Masalah yang dihadapi decision tree tersebut bisa dipecahkan oleh neural network karena memiliki kelebihan pada prediksi non linear, memiliki performance yang sangat baik di parallel processing dan kemampuan untuk mentoleransi kesalahan (Xiao& Shao, 2011). Hal ini sangat tepat untuk karakteristik data prediksi mahasiswa berpotensi drop out pada penelitian ini. “.... Algoritma neural network memiliki nilai accuracy yang cukup tinggi yaitu 94,56% dalam uji AUC mencapai hasil maksimal yaitu 0,976 dan bersifat dominan terhadap algoritma yang lain“(Hartati, 2012)

Penelitian dibidang neural network dimulai pada masa komputer digital. Mc Culloch dan Pitts (1943) mengemukakan

model perceptron dan algoritma pembelajaran pada tahun 1962 (Alpaydin,2010). Minsky dan paert (1969) menunjukkan keterbatasan single layer perceptron untuk menyelesaikan masalah yang nonlinearly separable. Kemudian Rumelhart, Hinton dan Williams (1986) yang mempresentasikan algoritma backpropagation untuk multilayer perceptron yang dapat menyelesaikan masalah yang nonlinearly separable (Han& Kamber,2007).

Secara garis besar metode learning dibedakan menjadi dua, yaitu supervised dan unsupervised. Dalam metode unsupervised learning metode diterapkan tanpa adanya latihan (training) dan tanpa guru. Guru yang dimaksud adalah label dari data. Sedangkan dalam metode supervised., kita melatih dulu teknik yang digunakan dengan data yang ada, setelah dilatih dengan cukup data maka tools kita (misalkan neuro network) bisa digunakan untuk memprediksi keputusan apa yang harus diambil bila data baru masuk dan belum diketahui keluarannya.

Ada beberapa metode supervised learning yang telah ditemukan oleh peneliti, diantaranya yang paling sering diaplikasikan adalah *perceptron* dan *backpropagation*.

3.3 Algoritma Backpropagation

Teknik paling populer pada metode neural network adalah algoritma backpropagation yang banyak digunakan untuk memecahkan banyak masalah didunia nyata dengan membangun model terlatih yang menunjukkan kinerja yang baik dalam beberapa masalah non-linear (Park, Lee &Choi, 2009).

Algoritma pelatihan backpropagation atau ada yang menterjemahkan menjadi propagasi balik pertama kali dirumuskan oleh Paul Werbos pada tahun 1974 dan dipopulerkan oleh Rumelhart bersama McClelland untuk dipakai pada neural network. Metode backpropagation pada awalnya dirancang untuk neural network feedforward, tetapi pada perkembangannya, metode ini diadaptasi untuk pembelajaran pada model neural network lainnya(Astuti, 2009).

Salah satu metode pelatihan terawasi pada neural network adalah metode backpropagation, dimana ciri adalah meminimalkan error pada output yang dihasilkan oleh jaring.

Inisial awal bobot jaring backpropagation yang terdiri atas lapisan input, lapisan tersembunyi dan lisan output (Astuti, 2009). Tahap pelatihan backpropagation merupakan langkah untuk melatih suatu neural network yaitu dengan cara melakukan perubahan penimbang (sambungan antar lapis yang

membentuk neural network melalui masing-masing unitnya.

Dalam algoritma backpropagation terdapat dua parameter yaitu momentum dan learning Rate (Deepa, 2011) yaitu algoritma pelatihan backpropagation terdiri dari dua tahapan yaitu feedforward dan backpropagation dari galatnya. Langkah pembelajaran dalam algoritma backpropagation adalah sebagai berikut (Myatt, 2007) :

