

PREDIKSI HASIL PEMILU LEGISLATIF DKI JAKARTA DENGAN METODE NEURAL NETWORK BERBASIS PARTICLE SWARM OPTIMIZATION

Mohammad Badrul

Jurusan Teknik Informatika

Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri

Jl. Kramat raya No. 25 Jakarta Pusat

Mohammad.mbl@nusamandiri.ac.id

ABSTRACT

General elections are a means of implementation of the sovereignty of the people in the Unitary State of Indonesia based on Pancasila and 1945 Constitution. Elections held in Indonesia is to choose the leadership of both the president and vice president, member of parliament, parliament, and the DPD. The related research of general election usually using decision tree algorithm or neural network algorithm. Each of method has strong and weakness, but neural network algorithm can solve problem in decision tree algorithm. The accuracy using neural network algorithm in predicting the election has less accurate. In this study created a model neural network algorithm and neural network algorithm model based genetic algorithm to get the rule in predicting the outcome of legislative elections and provide a more accurate value of accuracy. After testing the two models namely neural network algorithm and neural network algorithm based on particle swarm optimization, the results obtained are the neural network algorithm produces an accuracy value by 98,50% and the AUC value of 0.982, but after the addition of neural network algorithm based on particle swarm optimization value of 98,85 % accuracy and AUC value of 0.996. So both methods have accuracy rate of 0.35 % difference and the difference in the AUC of 0.14.

Keyword: Elections, neural network algorithm, particle swarm optimization algorithm, accuracy.

I. Pendahuluan

Pemilihan umum adalah sarana pelaksanaan kedaulatan rakyat dalam negara kesatuan Republik Indonesia yang berdasarkan pancasila dan UUD 1945 (Undang-Undang RI No.10, 2008). Pemilihan umum adalah salah satu pilar utama untuk memilih pemimpin dari sebuah demokrasi atau bisa disebut yang terutama (Santoso, 2004). Pemilu merupakan sarana yang sangat penting bagi terselenggaranya sistem politik yang demokratis. Karena itu, tidak mengherankan banyak negara yang ingin disebut sebagai negara demokratis menggunakan pemilu sebagai mekanisme membangun legitimasi. Pemilu bertujuan untuk memilih anggota DPR, DPRD provinsi, dan DPRD kabupaten/kota yang dilaksanakan dengan sistem proporsional terbuka (Undang-Undang RI No.10, 2008). Dengan sistem pemilu langsung dan jumlah partai yang besar maka pemilu legislatif memberikan peluang yang besar pula bagi rakyat Indonesia untuk berkompetisi menaikkan diri menjadi anggota legislatif. Pemilu legislatif tahun 2009 diikuti sebanyak

44 partai yang terdiri dari partai nasional dan partai lokal. Pemilu Legislatif DKI Jakarta Tahun 2009 terdapat 2.268 calon anggota DPRD dari 44 partai yang akan bersaing memperebutkan 94 kursi anggota Dewan Perwakilan Rakyat DKI Jakarta. Prediksi hasil pemilihan umum perlu diprediksi dengan akurat, karena hasil prediksi yang akurat sangat penting karena mempunyai dampak pada berbagai macam aspek sosial, ekonomi, keamanan, dan lain-lain (Borisjuk, Borisjuk, Rallings, & Thrasher, 2005). Bagi para pelaku ekonomi, peristiwa politik seperti pemilu tidak dapat dipandang sebelah mata, mengingat hal tersebut dapat mengakibatkan risiko positif maupun negatif terhadap kelangsungan usaha yang dijalankan.

Metode prediksi hasil pemilihan umum sudah pernah dilakukan oleh peneliti (Rigdon, Jacobson, Sewell, & Rigdon, 2009) melakukan prediksi hasil pemilihan umum dengan menggunakan metode *Estimator Bayesian*. (Moscato, Mathieson, Mendes, & Berreta, 2005) melakukan penelitian untuk memprediksi pemilihan presiden Amerika Serikat menggunakan *decision tree*. (Choi &

Han, 1999) memprediksi hasil pemilihan presiden di Korea dengan metode Decision Tree. (Nagadevara & Vishnuprasad, 2005) memprediksi hasil pemilihan umum dengan model *classification tree dan neural network*. (Borisyyuk, Borisyyuk, Rallings, & Thrasher, 2005) yang memprediksi hasil pemilihan umum dengan menggunakan metode *neural network*.

