

MODEL PREDIKSI RENTET WAKTU NEURAL NETWORK BERBASIS PARTICLE SWARM OPTIMIZATION UNTUK PREDIKSI HARGA SAHAM

Andri Pramuntadi
Universitas Alma Ata Yogyakarta
e-mail : andri.pramuntadi@gmail.com

Abstract

This research started from the concerns of stock where is difficult to predict stock. Stock is an indicator of market trends, profitability, employment benchmark portfolio, as well as the determination of passive strategies and derivative products. Information on the sale of long stock data is very helpful in prediction, the greater the data from the past the greater accuracy obtained. However, the size of the data also affects the performance of the algorithms used. Applied in this study how the Neural Network (NN) with Particle Swarm Optimization (PSO) to perform feature selection dataset used. Experiments were performed in the first experiment conducted training dataset with NN, then on the second try with the training dataset will be made PSO-NN for feature selection. Results of PSO-NN weighting attributes in the form of a dataset, the attribute with the highest weight is the most influential attributes in training. New dataset with feature selection then do training again. Using the NN, results from experiments conducted by NN training cycle 500, 3 hidden layer, momentum of 0 and a learning rate of 0,2 getting 0,466 rmse. While the results of the experiment NN with feature selection PSO or PSO-NN 0.373 rmse get results. The research is based on PSO-NN is able to predict more accurately.

Keywords: Stock, Neural Network, Particle Swarm Optimization, Prediction, Feature Selection.

Abstrak

Penelitian ini dimulai dari sulitnya para pialang saham untuk memprediksi stok harga saham. Stok merupakan indikator tren pasar, profitabilitas, benchmark benchmark kerja, serta penentuan strategi pasif dan produk derivatif. Informasi penjualan data saham lama sangat membantu dalam prediksi, semakin besar data dari masa lalu semakin besar ketepatan yang didapat. Namun, ukuran data juga mempengaruhi kinerja algoritma yang digunakan. Yang dilakukan dalam penelitian ini bagaimana Neural Network (NN) dengan Particle Swarm Optimization (PSO) yang digunakan untuk melakukan pemilihan fitur dataset saham yang digunakan. Eksperimen yang dilakukan pada percobaan pertama dilakukan dataset pelatihan dengan NN, dan pada percobaan kedua dengan dataset pelatihan akan dibuat PSO-NN untuk pemilihan fitur. Hasil atribut bobot PSO-NN dalam bentuk dataset, atribut dengan bobot tertinggi adalah atribut yang paling berpengaruh dalam latihan. Dataset baru dengan seleksi fitur kemudian melakukan latihan lagi. Dengan menggunakan NN, hasil percobaan yang dilakukan training cycle Neural Network 500, 3 Hidden layer, Momentum 0 dan Learning rate 0,2 mendapatkan 0,466 rmse. Sedangkan hasil percobaan NN dengan seleksi fitur PSO atau PSO-NN 0,373 rmse mendapatkan hasil. Penelitian yang berbasis PSO-NN ini mampu memprediksi secara lebih akurat.

Kata kunci: Saham, Jaringan Syaraf Tiruan, Optimisme Partikel Swarm, Prediksi, Seleksi Fitur.

1. PENDAHULUAN

Saham merupakan indikator trend pasar, tingkat keuntungan, tolak ukur kerja portofolio, serta penentuan strategi pasif dan produk derivatif. Saham diterbitkan oleh perusahaan atau pendirinya yang membutuhkan dana jangka panjang untuk kepentingan dalam bisnis dengan imbalan uang tunai. Ini merupakan metode utama untuk meningkatkan modal bisnis dari perusahaan selain dengan cara obligasi (Darmadji & Hendy, 2005).

Data harus dapat didekomposisikan seperti trend, siklus, musiman dan ketidakteraturan. prediksi menghasilkan suatu model sebagai acuan prediksi harga saham. Salah satu teknik data mining yang sering digunakan dalam peramalan saham adalah Neural network dan Support Vector Machine. Support Vector Machine mempunyai kelebihan dalam over-fitting, lambatnya konvergensi, dan sedikitnya data training namun memiliki kelemahan dalam melakukan training dengan jumlah data yang besar dan lemah pada sulitnya pemilihan parameter SVM yang optimal. Sedangkan neural network memiliki kelebihan yang tidak dapat dilakukan oleh SVM yaitu training data yang besar dan penggunaan parameter serta neural network memiliki kelebihan pada prediksi nonlinear, kuat di parallel processing dan kemampuan untuk mentoleransi kesalahan (Yakup , 2011) (Ravisankar P, 2012) (Sheng, 2009).