- a. Langkah 1
Inisialisasi bobot jaringan secara acak (biasanya antara -0.1 sampai 0.1)
- b. Langkah 2
Untuk setiap data pada data training, hitung input untuk simpul berdasarkan nilai input dan bobot jaringan saat itu, menggunakan rumus :

$$\text{Input } j = \sum_{i=1}^n O_i W_{ij} + \theta_j$$

- c. Langkah 3
Berdasarkan input yang didapat dari langkah dua, selanjutnya membangkitkan output untuk simpul menggunakan fungsi aktifasi sigmoid:

$$\text{Output} = \frac{1}{1 + e^{-\text{input}}}$$

- d. Langkah 4
Hitung nilai error antara nilai yang diprediksi dengan nilai yang sesungguhnya menggunakan rumus :

$$\text{Error}_j = \text{output}_j \cdot (1 - \text{output}_j) \cdot (\text{Target}_j - \text{Output}_j)$$

- e. Langkah 5
Setelah nilai error dihitung, selanjutnya dibalik ke layer sebelumnya (backpropagation) untuk menghitung nilai error pada *hidden layer*, menggunakan rumus :

$$\text{Error}_j = \text{Output}_j (1 - \text{Output}_j) \sum_{k=1}^n \text{Error}_k W_{jk}$$

- f. Langkah 6
Nilai Error yang dihasilkan dari langkah sebelumnya digunakan untuk memperbarui bobot relasi menggunakan rumus :

$$W_{ij} = W_{ij} + l \cdot \text{Error}_j \cdot \text{Output}_i$$

Keterangan Notasi:

Output_j = Output aktual dari simpul j

Error_k = error simpul k

W_{jk} = Bobot relasi dari simpul j ke simpul k pada layer berikutnya

Output_j = Output aktual dari simpul j

Target_j = Nilai target yang sudah diketahui pada data training

O_i = Output simpul i dari layer sebelumnya

w_{ij} = bobot relasi dari simpul i pada layer sebelumnya ke simpul j

θ_j = bias (sebagai pembatas)

W_{ij} = bobot relasi dari unit i pada layer sebelumnya ke unit j

l = learning rate (konstanta, nilainya antara 0 sampai dengan 1)

Error_j = Error pada output layer simpul j

Output_i = Output dari simpul i

Namun algoritma *backpropagation* mempunyai beberapa kelemahan (Park, Lee, & Choi, 2009) :

1. Pertama, algoritma *backpropagation* (2.1) bisa terjebak dalam masalah lokal minimum, hal ini dapat menyebabkan kegagalan untuk mencari solusi yang optimal dalam pemilihan fitur pada bobot atribut yang digunakan.
2. Algoritma *backpropagation* memiliki kecepatan konvergen yang terlalu lambat yang pada akhirnya algoritma *backpropagation* sangat tergantung pada parameter awal seperti jumlah masukan, node tersembunyi, output, learning rate dan bobot koneksi dalam jaringan

Masalah umum lainnya adalah algoritma *backpropagation* mempunyai kelemahan pada perlunya data training yang besar dan optimasi yang digunakan kurang efisien (Xiao & Shao, 2011). Hal ini dapat diselesaikan dengan algoritma Particle Swarm Optimization.

3.4 Particle swarm optimization

PSO memiliki perbandingan lebih untuk pemilihan fitur dan memiliki kinerja lebih unggul untuk banyak masalah optimasi dengan lebih cepat dan tingkat konvergensi yang lebih stabil (Ling, Nguyen, & Chan, 2009). Karakteristik PSO adalah interaksi sosial yang mempromosikan pembagian informasi antara partikel yang akan membantu dalam pencarian solusi yang optimal (Park, Lee, & Choi, 2009).

PSO memiliki beberapa parameter seperti posisi, kecepatan, kecepatan maksimum, percepatan konstanta dan berat inersia. Dalam teknik PSO terdapat beberapa cara untuk

melakukan pengoptimasian diantaranya: meningkatkan bobot atribut (attribute weight) terhadap semua atribut atau variabel yang dipakai, menyeleksi atribut (attribute selection) dan *feature selection*.