Decision tree mempunyai kelebihan yaitu mempunyai kelebihan dalam 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 di atasnya, pohon keputusan yang dihasilkan tidak optimal dan tidak bisa menggunakan sampel yang lebih besar (Sug, 2009), karena itu tidak mudah untuk memahami pohon keputusan besar dan masalah *overfitting* data bisa terjadi dengan target data terbatas yang ditetapkan.

Neural network dapat memecahkan masalah decision tree 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 hasil pemilu pada penelitian ini. *Neural network* merupakan metode yang sering digunakan untuk memprediksi hasil pemilu legislatif karena data yang di sajikan untuk metode ini harus besar dan *non linear* (Gill, 2005). Teknik paling populer pada metode *neural network* adalah algoritma *backpropation* yang banyak digunakan untuk memecahkan banyak masalah di dunia nyata dengan membangun model terlatih yang menunjukkan kinerja yang baik dalam beberapa masalah *non-liner* (Park, Lee, & Choi, 2009). Namun algoritma *backpropagation* mempunyai beberapa kelemahan. Pertama, algoritma *backpropagation* bisa terjebak dalam masalah lokal minimum, hal ini dapat menyebabkan kegagalan untuk mencari solusi yang optimal dalam pemilihan fitur pada bobot atribut yang digunakan. Kedua, 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 (Park, Lee, & Choi, 2009). Masalah umum lainnya adalah mempunyai kelemahan pada perlunya data training yang besar dan optimasi yang digunakan kurang efisien (Xiao & Shao, 2011). Hal ini dapat dipecahkan karena jumlah

data training pada penelitian ini sebanyak 2268 record.

Particle swarm optimization(PSO) merupakan algoritma optimasi yang efektif yang dapat memecahkan masalah yang ada pada algoritma *neural network* yang pada umumnya menggunakan algoritma *backpropagation* (Park, Lee, & Choi, 2009). *Particle swarm optimization* 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 *particle swarm optimization* adalah interaksi sosial yang mempromosikan pembagian informasi antara partikel yang akan membantu dalam pencarian solusi yang optimal (Park, Lee, & Choi, 2009). *Particle swarm optimization* memiliki beberapa parameter seperti posisi, kecepatan, kecepatan maksimum, percepatan konstanta dan berat inersia. 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*.

Pada penelitian ini *particle swarm optimization* 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 pemilu legislatif DKI Jakarta lebih akurat.

II. Kajian Literatur

2.1 Pemilihan Umum

Pemilihan umum adalah salah satu pilar utama dari sebuah demokrasi, kalau tidak dapat yang disebut yang terutama. Pemilu di Indonesia terbagi dari dua bagian, yaitu (Sardini, 2011): yaitu Pemilu orde baru yaitu Sistem pemilihannya dilakukan secara proporsional tidak murni, yang artinya jumlah penentuan kursi tidak ditentukan oleh jumlah penduduk saja tetapi juga didasarkan pada wilayah administrasi dan pemilu era reformasi yaitu dikatakan sebagai pemilu reformasi karena dipercepatnya proses pemilu di tahun 1999 sebelum habis masa kepemimpinan di pemilu tahun 1997. Terjadinya pemilu era reformasi ini karena produk pemilu pada tahun 1997 dianggap pemerintah dan lembaga lainnya tidak dapat dipercaya.

Sistem pemilihan DPR/DPD berdasarkan ketentuan dalam UU nomor 10 tahun 2008 pasal 5 ayat 1 sistem yang digunakan dalam pemilihan legislatif adalah sistem proporsional dengan daftar terbuka, sistem pemilihan DPD dilaksanakan dengan sistem distrik berwakil banyak UU nomor 10 tahun 2008 pasal 5 ayat 2. Menurut UU No. 10 tahun 2008, Peserta pemilihan anggota DPR/D adalah partai politik peserta Pemilu, sedangkan peserta pemilihan anggota DPD adalah perseorangan. Partai politik peserta Pemilu dapat mengajukan calon sebanyak-banyaknya 120 persen dari jumlah kursi yang diperebutkan pada setiap daerah pemilihan demokratis dan terbuka serta dapat mengajukan calon dengan memperhatikan keterwakilan perempuan sekurang-kurangnya 30 %. Partai Politik Peserta Pemilu diharuskan UU untuk mengajukan daftar calon dengan nomor urut (untuk mendapatkan Kursi). Karena itu dari segi pencalonan UU No.10 Tahun 2008 mengadopsi sistem daftar calon tertutup.