Dalam teknik Particle swarm optimization terdapat beberapa cara untuk melakukan pengoptimasian diantaranya: meningkatkan bobot atribut (attribute weight) terhadap semua atribut atau variabel yang dipakai, menseleksi atribut (attribute selection), dan feature selection. Particle swarm optimization dapat digunakan untuk melakukan training pada single neuron pada neural network untuk mengoptimalkan model estimasi parameter (Chen, 2009).

Particle swarm optimization akan diterapkan untuk memecahkan masalah dimensi data yang besar dengan memilih fitur untuk memaksimalkan kinerja dari model yang dihasilkan sehingga hasil prediksi menjadi lebih akurat. Tujuan dari penelitian ini adalah menerapkan Particle Swarm Optimization untuk pemilihan fitur pada dataset prediksi harga saham untuk meningkatkan akurasi pada pembelajaran Neural network. Dan memberikan kontribusi keilmuan pada penelitian algoritma neural network berbasis particle swarm optimization sebagai pemilihan fitur pada harga saham dan diharapkan dapat menjadi pertimbangan dalam melakukan prediksi harga saham.

2. METODE PENELITIAN

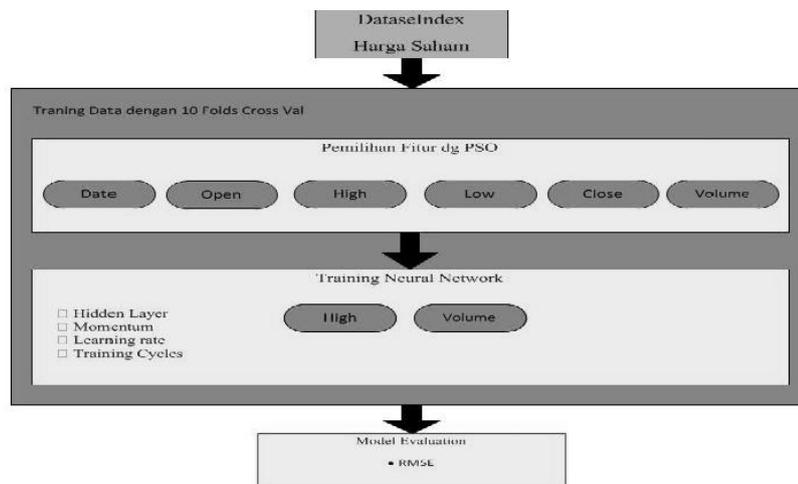
Dan data diambil dari salah satu perusahaan yang aktif di pasar saham Indonesia. Data yang diperoleh merupakan data transaksi penjualan saham yang tercatat setiap hari mulai tanggal 3 Januari 2000 sampai dengan tanggal 13 Juli 2011. Yang terdiri dari data pelatihan dan data pengujian, jumlah data pelatihan sebanyak 2762 yang merupakan data transaksi mulai tanggal 3 Januari 2000 hingga 31 Desember 2010, dan data pengujian sebanyak 134 data transaksi mulai tanggal 3 Januari 2011 sampai dengan 13 Juli 2011. Data yang diperoleh merupakan data time-series dengan jumlah data 2896 dengan 5 kolom.

Pengolahan data awal data masukan dan target yang diimplementasikan ke dalam *neural network* harus terlebih dahulu melalui *preprocessing* data berupa penskalaan yaitu normalisasi data. Tujuannya agar *neural network* dapat mengenali data yang akan menjadi masukan bobot-bobotnya. Pada proses pelatihan *neural network* data yang digunakan bernilai kategorikal, data ditransformasikan ke dalam bentuk bilangan real menggunakan *tools* RapidMiner.