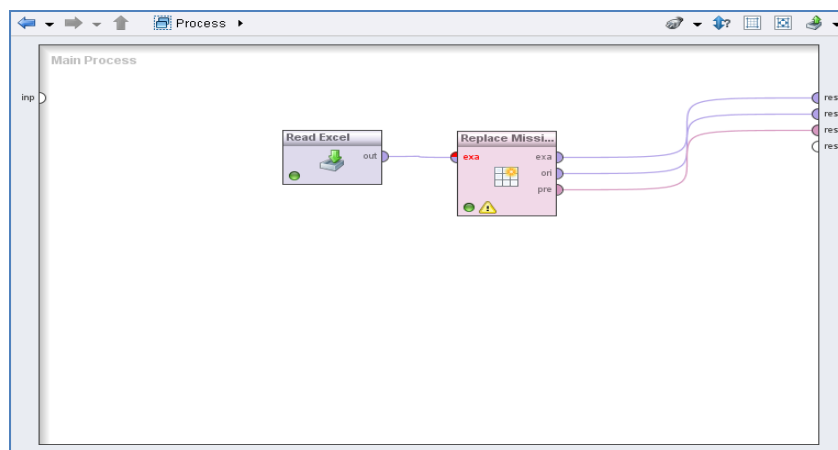
Pada penelitian ini PSO akan diterapkan untuk memecahkan masalah yang terjadi pada *neural network* dengan memilih fitur pada bobot atribut untuk memaksimalkan kinerja dari model yang dihasilkan sehingga hasil prediksi mahasiswa berpotensi DO lebih akurat.

IV. PEMBAHASAN

Pada tahapan ini dilakukan uji coba dan teknik pengujian yang digunakan untuk menyeleksi atribut yang berpengaruh untuk meningkatkan nilai akurasi berdasarkan data set yang digunakan.

Jumlah data awal yang diperoleh dari pengumpulan data yaitu sebanyak 3460 data, namun tidak semua data dapat digunakan dan tidak semua atribut digunakan karena harus melalui beberapa tahap pengolahan awal data (*preparation data*). Untuk mendapatkan data yang berkualitas, beberapa teknik yang dilakukan sebagai berikut (Vercellis, 2009):

Data validation, untuk mengidentifikasi dan menghapus data ganjil (*outlier/ noise*), data yang tidak konsisten dan data yang tidak lengkap (*missing value*)



Gambar 1. Proses *Replace Missing value*

ExampleSet (1319 examples, 1 special attribute, 12 regular attributes)						
Role	Name	Type	Statistics	Range	Missings	
label	asl_sltta	polynomial	mode = SMA (771), least = SW	SMA (771), SMK (91), BINA SA	0	
regular	Nim	integer	avg = 13383841.190 +/- 3882	[11030010.000 ; 23060006.000]	0	
regular	Nama	polynomial	mode = SUPRIYANTO (3), lea	INDRA (3), SUPRIYANTO (3),	0	
regular	Jekel	binominal	mode = L (979), least = P (341)	L (979), P (340)	0	
regular	alamat	polynomial	mode = JL RAYA KEBAYORAN	JL RAYA KEBAYORAN LAMA C	0	
regular	alamat1	polynomial	mode = TANGERANG (225), li	TANGERANG (225), JAKARTA	117	
regular	jurusan	integer	avg = 13.358 +/- 3.931	[2.000 ; 23.000]	27	
regular	no_by_klh	integer	avg = 0.861 +/- 1.888	[0.000 ; 5.000]	28	
regular	no_ta	polynomial	mode = 09.00320.2 (2), least =	08.00177.2 (2), 09.00320.2 (2)	1022	
regular	kondisi	integer	avg = 13.353 +/- 307.729	[2.000 ; 11070.000]	26	
regular	jrs_sltta	polynomial	mode = IPS (116), least = 143	IPS (116), IPA (76), MI (57), AK	945	
regular	waktu	integer	avg = 1.921 +/- 9.894	[1.000 ; 357.000]	26	
regular	cabang	integer	avg = 2.127 +/- 1.555	[1.000 ; 7.000]	0	