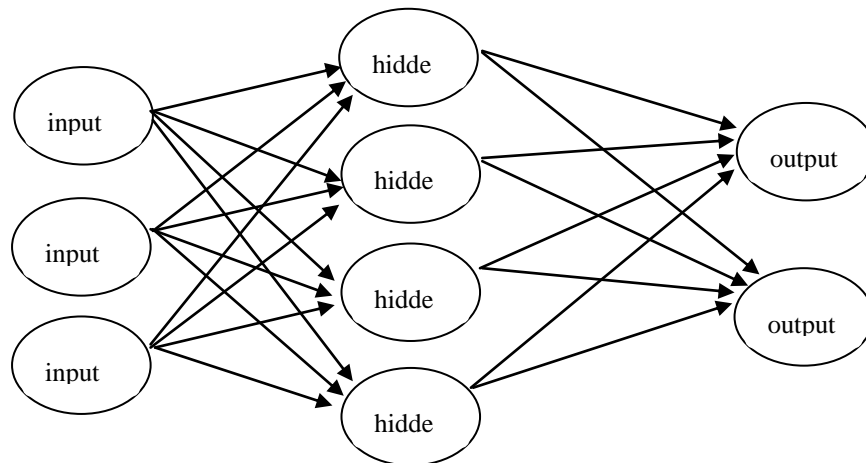
UU No.10 Tahun 2008 mengadopsi sistem proporsional dengan daftar terbuka. sistem proporsional merujuk pada formula pembagian kursi dan/atau penentuan calon terpilih, yaitu setiap partai politik peserta

pemilu mendapatkan kursi proporsional dengan jumlah suara sah yang diperolehnya. Penerapan formula proporsional dimulai dengan menghitung bilangan pembagi pemilih (BPP), yaitu jumlah keseluruhan suara sah yang diperoleh seluruh partai politik peserta pemilu pada suatu daerah pemilihan dibagi dengan jumlah kursi yang diperebutkan pada daerah pemilihan tersebut.

2.2 Neural Network

Neural network adalah suatu sistem pemroses informasi yang memiliki karakteristik menyerupai dengan jaringan saraf biologi pada manusia. *Neural network* didefinisikan sebagai sistem komputasi di mana arsitektur dan operasi diilhami dari pengetahuan tentang sel saraf biologis di dalam otak, yang merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba menstimulasi proses pembelajaran pada otak manusia tersebut (Astuti, 2009).

Neural network dibuat berdasarkan model saraf manusia tetapi dengan bagian-bagian yang lebih sederhana. Komponen terkecil dari *neural network* adalah unit atau yang biasa disebut dengan *neuron* dimana *neuron* tersebut akan mentransformasikan informasi yang diterima menuju *neuron* lainnya (Shukla, Tiwari, & Kala, 2010).



Gambar 1. Model *neural network*

Neural network terdiri dari dua atau lebih lapisan, meskipun sebagian besar jaringan terdiri dari tiga lapisan: lapisan input, lapisan tersembunyi, dan lapisan output (Larose, 2005). Pendekatan *neural network* dimotivasi oleh jaringan saraf biologis. Secara kasar, *neural network* adalah satu set terhubung *input/output* unit, di mana masing-masing sambungan memiliki berat yang terkait

dengannya. *Neural network* memiliki beberapa properti yang membuat mereka populer untuk *clustering*. Pertama, *neural network* adalah arsitektur pengolahan *inherent* paralel dan terdistribusi. Kedua, *neural network* belajar dengan menyesuaikan bobot interkoneksi dengan data, Hal ini memungkinkan *neural network* untuk "menormalkan" pola dan bertindak sebagai fitur (atribut) *extractors*

untuk kelompok yang berbeda. Ketiga, *neural network* memproses vektor numerik dan membutuhkan pola objek untuk diwakili oleh fitur kuantitatif saja (Gorunescu, 2011).

