

Implementasi Algoritma Genetika dan Neural Network Pada Aplikasi Peramalan Produksi Mie (Studi Kasus : Omega Mie Jaya)

Adhi Kusnadi¹, Jansen Pratama²

Fakultas Teknik dan Informatika, Universitas Multimedia Nusantara, Gading Serpong, Indonesia
adhi.kusnadi@umn.ac.id¹, jansen_pratama@yahoo.co.id²

Diterima 15 Mei 2017

Disetujui 16 Juni 2017

Abstract— Companies that produce products must be able to regulate the amount of production so that it have plan production. Therefore, it is necessary to be able to predict the amount of production. This research aims to create an application that is useful in determining the amount of production. These applications using genetic algorithms and neural network. Genetic algorithm is used to optimize the weights in the neural network. From the test results, this application uses network with 12 inputs, 5 neuron in first hidden layer, 3 neurons in the second hidden layer, and 3 neurons in the last hidden layer. Then for the genetic algorithm parameters used were 10 individuals, 50 generations, crossover probability 0.8 and mutations probability 0.1. Based on the test results, this application has the forecasting's accuracy rate reaches 86%.

Keyword : forecasting, production forecasting, genetic algorithm neural network, optimization.

I. PENDAHULUAN

Kegiatan ekonomi sangat penting bagi kehidupan manusia untuk memenuhi kebutuhannya, terdiri dari produksi, distribusi dan konsumsi. Produksi mencakup setiap usaha manusia yang menghasilkan barang/jasa yang langsung atau tidak berguna untuk memenuhi kebutuhan manusia [1]. Tentunya produksi dilakukan atas dasar permintaan dari konsumen. Perusahaan yang menghasilkan produk harus bisa mengatur jumlah produksi agar produksi menjadi terencana dengan baik. Oleh karena itu diperlukan suatu kemampuan untuk dapat meramalkan jumlah produksi [2]. Dengan demikian, maka produksi akan menjadi lebih efisien dan efektif serta dapat menghemat biaya. Untuk itu, dibutuhkan suatu sistem yang dapat membantu perusahaan dalam menentukan jumlah produksi. Menurut Gaspersz, peramalan adalah dugaan yang dibuat secara sederhana tentang apa yang akan terjadi di masa depan berdasarkan informasi yang tersedia saat ini. Jadi perusahaan dapat melakukan peramalan menggunakan data produksi yang ada. Kemudian dapat ditemukan pola-pola produksi dari data tersebut. Dengan data pola pola dapat dilakukan peramalan untuk jumlah produksi.

Ada beberapa algoritma yang dapat digunakan untuk menentukan jumlah produksi, salah satunya dengan Artificial Neural Network (ANN) dengan metode backpropagation. Metode Backpropagation Neural Network telah digunakan untuk oleh Rini Oktavian Maru'ao dalam jurnalnya dengan judul Neural Network Implementation in Foreign Exchange Kurs Prediction, untuk membuat suatu aplikasi untuk memprediksi kurs valuta asing [3]. Selain itu ANN juga dapat dikombinasikan dengan beberapa metode lainnya untuk meningkatkan keakuratan hasil peramalan seperti Particle Swarm Optimization (PSO). Particle Swarm Optimization Neural Network dapat digunakan untuk memprediksi laju inflasi [4]. Selain PSO ada lagi algoritma yang dapat dikombinasikan, yaitu algoritma genetika. Algoritma ini dipakai untuk mendapatkan solusi optimal yang tepat untuk masalah dari satu variabel atau multi variabel. Terdapat penelitian sebelumnya mengenai Algoritma Genetika oleh Devi Rahmayanti untuk menentukan nilai optimal dalam routing [5] dan Algoritma Genetika juga bisa diterapkan dalam ANN[6].