- a. Validasi Data dalam proses validasi ini atribut yang kurang lengkap, tidak konsisten, kurang rapi ditangani secara sistematis melalui algoritma data mining. Untuk menemukan dan mengkorversi data agar dapat digunakan dalam data mining. Data tersebut dapat di tangani dengan eliminasi inspeksi, identifikasi dan substitusi. Dalam dataset ini yang digunakan adalah dengan mengidentifikasi data dari pencatatan format waktu yang tidak seragam yang dapat mengakibatkan missing data atau tidak terbacanya data serta mengeleminasi data yang tidak lengkap.

- b. Transformasi Data untuk meningkatkan akurasi dan efisiensi algoritma. Data yang digunakan dalam penulisan ini bernilai real. Data ditransformasikan kedalam software Rapidminer.
- c. Data size reduction and discretization, untuk memperoleh data set dengan jumlah atribut yang lebih sedikit tetapi bersifat informative.

Model experiment yang diusulkan pada penelitian ini adalah dengan menerapkan *particle swarm optimization* untuk meningkatkan bobot atribut dari *neural network*. Seperti pada gambar dibawah ini model yang diusulkan dalam penelitian ini mulai dari pemilihan fitur dataset dengan menggunakan PSO hingga menghasilkan hasil dataset dengan attribut yang lebih berpengaruh, dan mengukur peningkatan akurasi dibandingkan model sebelum dan sesudah pemilihan fitur dengan menggunakan algoritma *neural network* adalah sebagai berikut:



Gambar 1. Metode eksperimen

Dalam melakukan penelitian ini diperlukan eksperimen dan proses pengujian model yang diusulkan. Proses eksperimen dan pengujian model menggunakan bagian dari dataset yang ada. Semua dataset kemudian diuji dengan metode yang diusulkan pada aplikasi *Rapidminer 5*.

Evaluasi dan validasi hasil penelitian ini akan menghitung nilai accuracy dalam RMSE dari harga saham dengan menggunakan tools *Rapidminer*, metode/algoritma yang digunakan yaitu *particle swarm optimization* untuk pemilihan fitur pada dataset. Data yang sudah diukur menggunakan *neural network* untuk mendapatkan weight untuk kemudian weight tersebut dilakukan pengukuran untuk mendapatkan weight yang paling berpengaruh dari atribut dalam dataset. Atribut yang paling berpengaruh akan dilakukan training kembali dengan *neural network* diambil hasil terbaik dan kemudian dihitung. Dari hasil yang diperoleh kemudian dibandingkan untuk mengetahui ada perubahan atau tidak. Diharapkan dalam penelitian ini optimasi bekerja dengan baik sehingga bisa meningkatkan akurasi dari sebelumnya.

3. HASIL DAN PEMBAHASAAN

Tujuan dari penelitian ini adalah mengembangkan model yang sudah terbentuk dengan algoritma *neural network*. Data dianalisa dengan melakukan dua perbandingan yaitu menggunakan Algoritma *Neural network* dan Algoritma *Neural network* dengan menggunakan *particle swarm optimization*.

- Pengujian Neural network

Penentuan parameter *neural network* dilakukan dengan mencari nilai terbaik dari momentum, learning rate dan hidden neuron yang digunakan dimulai dengan learning rate. Menentukan nilai learning rate dengan cara melakukan dengan uji coba memasukkan nilai dengan range 0.1 sampai dengan 1. Berikut ini adalah hasil dari percobaan yang telah dilakukan untuk penentuan nilai learning rate.

Tabel 1: Penentuan nilai learning rate

Learning Rate	Training Cycles	Momentum	RMSE
0.1	500	0.2	0.472
0.2	500	0.2	0.471
0.3	500	0.2	0.487
0.4	500	0.2	0.487
0.5	500	0.2	0.577
0.6	500	0.2	0.472
0.7	500	0.2	0.470
0.8	500	0.2	0.569
0.9	500	0.2	0.470
1	500	0.2	0.474

Nilai momentum ditentukan dengan cara melakukan dengan uji coba memasukkan nilai dengan range 0 sampai dengan 0.9. Nilai training cycles dan learning rate dipilih berdasarkan percobaan sebelumnya. Berikut ini adalah hasil dari percobaan yang telah dilakukan untuk penentuan nilai momentum.