Neural network terdiri dari kumpulan node (*neuron*) dan relasi. Ada tiga tipe node (*neuron*) yaitu, *input*, *hidden* dan *output*. Setiap relasi menghubungkan dua buah node dengan bobot tertentu dan juga terdapat arah yang menunjukkan aliran data dalam proses (Kusrini & Luthfi, 2009). Kemampuan otak manusia seperti mengingat, menghitung, menggeneralisasi, adaptasi, diharapkan *neural network* dapat meniru kemampuan otak manusia. *Neural network* berusaha meniru struktur/arsitektur dan cara kerja otak manusia sehingga diharapkan bisa dan mampu menggantikan beberapa pekerjaan manusia. *Neural network* berguna untuk memecahkan persoalan yang berkaitan dengan pengenalan pola, klasifikasi, prediksi dan data mining (Shukla, Tiwari, & Kala, 2010).

Input node terdapat pada *layer* pertama dalam *neural network*. Secara umum setiap *input node* merepresentasikan sebuah input parameter seperti umur, jenis kelamin, atau pendapatan. *Hidden node* merupakan node yang terdapat di bagian tengah. *Hidden node* ini menerima masukan dari *input node* pada *layer* pertama atau dari *hidden node* dari *layer* sebelumnya. *Hidden node* mengombinasikan semua masukan berdasarkan bobot dari relasi yang terhubung, mengkalkulasikan, dan memberikan keluaran untuk *layer* berikutnya. *Output node* mempresentasikan atribut yang diprediksi (Kusrini & Luthfi, 2009).

Setiap node (*neuron*) dalam *neural network* merupakan sebuah unit pemrosesan. Tiap node memiliki beberapa masukan dan sebuah keluaran. Setiap node mengombinasikan beberapa nilai masukan, melakukan kalkulasi, dan membangkitkan nilai keluaran (aktifasi). Dalam setiap node terdapat dua fungsi, yaitu fungsi untuk mengombinasikan masukan dan fungsi aktifasi untuk menghitung keluaran. Terdapat beberapa metode untuk mengombinasikan masukan antara lain *weighted sum*, *mean*, *max*, logika OR, atau logika AND (Kusrini & Luthfi, 2009). Serta beberapa fungsi aktifasi yang dapat digunakan yaitu *heaviside (threshold)*, *step activation*, *piecewise*, *linear*, *gaussian*, *sigmoid*, *hyperbolic tangent* (Gorunescu, 2011).

Salah satu keuntungan menggunakan *neural network* adalah bahwa *neural network* cukup kuat sehubungan dengan data. Karena *neural network* berisi banyak node (*neuron*

buatan) dengan bobot ditugaskan untuk setiap koneksi (Larose, 2005).

Aplikasi *neural network* telah banyak dimanfaatkan untuk berbagai kepentingan seperti di bidang Elektronik, Otomotif, Perbankan, Sistem penerbangan udara, Dunia hiburan, transportasi publik, telekomunikasi, bidang Kesehatan, Keamanan, bidang Robotika, Asuransi, Pabrik, Financial, Suara, Pertambangan dan sistem pertahanan (Astuti, 2009). Metode pelatihan terbimbing/terawasi (*supervised learning*) adalah pelatihan yang memasukkan target keluaran dalam data untuk proses pelatihan. Ada beberapa metode pelatihan terbimbing yang telah ditemukan oleh para peneliti, diantaranya adalah yang paling sering diaplikasikan adalah *persepton* dan *backpropagation (feedforward)*.

Algoritma yang paling populer pada algoritma *neural network* adalah algoritma *backpropagation*. 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*, di mana ciri dari metode ini adalah meminimalkan *error* pada output yang dihasilkan oleh jaringan.

Algoritma *backpropagation* mempunyai pengatuaran hubungan yang sangat sederhana yaitu: jika keluaran memberikan hasil yang salah, maka penimbang (*weight*) dikoreksi supaya galatnya dapat diperkecil dan respon jaringan selanjutnya diharapkan akan mendekati nilai yang benar. Algoritma ini juga berkemampuan untuk memperbaiki penimbang pada lapisan tersembunyi (*hidden layer*) (Purnomo & Kurniawan, 2006).