Perusahaan Omega Mie Jaya adalah salah satu produsen mie yang berada di Pemangkat, Kalimantan Barat. Omega Mie Jaya belum memiliki suatu sistem yang dapat membantu untuk menetapkan jumlah produksi mie. Hal ini dikarenakan kurangnya sumber daya manusia yang mengerti tentang sistem untuk menentukan jumlah produksi. Maka perlu dibuat suatu system yang dapat meramalkan jumlah produksi mi. Dengan dibuatnya sistem ini, maka dapat perusahaan memiliki proses produksi yang semakin baik dan terencana. Dari hasil uraian tersebut, maka dibuatlah suatu aplikasi peramalan produksi mie yang akan diimplementasikan ke Perusahaan Omega Mie Jaya dengan menggunakan Algoritma Genetika Neural Network.

II. TINJAUAN PUSTAKA

A. Peramalan

Metode peramalan digunakan untuk mengukur keadaan di masa datang. Peramalan dapat dilakukan secara kualitatif dan kuantitatif. Pengukuran kuantitatif menggunakan metode statistik, sedangkan pengukuran secara kualitatif berdasarkan pendapat (*judgement*) dari yang melakukan peramalan [7]. Peramalan biasanya diklasifikasikan berdasarkan horizon waktu di masa depan yang terbagi atas beberapa kategori [8]:

1. Peramalan jangka pendek, peramalan ini mencakup jangka waktu hingga satu tahun, tetapi umumnya kurang dari tiga bulan. Peramalan ini digunakan untuk merencanakan pembelian, penjadwalan kerja, jumlah tenaga kerja, penugasan kerja, dan tingkat produksi.
2. Peramalan jangka menengah, peramalan ini umumnya mencakup hitungan bulanan hingga tiga tahun. Peramalan ini digunakan untuk merencanakan penjualan, perencanaan dan anggaran produksi, anggaran kas, dan menganalisis bermacam-macam rencana operasi.
3. Peramalan jangka panjang, peramalan ini umumnya untuk perencanaan masa tiga tahun atau lebih. Peramalan ini digunakan untuk merencanakan produk baru, pembelanjaan modal, lokasi atau pengembangan fasilitas, serta penelitian dan pengembangan.

B. Produksi

Produksi adalah usaha manusia untuk mengubah serta mengolah sumber daya ekonomi menjadi bentuk serta kegunaan baru. Dengan kata lain, kegiatan produksi adalah proses mengolah produk yang dapat berupa barang dan jasa [9]. Kegiatan produksi adalah kegiatan menciptakan serta menambah nilai guna suatu barang atau jasa dengan menggunakan faktor – faktor produksi [10]. Faktor – faktor produksi antara lain sumber daya alam, sumber daya manusia, sumber daya modal, sumber daya pengusaha.

C. Neural Network

Neural Network merupakan salah satu representasi dari otak manusia yang mencoba mensimulasikan proses pembelajaran pada otak manusia [11]. *Neural Network* ini pertama kali dipresentasikan oleh Warren McCulloch dan Walter Pitt pada tahun 1943 [12]. *Neural Network* memiliki beberapa fitur atraktif antara lain:

- Ketahanan dan toleransi kesalahan. Rusaknya salah satu sel tidak mempengaruhi *performance* secara signifikan.
- Fleksibel. Jaringan dapat menyesuaikan diri dengan lingkungan baru tanpa perlu menggunakan program terinstruksi. - Memiliki kemampuan komputasi yang kolektif.

- Kemampuan dalam berhubungan dengan variasi data.

D. Backpropagation

Dalam *neural network* ada dua proses pembelajaran yaitu terawasi dan tidak terawasi. Pelatihan yang terawasi dilakukan dengan cara memberi *neural network* data beserta dengan output yang telah diantisipasi oleh data tersebut. *Neural network* akan melakukan proses iterasi hingga output sesuai dengan diharapkan dengan rate-error yang kecil. Ada beberapa algoritma dalam pembelajaran terawasi, salah satunya adalah backpropagation [13]. Algoritma *Backpropagation* menggunakan *error output* untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*) [11]. Untuk mendapatkan error ini, tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu. Pada saat perambatan maju, neuron-neuron diaktifkan dengan menggunakan fungsi aktivasi sigmoid.