Gambar 2. Hasil Pengecekan *Replace Missing Value* dengan Rapid Miner

Role	Name	Type	Statistics	Range	Missings
label	kondisi	binominal	mode = Tdk_Do (783), least= DO (506), Tdk_Do (783)		0
regular	jenis	integer	avg = 1.247 +/- 0.431	[1.000 ; 2.000]	0
regular	jurusan	integer	avg = 13.158 +/- 3.745	[11.000 ; 23.000]	0
regular	no_by_klh	integer	avg = 0.872 +/- 1.897	[0.000 ; 5.000]	0
regular	waktu	integer	avg = 1.671 +/- 0.470	[1.000 ; 2.000]	0
regular	cabang	integer	avg = 2.076 +/- 1.546	[1.000 ; 7.000]	0
regular	kedekatan_jarak	integer	avg = 0.970 +/- 0.171	[0.000 ; 1.000]	0
regular	sekolah_negeri	integer	avg = 0.079 +/- 0.270	[0.000 ; 1.000]	0
regular	transfer	integer	avg = 0.073 +/- 0.261	[0.000 ; 1.000]	0
regular	usia_mendaftar	integer	avg = 1.856 +/- 0.493	[1.000 ; 4.000]	0
regular	total_sks	integer	avg = 96.567 +/- 50.654	[16.000 ; 163.000]	0
regular	total_mutu	integer	avg = 226.387 +/- 161.805	[0.000 ; 575.000]	0
regular	ipk	real	avg = 2.034 +/- 0.983	[0.000 ; 3.770]	0

Gambar 3. Tampilan Example Set Setelah Proses Missing Value

- a) Data *integration and transformation*, untuk meningkatkan akurasi dan efisiensi algoritma. Data yang digunakan dalam penulisan ini bernilai kategorikal. Data ditransformasikan kedalam *software Rapidminer*. Tabel kategorikal atribut terlihat pada tabel 1.
- b) Data *size reduction and discretization*, untuk memperoleh data set dengan jumlah atribut dan *record* yang lebih sedikit tetapi bersifat informatif