Tabel 2: Penentuan nilai momentum

Learning Rate	Training Cycles	Momentum	RMSE
0.8	500	0	0.468
0.8	500	0.1	0.469
0.8	500	0.2	0.469
0.8	500	0.3	0.473
0.8	500	0.4	0.492
0.8	500	0.5	0.467
0.8	500	0.6	0.881
0.8	500	0.7	1.871
0.8	500	0.8	1.919
0.8	500	0.9	2.178
0.8	500	1	2.764

Berdasarkan hasil percobaan di atas, maka untuk parameter *neural network* dipilih nilai momentum sebesar 0, learning rate 0.8. Penentuan jumlah dan size hidden layer dalam percobaan ini digunakan satu di *hiddenlayer* dengan range size dari 1 sampai dengan 30. Dan berikut merupakan hasil percobaan yang dilakukan.

Tabel 3: Penentuan Hidden Layer

Size	RMSE	Size	RMSE
1	3.293	16	1.964
2	0.483	17	1.367
3	2.189	18	0.620
4	0.466	19	0.803
5	0.587	20	0.531
6	0.494	21	2.630
7	1.042	22	0.495
8	0.535	23	0.595
9	0.664	24	0.689
10	0.468	25	0.960
11	0.484	26	0.550
12	0.871	27	0.658
13	0.530	28	34.382
14	0.598	29	0.462
15	0.973	30	0.713

Hasil terbaik pada percobaan hidden layer yaitu hidden layer dengan size 4 dan rmse yang dihasilkan sebesar 0.466.

Hasil dari data tersebut diatas kita dapat mengukur RMSE dengan rumus:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (X_{obs,i} - X_{mo del,i})^2}{n}}$$

$$RMSE = \text{Jumlah (Close - Prediksi)}^2 / \text{jumlah data}$$

$$= (0.15^2 + 0.34^2 + 0.34^2 + 0.06^2 + \dots + 0.8^2) / 133 = 0,4643459877456$$

Tabel 4. Hasil rmse untuk neural network

root_mean_squared_error
0.466 +/- 0.141 (mikro: 0.487 +/- 0.000)
squared_error
0.237 +/- 0.172 (mikro: 0.237 +/- 0.341)

- Pengujian Neural Network dengan Particel Swarm Optimization

Pembuatan model *neural network* dengan Particel swarm optimization dilakukan dengan melakukan normalisasi data seperti pada table, setelah data dinormalisasikan tahap selanjutnya karena menggunakan *neural network* sebagai algoritma pembobotanya maka setelah tahap *neural network* diatas maka proses yang dilakukan adalah melakukan Inisialisasi komponen-komponen pada particle swarm optimization, Jumlah partikel pada *neural network* merupakan representasi dari partikel pada *particle swarm optimization*.

Dari perhitungan tersebut diperoleh hasil dengan perhitungan dengan keseluruhan data diperoleh bahwa attribut yang memiliki atribut yang berpengaruh terhadap bobot atribut yaitu: high atau harga tertinggi yang memiliki bobot 1 dan volume yang juga memiliki bobot 1.

Tabel 5. Bobot atribut pada neural network dengan PSO

Atribut	Weight
Date	0
Open	0
High	1
Low	0
Volume	1

Dengan experiment tersebut kemudian data yang memiliki bobot atribut lebih besar kembali dilakukan training dengan *neural network*. Diketahui bahwa atribut yang bernilai 1 adalah harga tertinggi dan nilai volume. Atribut ini memiliki nilai weight 1 dimana atribut tersebut paling berpengaruh jika isinya mengalami perubahan. Dari hasil diatas kemudian attribute dengan nilai 0 akan dihilangkan dari training pada *neural network*. Dan di lakukan training trail and error kembali dengan *neural network* untuk menentukan hidden size pada neuron mana yang terbaik.

Tabel 6. Experiment Hidden layer PSO-NN

Size	RMSE	Time	Size	RMSE
1	3.293	44.50	16	1.964
2	2.189	46.37	17	1.367
3	0.373	47.45	18	0.620
4	0.397	48.15	19	0.803
5	0.384	48.55	20	0.380
6	0.396	59.10	21	2.630
7	1.042	51.37	22	0.459
8	0.468	49.32	23	0.459
9	0.664	45.10	24	0.570
10	0.398	46.16	25	0.960
11	0.388	45.28	26	0.441
12	0.871	50.45	27	0.658
13	0.530	45.30	28	34.382
14	0.598	43.51	29	0.462
15	0.973.	48.33	30	0.713

Hasil dari pengujian model yang dilakukan adalah memprediksi harga saham dengan *neural network* dan *neural network* dengan particle swarm optimization untuk menentukan nilai rmse. Dalam menentukan nilai tingkat keakurasian dalam model *neural network* dan algoritma *neural network* berbasis particle swarm optimization. Metode pengujiannya menggunakan cross sebagai berikut.