Pelatihan *backpropagation* meliputi tiga fase. Fase pertama adalah fase maju. Pola masukan dihitung maju mulai dari lapisan masukan hingga lapisan keluaran menggunakan fungsi aktivasi yang ditentukan. Fase kedua adalah fase mundur. Selisih antara keluaran jaringan dengan target yang diinginkan merupakan kesalahan yang terjadi.

Kesalahan tersebut dipropagasikan mundur, dimulai dari garis yang berhubungan langsung dengan unit-unit di lapisan keluaran. Fase ketiga adalah fase modifikasi bobot untuk menurunkan kesalahan yang terjadi. Ketiga fase tersebut diulang-ulang terus hingga kondisi penghentian dipenuhi. Umumnya kondisi penghentian yang sering dipakai adalah jumlah iterasi atau kesalahan [14].

E. Algoritma Genetika

Pada tahun 1975, Holland mengembangkan gagasan ini dalam bukunya "*Adaption in natural and artificial systems*". Ia menggambarkan bagaimana menerapkan prinsip evolusi alami untuk masalah optimasi dan dibangun algoritma genetika pertama. Teori Holland telah dikembangkan lebih lanjut dan sekarang algoritma genetika berdiri sebagai alat yang ampuh untuk memecahkan masalah pencarian dan optimasi. Algoritma genetika didasarkan pada prinsip genetik dan evolusi [15]. Algoritma genetika adalah merupakan metode adaptif yang biasa digunakan untuk pencarian nilai dalam suatu masalah optimasi. Algoritma ini adalah salah satu evolusi yang paling populer algoritma di mana populasi individu berkembang sesuai dengan set aturan seperti seleksi, crossover dan mutasi [16].

F. Algoritma Genetika dan Neural Network

Algoritma genetika dapat digunakan untuk meningkatkan kinerja ANN dengan banyak cara yang

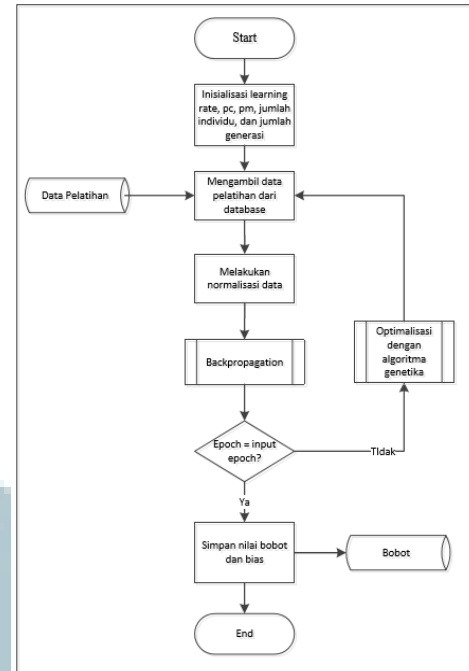
berbeda. Algoritma genetika adalah metode pencarian umum stochastic, yang dapat digunakan dengan *backpropagation neural network* untuk menentukan jumlah node tersembunyi dan lapisan tersembunyi, pilih subset fitur yang relevan, tingkat pembelajaran, momentum, dan menginisialisasi serta mengoptimalkan bobot koneksi jaringan dari *backpropagation neural network*. Algoritma genetika telah digunakan untuk secara optimal merancang parameter neural network termasuk ANN arsitektur, bobot, pilihan input, fungsi aktivasi, jenis neural network, algoritma pelatihan, jumlah iterasi, dan rasio partisi data [17] [18].