Inisialisasi awal bobot jaringan *backpropagation* yang terdiri atas lapisan input, lapisan tersembunyi, dan lapisan 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). Sedangkan penyelesaian masalah akan dilakukan jika

proses pelatihan tersebut telah selesai, fase ini disebut dengan fase *mapping* atau proses pengujian/testing.

Berikut langkah pembelajaran dalam algoritma *backpropagation* adalah sebagai berikut (Myatt, 2007):

1. Inisialisasi bobot jaringan secara acak (biasanya antara -0.1 sampai 1.0)
2. Untuk setiap data pada data *training*, hitung input untuk simpul berdasarkan nilai input dan bobot jaringan saat itu, menggunakan rumus:

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

3. Berdasarkan input dari langkah dua, selanjutnya membangkitkan output. Untuk simpul menggunakan fungsi aktivasi sigmoid:

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

4. Hitung nilai *Error* antara nilai yang diprediksi dengan nilai yang sesungguhnya menggunakan rumus:

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

5. Setelah nilai *Error* dihitung, selanjutnya dibalik ke *layer* sebelumnya (*backpropagation*). Untuk menghitung nilai *Error* pada *hidden layer*, menggunakan rumus:

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

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

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

2.3 Particle Swarm Optimization

Feature Selection terkait erat dengan masalah pengurangan dimensi dimana tujuannya adalah untuk mengidentifikasi fitur dalam kumpulan data-sama pentingnya, dan membuang fitur lain seperti informasi yang tidak relevan dan berlebihan dan akurasi dari seleksinya pada masa depan yang dapat ditingkatkan. Pengurangan dimensi tersebut dilakukan dengan menekan seminimal mungkin kerugian yang dapat terjadi akibat kehilangan sebagian informasi. Tujuan pengurangan dimensi dalam domain *data mining* adalah untuk mengidentifikasi biaya terkecil di mana algoritma *data mining* dapat menjaga tingkat kesalahan di bawah perbatasan garis efisiensi (Maimon & Rokach, 2010).

Masalah *feature selection* mengacu pada pemilihan fitur yang sesuai yang harus

diperkenalkan dalam analisis untuk memaksimalkan kinerja dari model yang dihasilkan. *Feature selection* adalah proses komputasi, yang digunakan untuk memilih satu set fitur yang mengoptimalkan langkah evaluasi yang mewakili kualitas fitur (Salappa, Doumpos, & Zopounidis, 2007).

Sebuah algoritma *feature selection* ditandai dengan strategi yang digunakan untuk mencari subset yang tepat dari fitur, proses seleksi fitur, ukuran evaluasi yang digunakan untuk menilai kualitas fitur dan interaksi dengan metode klasifikasi yang digunakan untuk mengembangkan model akhir (Maimon & Rokach, 2010). Salah satu metode yang sering digunakan adalah metode Particle swarm optimization.

Particle Swarm Optimization (PSO) adalah teknik optimasi berbasis populasi yang dikembangkan oleh Eberhart dan Kennedy pada tahun 1995, yang terinspirasi oleh perilaku sosial kawanan burung atau ikan (Park, Lee, & Choi, 2009). *Particle swarm optimization* dapat diasumsikan sebagai kelompok burung secara mencari makanan disuatu daerah. Burung tersebut tidak tahu dimana makanan tersebut berada, tapi mereka tahu seberapa jauh makanan itu berada, jadi strategi terbaik untuk menemukan makanan tersebut adalah dengan mengikuti burung yang terdekat dari makanan tersebut (Salappa, Doumpos, & Zopounidis, 2007). *Particle swarm optimization* digunakan untuk memecahkan masalah optimasi.

Serupa dengan algoritma genetik (GA), *Particle swarm optimization* melakukan pencarian menggunakan populasi (*swarm*) dari individu (partikel) yang akan diperbaharui dari iterasi. *Particle swarm optimization* memiliki beberapa parameter seperti posisi, kecepatan, kecepatan maksimum, konstanta percepatan, dan berat inersia. *Particle swarm optimization* memiliki perbandingan lebih atau bahkan pencarian kinerja lebih unggul untuk banyak masalah optimasi dengan lebih cepat dan tingkat *konvergensi* yang lebih stabil (Park, Lee, & Choi, 2009).