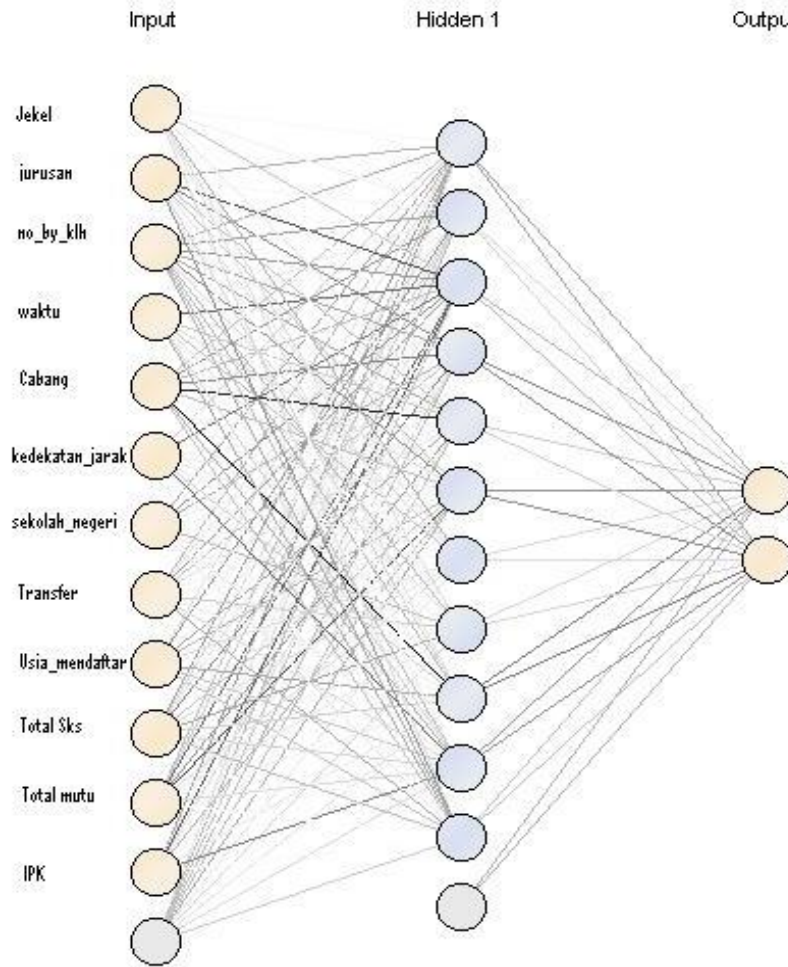
Tabel 1. Atribut data mahasiswa

No	Atribut	Nilai
1	Jenis kelamin	L (laki-laki), P(perempuan)
2	Jurusan	
3	No_by_kuliah	Kondisi administrasi biaya kuliah
4	Waktu	Waktu perkuliahan 1=pagi ; 2=sore
5	Usia saat mendaftar	1=jika usia saat mendaftar <20 ; 2= jika usia saat mendaftar >=20; 3= jika usia saat mendaftar >30
6	Asal Sekolah negeri	0= asal sekolah swasta 1= asal sekolah negeri
7	Transfer	0=mhs murni ,dual kampus 1=transfer D3
8	Cabang	01;02;03;04;05;06 dan 07
9	Total sks	Jumlah satuan kredit semester yang pernah diambil
10	Total Mutu	Bobot nilai sks yang telah diambil
11	Ipk	Indeks Prestasi Kumulatif
12	Kondisi akademik	Nonaktif, cuti, registrasi dan DO

4.1 Pengolahan Data

Pada penelitian ini dilakukan dua kali pengolahan data, yang pertama adalah pengolahan data hanya dengan menggunakan metode Neural Network, dan yang kedua adalah pengolahan data dengan menggunakan

Neural Network berbasis Particle Swarm Optimization Pertama kali ujicoba dilakukan dengan menggunakan metode Neural Network saja dan memiliki nilai akurasi sebesar 79.36% dengan arsitektur NN-nya seperti yang terlihat pada gambar



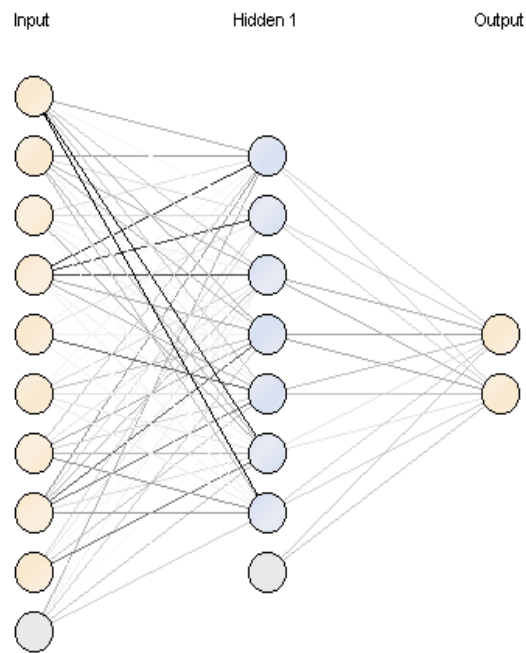
Gambar 4. Arsitektur Neural Network yang terbentuk

Uji coba kedua dilakukan dengan menggunakan metode Neural Network berbasis Particle Swarm Optimization, dimana pada ujicoba kedua ini nilai akurasi yang didapat meningkat sebesar 2.99%,

serta dihasilkan atribut-atribut yang berpengaruh terhadap prediksi mahasiswa Drop Out seperti yang terdapat pada gambar 2. Dan tabel perbandingan hasil dapat dilihat dari tabel 2.