Proses algoritma GA - NN untuk proses peramalan ini adalah sebagai berikut:

1. Inisialisasi count = 0 , fitness = 0 , jumlah siklus
2. Generasi populasi awal . Kromosom individu dirumuskan sebagai urutan gen berturut-turut, masing-masing pengkodean input.
3. Desain jaringan yang cocok
4. Menetapkan bobot
5. Melakukan training dengan backpropagation
6. Cari kesalahan kumulatif dan nilai fitness. Kemudian dievaluasi berdasarkan nilai fitness.
7. Jika fitness sebelumnya < nilai fitness saat ini, simpan nilai saat ini
8. Count = count +1
9. Seleksi : Dua induk dipilih dengan menggunakan mekanisme wheel roulette
10. Operasi Genetik : crossover, mutasi dan reproduksi untuk menghasilkan fitur baru set
11. Jika (jumlah siklus <= count) kembali ke nomor empat
12. Pelatihan jaringan dengan fitur yang dipilih
13. Studi kinerja dengan data uji.

III. PERANCANGAN DAN IMPLEMENTASI

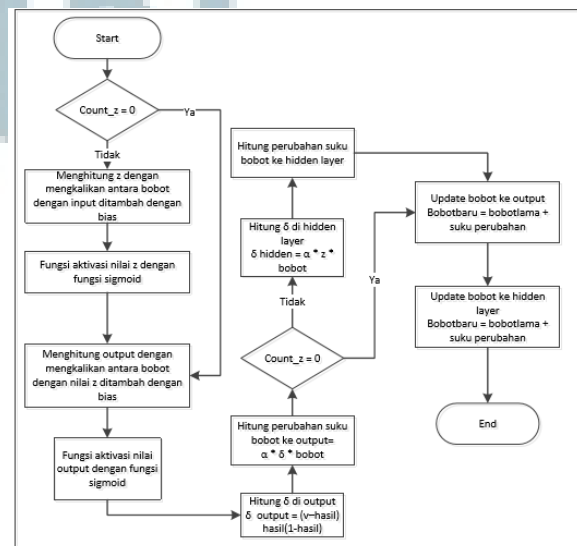
Berikut ini akan digambarkan flowchart sistem peramalan produksi :



Gambar 1. Flowchart Pelatihan

Sistem terdiri dari dua proses utama yaitu pelatihan dan peramalan. Untuk dapat melakukan suatu peramalan menggunakan *neural network*, diperlukan suatu proses pelatihan. Proses pelatihan digabung dengan algoritma genetika untuk mendapatkan bobot yang lebih optimal.

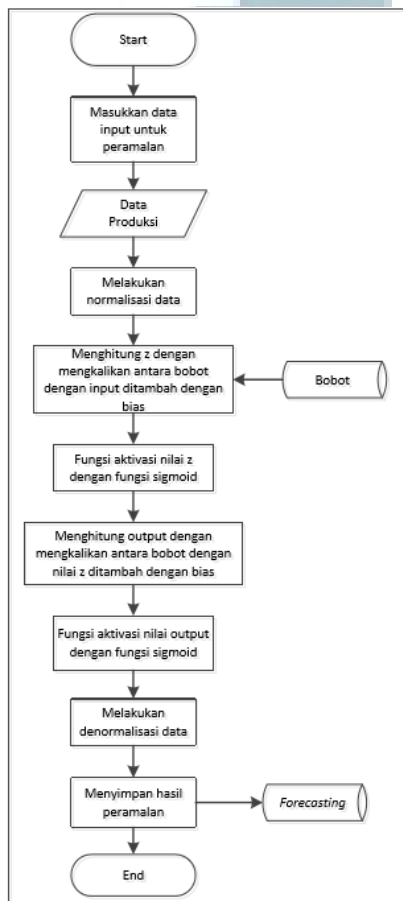
Langkah pertama tentunya kita menginisialisasi semua nilai yang akan digunakan selama proses pelatihan. Kemudian kita mengambil data-data yang digunakan untuk proses pelatihan yang diambil dari database. Setelah itu menormalisasi data tersebut untuk menyesuaikan antara range input-output dengan fungsi aktivasinya. Langkah selanjutnya adalah masuk ke *backpropagation* sebagai berikut :



Gambar 2. Proses Backpropagation

Pada subproses ini, dilakukan penghitungan nilai neuron di *hidden layer* dan ditambah dengan biasanya. Kemudian hasil yang didapat diberi fungsi aktivasi sigmoid. Hal ini juga dilakukan di *layer output*. Proses tersebut merupakan fase pertama dari *backpropagation*. Fase kedua adalah mencari δ di *layer output* dan menghitung suku perubahan ke *layer output*. Setelah itu, juga mencari δ di *hidden layer* dan menghitung suku perubahan ke *hidden layer*. Lalu meng-update ke *hidden layer* dan ke *layer output*.