Untuk menemukan solusi yang optimal, masing-masing partikel bergerak ke arah posisi yang terbaik sebelumnya dan posisi terbaik secara global. Sebagai contoh, partikel ke-*i* dinyatakan sebagai: $x_i = (x_{i1}, x_{i2}, \dots, x_{id})$ dalam ruang *d*-dimensi. Posisi terbaik sebelumnya dari partikel ke-*i* disimpan dan dinyatakan sebagai $pbest_i = (pbest_{i1}, pbest_{i2}, \dots, pbest_{id})$. Indeks partikel terbaik diantara semua partikel dalam kawanan group dinyatakan sebagai $gbest_d$. Kecepatan partikel dinyatakan sebagai: $v_i = (v_{i1}, v_{i2}, \dots, v_{id})$. Modifikasi kecepatan

dan posisi partikel dapat dihitung menggunakan kecepatan saat ini dan jarak $pbest_i$, $gbest_d$ seperti ditunjukkan persamaan berikut:

$$v_{i,d} = w * v_{i,d} + c1 * R * (pbest_{i,d} - x_{i,d}) + c2 * R * (gbest_d - x_{i,d}) \quad (2.1)$$

$$x_{i,d} = x_{i,d} + v_{i,d} \quad (2.2)$$

Dimana:

- $V_{i,d}$ = Kecepatan partikel ke-i pada iterasi ke-i
- w = Faktor bobot inersia
- $c1, c2$ = Konstanta akselerasi (learning rate)
- R = Bilangan random (0-1)
- $X_{i,d}$ = Posisi saat ini dari partikel ke-i pada iterasi ke-i
- $pbest_i$ = Posisi terbaik sebelumnya dari partikel ke-i
- $gbest_i$ = Partikel terbaik diantara semua partikel dalam satu kelompok atau populasi
- n = Jumlah partikel dalam kelompok
- d = Dimensi

Persamaan (2.1) menghitung kecepatan baru untuk tiap partikel (solusi potensial) berdasarkan pada kecepatan sebelumnya ($V_{i,m}$), lokasi partikel dimana nilai *fitness* terbaik telah dicapai ($pbest$), dan lokasi populasi global ($gbest$ untuk versi global, $lbest$ untuk versi local) atau local *neighborhood* pada algoritma versi local dimana nilai *fitness* terbaik telah dicapai.

Persamaan (2.2) memperbaharui posisi tiap partikel pada ruang solusi. Dua bilangan acak $c1$ dan $c2$ dibangkitkan sendiri. Penggunaan berat inersia w telah memberikan performa yang meningkat pada sejumlah aplikasi. Hasil dari perhitungan partikel yaitu kecepatan partikel diantara interval $[0,1]$ (Park, Lee, & Choi, 2009).

III. Metode Penelitian

Menurut Sharp et al (Dawson, 2009) penelitian adalah mencari melalui proses yang metodis untuk menambahkan pengetahuan itu sendiri dan dengan yang lainnya, oleh penemuan fakta dan wawasan tidak biasa. Pengertian lainnya, penelitian adalah sebuah kegiatan yang bertujuan untuk membuat kontribusi orisinal terhadap ilmu pengetahuan (Dawson, 2009).

Penelitian ini adalah penelitian eksperimen dengan metode penelitian sebagai berikut