Hasil *neural network* dengan menggunakan particle swarm optimization dan didapat hasil percobaan sebagai berikut:

$$\text{RMSE} = \text{Jumlah (Close - Prediksi)}^2 / \text{jumlah data}$$

$$= (0.05^2 + 0.35^2 + 0.29^2 + 0.02^2 + 0,87^2 \dots + 0.8^2) / 133 = 0,373256784268$$

Table: Hasil Pengujian *Neural network* dengan Particel swarm optimization

root_mean_squared_error

0,373 +/- 0.201 (mikro: 0.372 +/- 0.000)
--

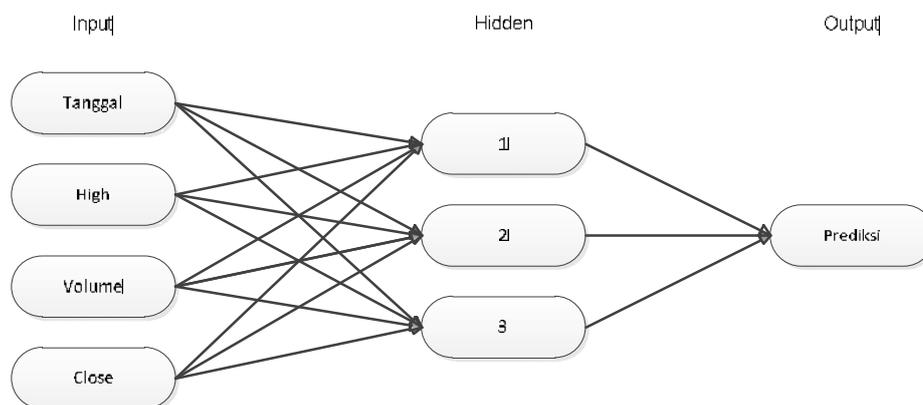
squared_error

0.205 +/- 0.272 (mikro: 0.205 +/- 0.141)
--

Root mean square error (rmse) digunakan untuk mengukur kinerja dari *neural network* dengan particle swarm optimization rmse yang dihasilkan adalah 0,373.

Dalam penelitian ini dilakukan pengujian model dengan menggunakan *neural network* dan *neural network* dengan *particleswarmoptimization* dengan menggunakan data harga saham. Model yang dihasilkan diuji untuk mendapatkan nilai RMSE yang lebih rendah, didalam pengujian tersebut *neural network* mendapatkan nilai rmse sebesar 0,466 sedangkan *neural network* dengan pemilihan fitur dengan menggunakan particle swarm optimization mendapat nilai rmse 0,373 atau 0,093 lebih kecil dari menggunakan *neural network* saja. Dengan demikian dapat disimpulkan bahwa *neural network* dengan *particleswarmoptimization* untuk pemilihan fitur pada harga saham lebih baik dari pada dengan menggunakan *neural network* saja. Dapat disimpulkan bahwa *neural network* dengan particle swarm optimization untuk pemilihan fitur pada harga saham dapat memecahkan masalah prediksi menjadi lebih akurat.

Adapun model yang diperoleh dari penelitian tersebut dapat dilihat dari gambar dibawah ini:



Gambar 2. Model Neural network dengan optimisasi dengan PSO

Masukan tidak mempunyai pengaruh akan dihilangkan sedangkan *hidden layer* yang efektif dalam percobaan ini menggunakan 3 *hidden layer*.

Walaupun dalam penelitian ini penggunaan algoritma *neural network* dengan particle swarm optimization sudah lebih baik, namun ada beberapa hal yang dapat ditambahkan untuk penelitian selanjutnya:

- Dapat menerapkan algoritma optimasi penyesuaian nilai parameter untuk digabungkan dengan pemilihan fitur mendapatkan model yang benar-benar akurat.
- Dapat dilakukan komparasi dengan menggunakan algoritma lain dengan menggunakan pemilihan fitur dari algoritma optimasi yang sama.
- Penelitian dapat menjadi masukan bagi para perdisksi saham untuk dapat mengaplikasikan langsung kelalam saham agar dapat menjadi pertimbangan dalam mengambil keputusan dlam pasar bursa saham.