Proses peramalan dimulai ketika user memasukkan data input produksi perbulan selama 12 bulan. Lalu sistem akan melakukan normalisasi data tersebut. Kemudian akan dilakukan *feedforward* dengan menggunakan bobot terbaik yang didapat dari proses pelatihan sebelumnya. Setelah nilai output berhasil dihitung, maka akan dinormalisasikan lagi sehingga akan menampilkan dan menyimpan data peramalan tersebut. Berikut ini adalah gambarnya dalam bentuk flowchart.



Gambar 3. Proses Peramalan Dengan *Backpropagation*

IV. UJI COBA

Uji coba dilakukan untuk menentukan parameter-parameter yang tepat untuk digunakan dalam aplikasi ini. Caranya adalah dengan melakukan uji coba terhadap parameter-parameter tersebut dengan menggunakan data – data produksi Omega Mie Jaya. Parameter-parameter yang tepat adalah parameter-parameter yang saat dilakukan pelatihan dan testing memiliki nilai error yang terkecil dengan data validasi.

Tabel 1. *Learning Data*

Bulan1	Bulan2	Bulan3	Bulan4	Bulan5	Bulan6	Bulan7	Bulan8	Bulan9	Bulan10	Bulan11	Bulan12	Target
2816	2992	3366	3960	4048	4576	5324	5258	4774	4554	3690	3168	2508
2992	3366	3960	4048	4576	5324	5258	4774	4554	3690	3168	2508	3916
3366	3960	4048	4576	5324	5258	4774	4554	3690	3168	2508	3916	4026
3960	4048	4576	5324	5258	4774	4554	3690	3168	2508	3916	4026	4334
4048	4576	5324	5258	4774	4554	3690	3168	2508	3916	4026	4334	5478
4576	5324	5258	4774	4554	3690	3168	2508	3916	4026	4334	5478	5852
5324	5258	4774	4554	3690	3168	2508	3916	4026	4334	5478	5852	6600
5258	4774	4554	3690	3168	2508	3916	4026	4334	5478	5852	6600	5874
4774	4554	3690	3168	2508	3916	4026	4334	5478	5852	6600	5874	5082
4554	3690	3168	2508	3916	4026	4334	5478	5852	6600	5874	5082	4598
3690	3168	2508	3916	4026	4334	5478	5852	6600	5874	5082	4598	3872
3168	2508	3916	4026	4334	5478	5852	6600	5874	5082	4598	3872	3520
2508	3916	4026	4334	5478	5852	6600	5874	5082	4598	3872	3520	2684
3916	4026	4334	5478	5852	6600	5874	5082	4598	3872	3520	2684	4094
4026	4334	5478	5852	6600	5874	5082	4598	3872	3520	2684	4094	4378
4334	5478	5852	6600	5874	5082	4598	3872	3520	2684	4094	4378	4752
5478	5852	6600	5874	5082	4598	3872	3520	2684	4094	4378	4752	5682
5852	6600	5874	5082	4598	3872	3520	2684	4094	4378	4752	5682	5984
6600	5874	5082	4598	3872	3520	2684	4094	4378	4752	5682	5984	6798
5654	5082	4598	3872	3520	2684	4094	4378	4752	5682	5984	6798	6116

Berdasarkan data – data yang ada, terdapat 36 data yang membuat 24 pola data. 24 pola data dibagi menjadi dua bagian yaitu 20 pola data (tabel 1) untuk data pelatihan dan 4 data digunakan untuk testing (tabel 2).