atribut	weight
jekel	0
jurusan	0.739
no_by_klh	0.854
waktu	0.125
cabang	0.299
kedekatan_jarak	1
sekolah_negeri	0.002
transfer	1
usia_mendaftar	0
total_sks	0.081
total_mutu	0
ipk	1

Gambar 5. Bobot nilai atribut yang dihasilkan setelah dibasiskan dengan PSO



Gambar 6. Arsitektur Neural Network berbasis PSO yang terbentuk

Tabel 2. Perbandingan Hasil metode Neural Network dengan Neural Network berbasis PSO

Neural Network (%)		Neural Network + PSO (%)	
Accuracy	79,36	Accuracy	82,35
Sensitivity	73,83	Sensitivity	77,53
Specificity	144,27	Specificity	85,60
PPV	74,70	PPV	78,45
Npv	82,44	Npv	84,92

4.2 Analisa Evaluasi dan Validasi Model

Dari hasil pengujian di atas, baik evaluasi menggunakan *confusion matrix* maupun *ROC curve* terbukti bahwa hasil pengujian algoritma *neural network* berbasis *particle swarm optimization* memiliki nilai akurasi yang lebih

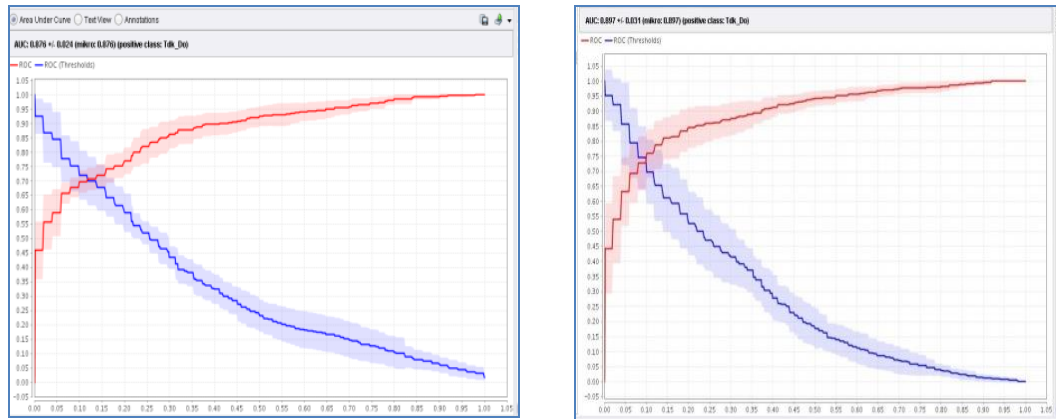
tinggi dibandingkan dengan algoritma *neural network* saja. Nilai akurasi untuk model algoritma *neural network* sebesar 79,36% dan nilai akurasi untuk model *neural network* berbasis PSO meningkat menjadi 82,35% dengan selisih akurasi 2,99%, dapat dilihat pada tabel 3.

Tabel 3. Pengujian algoritma Neural Network berbasis PSO

	NN	NN berbasis PSO
Accuracy	79,36%	82,35%
Auc	0,876	0,897

Untuk evaluasi menggunakan ROC curve sehingga menghasilkan nilai AUC untuk model algoritma Neural network menghasilkan nilai 0,876 dengan diagnosa *fair classification*, sedangkan untuk algoritma

neural network berbasis PSO menghasilkan nilai 0,897 dengan nilai *diagnosa classification*, dan selisih keduanya sebesar 0,021 dapat dilihat pada gambar 4 dibawah ini:



Gambar 7. Curva ROC dengan Neural Network dan Neural network berbasis Particle Swarm Optimization

V. KESIMPULAN

Dari hasil analisis optimasi model algoritma *neural network* berbasis *particle swarm optimization* memberikan nilai akurasi yang lebih tinggi yaitu 82.35% Dibandingkan dengan model algoritma neural network yaitu sebesar 79.36%. Dari hasil tersebut didapatkan selisih antara kedua model yaitu 2.99%. Sementara untuk evaluasi menggunakan ROC curve untuk kedua model yaitu, untuk model algoritma *neural network* memiliki nilai AUC adalah 0.876 dengan tingkat diagnosa *Good Classification* dan untuk model algoritma *neural network* berbasis *particle swarm optimization* nilai AUC adalah 0.897 dengan tingkat diagnosa *Good Classification*