4. KESIMPULAN

Dalam penelitian ini dilakukan pengujian model dengan menggunakan *neural network* dan *neural network* dengan *particleswarmoptimization* dengan menggunakan data harga saham. Model yang dihasilkan diuji untuk mendapatkan nilai RMSE yang lebih rendah, didalam pengujian tersebut *neural network* mendapatkan nilai rmse sebesar 0,466 sedangkan *neural network* dengan pemilihan fitur dengan menggunakan particle swarm optimization mendapat nilai rmse 0,373 atau 0,093 lebih kecil dari menggunakan *neural network* saja. Dengan demikian dapat disimpulkan bahwa *neural network* dengan *particleswarmoptimization* untuk pemilihan fitur pada harga saham lebih baik dari pada dengan menggunakan *neural network* saja. Dapat disimpulkan bahwa *neural network* dengan particle swarm optimization untuk pemilihan fitur pada harga saham dapat memecahkan masalah prediksi menjadi lebih akurat.

DAFTAR PUSTAKA

- Kara, Y., Boyacioglu, M. A., & Baykan, Ö. K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. *Expert Systems with Applications*, 38, 5311–5319.
- Abraham, B., & Merola, G. (2005). Dimensionality Reduction Approach to Multivariate Prediction. *Computational Statistics and Data Analysis*, 48, 5-16.
- Chen, S.-H. H.-H. (2009). Adaptive image interpolation using probabilistic neural network. *Expert Systems with Applications*, 36 (3), 6025-6029.
- Chen, Z., & Lu, K. (2006). A Preprocess Algorithm of Filtering Irrelevant Information Based on The Minimum Class Difference. *Knowledge-Based System*, 19, 422-429.
- Darmadji, T., & Hendy, F. M. (2005). *Pasar Modal Indonesia* (2nd ed.). Jakarta, Indonesia: Salemba Empat.
- Gorunescu, F. (2011). *Data Mining Concept And Technique*. Craiova, Romania: Springer.
- Heaton, J. (2008). *Introduction to Neural Networks* (2nd ed.). St. Louis: Heaton Research, Inc.
- Kim, K. i. (2010). Genetic algorithms approach to feature discretization in artificial neural network for the prediction of stock price index. *Expert System with Application* (19), 125-132.
- Kuok, K. K. (2010). Particle Swarm Optimization Feedforward Neural Network for Hourly Rain fall run off Modeling in Bedup Basin, Malaysia. *International Journal of Civil & Environmental Engineering*, 9 (10).
- Maimon, O. (2010). *Data Mining and Knowledge Discovery Handbook* (2nd ed.). (O. Maimon, & L. Rokach, Eds.) Israel: Springer.
- Martinez, L. C. (2009). From an artificial neural network to a stock market day-trading system: A case study on the BM&F BOVESPA. *International Joint Conference on Neural Networks*.
- Ming Hao Eng. (2008). Forecast Forex With ANN Using Fundamental Data. *International Conference on Information Management, Innovation Management and Industrial Engineering*. Singapore.
- Ravisankar P, R. V. (2012). Detection of Financial Statement Fraud and Feature Selection Using Data Mining Techniques. *Decision Support Systems*, 50, 491-500.
- Sheng, W. (2009). Chinese Grain Production Forecasting Method Based on Particle Swarm Optimization-based Support Vector Machine. *Recent Patents on Engineering*, 3.
- Vanstone, B. (2009). An Empirical Methodology for Developing Stockmarket Trading Systems using Artificial Neural Networks. *Information Technology papers*.
- Wijaya, Y. B. (2010). Stock price prediction: comparison of Arima and artificial neural network methods. *2010 Second International Conference on Advances in Computing, Control, and Telecommunication Technologies*, (p. 4). Jakarta, Indonesia.
- Yakup, K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. *Expert Systems with Applications*, 38 (5), 5311-5319.
- Zhu, F., & Guan, S. (2004). Feature Selection for Modular GA-based Classification. *Applied Soft Computing*, 4, 381-393.