Tabel 2. *Testing Data*

Bulan1	Bulan2	Bulan3	Bulan4	Bulan5	Bulan6	Bulan7	Bulan8	Bulan9	Bulan10	Bulan11	Bulan12	Target
5082	4598	3872	3520	2684	4094	4378	4752	5684	5984	6798	6116	5214
4598	3872	3520	2684	4094	4378	4752	5684	5984	6798	6116	5214	4686
3872	3520	2684	4094	4378	4752	5684	5984	6798	6116	5214	4686	4202
3520	2684	4094	4378	4752	5684	5984	6798	6116	5214	4686	4202	3612

Uji coba dimulai dari menentukan arsitektur neural network. Jaringan memiliki tiga bagian, yaitu input, *hidden layer*, dan juga output. Jadi kita menentukan jumlah neuron terbaik yang nantinya akan digunakan untuk aplikasi ini. Tahap kedua adalah menetapkan parameter-parameter genetika. Parameter algoritma genetika akan diuji coba pada tahapan berikutnya sehingga pada tahap ini nilainya belum diubah-ubah. Ketika sudah mendapatkan arsitektur jaringan terbaik, maka dilakukan proses uji coba untuk parameter algoritma genetika. Proses uji coba tahap dilakukan dengan menentukan jumlah neuron kemudian melakukan proses pelatihan dengan cara inialisasi bobot dengan angka random.

Kemudian dihitung nilai neuron dan juga bobotnya dengan menggunakan data pelatihan yang telah dinormalisasi terlebih dahulu. Proses tersebut merupakan proses yang dilakukan pada tahap fungsi evaluasi dalam algoritma genetika. Kemudian masuk ke tahap seleksi, crossover, dan mutasi. Setelah proses pelatihan, maka dilakukan testing feedforward dan

dihitung nilai error antara data yang dihasilkan program dengan data validasi. Variabel awal algoritma genetika yang digunakan adalah enam individu, sepuluh generasi, 0,6 peluang crossover, 0,1 peluang mutasi.

Tabel 1. Percobaan Mencari Parameter

Learning Rate	Arsitektur	Error Peramalan
0,05	12-3-1	24,1 %
0,1	12-3-1	26,9 %
0,2	12-3-1	22,5 %
0,05	12-5-1	19,3 %
0,1	12-5-1	21,4 %
0,2	12-5-1	17,5 %
0,05	12-7-1	20,5 %
0,1	12-7-1	28,7 %
0,2	12-7-1	29,3 %
0,1	12-5-2-1	18,6 %
0,2	12-5-2-1	18,4 %
0,05	12-5-2-1	19,5 %
0,2	12-5-3-1	19,2 %
0,1	12-5-3-2-1	18,1 %
0,2	12-5-3-2-1	17,8 %
0,2	12-5-3-3-1	17,3 %
0,1	12-5-3-3-1	19,4 %

Berdasarkan tabel diatas dapat disimpulkan bahwa arsitektur yang memiliki error yang terkecil adalah arsitektur dengan konfigurasi jaringan 12 – 5 – 3 – 3 – 1 yang memiliki tingkat akurasi mencapai 17,3 %.

Setelah melakukan tahap pertama, masuk ke tahap berikutnya yaitu ujicoba variabel algoritma genetika untuk mendapatkan nilai error yang terkecil. Pada tahap ini kita berusaha mendapatkan jumlah individu, jumlah Generasi peluang crossover, dan peluang mutasi yang terbaik dengan cara menghitung error dari bobot yang dihasilkan dalam proses genetika dengan data validasi. Dari hasil uji coba berikutnya, parameter untuk algoritma genetika yang terbaik adalah 10 individu, 50 generasi, 0,8 peluang crossover, dan 0,1 peluang mutasi dapat mencapai error sampai dengan 14%. Hal ini menunjukkan bahwa perubahan parameter algoritma genetika juga mempengaruhi penurunan tingkat error dari hasil peramalan produksi.