DAFTAR PUSTAKA

- Astuti, E. D. (2009). *Pengantar Jaringan Saraf Tiruan*. Wonosobo: Star Publishing.
- Berndtsson, M., Hansson, J., Olsson, B., & Lundell, B. (2008). *A Guide for Students in Computer Science and Information Systems*. London: Springer.
- Dawson, C. W. (2009). *Projects in Computing and Information System A Student's Guide*. England: Addison-Wesley.
- Gorunescu, F. (2011). *Data Mining Concept Model Technique*. India: Springer.
- Gray, D. E. (2004). *Doing Research in the Real World*. New Delhi: SAGE.
- Gerben W. Dekker, "Predicting Students Drop Out: A Case Study," *In International Conference on Educational Data Mining, Cordoba, Spain*, 41-50, 2009
- Han, J., & Kamber, M. (2007). *Data Mining Concepts and Technique*. Morgan Kaufmann publisher.
- Hastutik, khafiizh. (2012). Analisis Komparasi Algoritma Klasifikasi Data Mining Untuk Prediksi Mahasiswa Non Aktif. *Semantik 2012*. ISBN 979 - 26 - 0255 - 0
- K, G. S., & Deepa, D. S. (2011). Analysis of Computing Algorithm using Momentum in Neural

- Networks. *Journal of computing, volume 3, issue 6* , 163-166.
- Kusrini, & Luthfi, E. T. (2009). *Algoritma Data mining*. Yogyakarta: Andi.
- Larose, D. T. (2005). *Discovering Knowledge in Data*. Canada: Wiley Interscience.
- Ling, S. H., Nguyen, H. T., & Chan, K. Y. (2009). A New Particle Swarm Optimization Algorithm for Neural Network Optimization. *Network and System Security, third International Conference* , 516-521.
- Myatt, G. J. (2007). *Making Sense of Data A Practical Guide to Exploratory Data Analysis and Data Mining*. New Jersey: A John Wiley & Sons, inc., publication.
- Nagadevara, & Vishnuprasad. (2005). Building Predictive models for election result in india an application of classification trees and neural network. *Journal of Academy of Business and Economics Volume 5* .
- Park, T. S., Lee, J. H., & Choi, B. (2009). Optimization for Artificial Neural Network with Adaptive inertial weight of particle swarm optimization. *Cognitive Informatics, IEEE International Conference* , 481-485.
- Purnomo, M. H., & Kurniawan, A. (2006). *Supervised Neural Network*. Suarabaya: Garaha Ilmu.
- Salappa, A., Doumpos, M., & Zopounidis, C. (2007). Feature Selection Algorithms in Classification Problems: An Experimental Evaluation. *Systems*
- Shukla, A., Tiwari, R., & Kala, R. (2010). *Real Life Application of Soft Computing*. CRC Press.
- Sug, H. (2009). An Empirical Determination of Samples for Decision Trees. *AIKED'09 Proceeding of the 8th WSEAS international conference on Artificial intelligence, Knowledge engineering and data bases* , 413-416.
- Sotiris Kotsiantis, "Educational Data Mining: A Case Study for Predicting Dropout-Prone Students," *Int. J. of Knowledge Engineering and Soft Data Paradigms*, vol. X, 2010
- Vercellis, C. (2009). *Business Intelligence : Data Mining and Optimization for Decision Making*. John Wiley & Sons, Ltd.
- Xiao, & Shao, Q. (2011). Based on two Swarm Optimized algorithm of neural network to prediction the switch's traffic of coal. *ISCCS '11 Proceeding of the 2011 International Symposium on Computer Science and Society* , 299-302