V. SIMPULAN

Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa aplikasi peramalan produksi mie untuk Omega Mie Jaya dengan menggunakan algoritma genetika dan *neural network* berhasil dilakukan. Aplikasi peramalan ini memiliki arsitektur jaringan berupa 12 input, tiga jaringan *hidden layer* dimana layer pertama terdiri dari 5 neuron, layer kedua memiliki tiga neuron dan lapisan

terakhir mempunyai tiga neuron, dan satu buah output dengan laju pembelajaran yang digunakan adalah 0,2. Parameter yang digunakan algoritma genetika adalah sepuluh individu, 0,8 peluang crossover, 0,1 peluang mutasi, dan 50 generasi. Dari uji coba yang dilakukan, aplikasi peramalan ini memiliki tingkat akurasi peramalan mencapai 86%.

DAFTAR PUSTAKA

- [1] Gilarso, T. 2004. "Pengantar Ilmu Ekonomi Makro". Yogyakarta: Kanisius.
- [2] Gasperz, Vincent. 2002. "Total Quality Management". Jakarta: PT Gramedia Pustaka Utama .
- [3] Maru'ao, Dini Oktaviani. 2010. "Neural Network Implementation in Foreign Exchange Kurs Prediction". Universitas Gunadharma.
- [4] Raharjo, Joko S. Dwi. 2013. "Model Artificial Neural Network Berbasis Particle Swarm Optimization untuk Prediksi Laju Inflasi". Dalam Jurnal Sistem Komputer Volume 3 hlm. 11-21.
- [5] Rahmayanti, Devi. 2010. "Optimasi Routing Berbasis Algoritma Genetika pada Sistem Komunikasi Bergerak". Dalam Jurnal EEEICIS Volume IV hlm. 18-23.
- [6] Dhanwani, Deepak dan Avinash Wadhe. 2013. "Study of Hybrid Genetic Algorithm Using Artificial Neural Network In Data Mining For The Diagnoses of Stroke Disease" Dalam International Journal of Computational Engineering Research, hlm 95-100.
- [7] Herjanto, Eddy. 2008. "Manajemen Operasi (Edisi Ketiga)". Jakarta: Grasindo.
- [8] Prasetya, Heri dan Fitri Lukiastuti. 2009. "Manajemen Operasi". Yogyakarta: MedPress.
- [9] Ahman, Eeng dan Epi Indriani. 2007. "Membina Kompetensi Ekonomi". Bandung: Grafindo Media Pratama.
- [10] Nafari, M. 2007. "Penganggaran Peusahaan". Jakarta: Salemba Empat.
- [11] Kusumadewi, S. 2004. "Membangun Jaringan Syaraf Tiruan Menggunakan Matlab dan Excel Link". Yogyakarta: Graha Ilmu.
- [12] Rojas, Raul. 1996. "Neural Networks". New York: Springer-Verlaag Berlin Heidelberg.
- [13] Heaton, Jeff. 2008. "Introduction to Neural Network with Java (Second Edition)". Heaton Research.
- [14] Siang, J.J. 2005. "Jaringan Syaraf Tiruan dan Pemogramannya Dengan Matlab". Yogyakarta: Penerbit Andi.
- [15] Sivanandam, S.N dan S.N. Deepa. 2008. "Introduction to Genetic Algorthm". New York: Springer-Verlaag Berlin Heidelberg.
- [16] Suyanto. 2005. "Algoritma Genetika Dalam Matlab". Yogyakarta: Andi Offset.
- [17] Lanaraga, P. dkk. 1997. "Learning Bayesian networks by genetic algorithms: a case study in the rediction of survival in malignant skin melanoma".
- [18] Dhanwani, Deepak dan Avinash Wadhe. 2013. "Study of Hybrid Genetic Algorithm Using Artificial Neural Network In Data Mining For The Diagnoses of Stroke Disease" Dalam International Journal of Computational Engineering Research, hlm 95-100.