1. Pengumpulan data

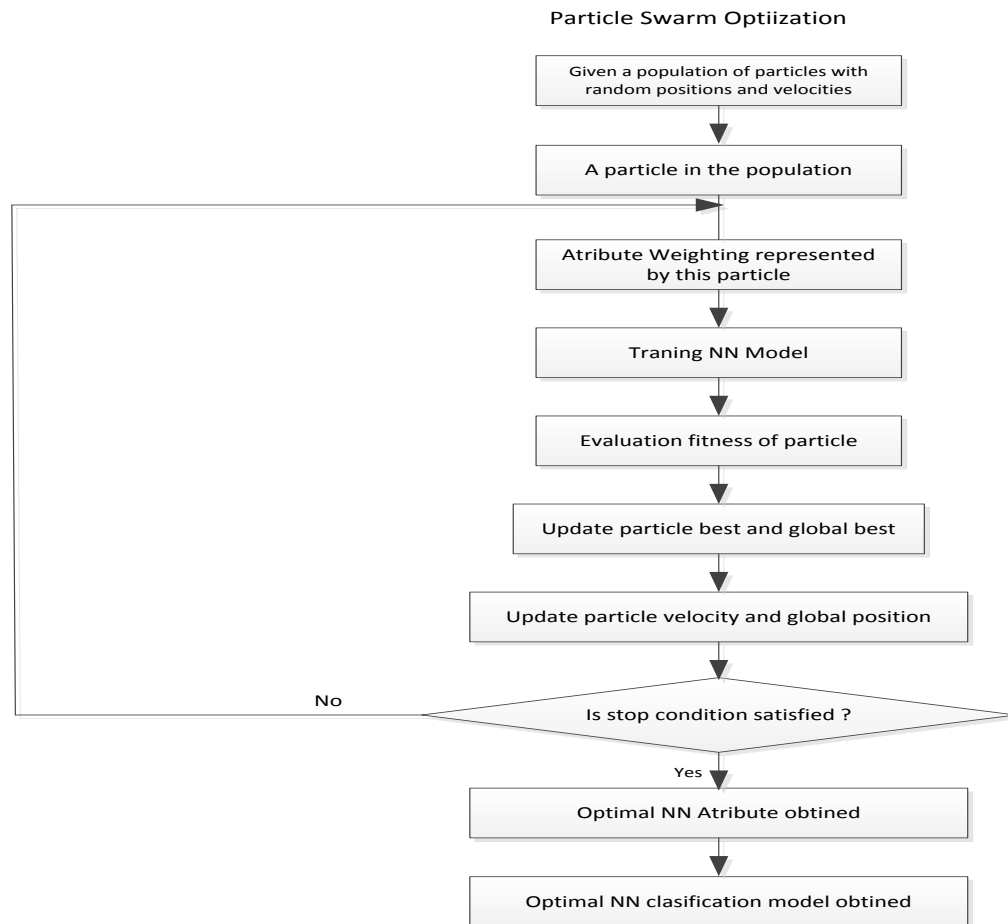
Pada pengumpulan data dijelaskan tentang bagaimana dan darimana data dalam penelitian ini didapatkan, ada dua tipe dalam pengumpulan data, yaitu pengumpulan data primer dan pengumpulan data sekunder. Data primer adalah data yang dikumpulkan pertama kali untuk melihat apa yang sesungguhnya terjadi. Data sekunder adalah data yang sebelumnya pernah dibuat oleh seseorang baik di terbitkan atau tidak (Kothari, 2004). Dalam pengumpulan data primer dalam penelitian ini menggunakan metode observasi dan interview, dengan menggunakan data-data yang berhubungan dengan pemilu ditahun 2009. Data yang didapat dari KPUD Jakarta adalah data pemilu tahun 2009 dengan jumlah data sebanyak 2268 *record*, terdiri dari 11 variabel atau atribut. Adapaun variabel yang digunakan yaitu no urut partai, nama partai, suara sah partai, no urut caleg, nama caleg, jenis kelamin, kota adminstrasi, daerah pemilihan, suarah sah caleg, jumlah perolehan kursi. Sedangkan varaibel tujuannya yaitu hasil pemilu.

2. Pengolahan awal data

Jumlah data awal yang diperoleh dari pengumpulan data yaitu sebanyak 2.268 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 yaitu (Vercellis, 2009): data validation, data integration and transformation dan data size reduction and discretization. Sehingga diperoleh atribut antara lain, jenis kelamin, no.urut parpol, suara sah partai, jumlah perolehan kursi, daerah pemilihan, nomor urut caleg dan suara sah caleg.

3. Model yang diusulkan

Model yang diusulkan pada penelitian ini berdasarkan *state of the art* tentang prediksi hasil pemilihan umum adalah dengan menerapkan *neural network* dan *neural network* berbasis *Particle swarm optimization*, yang terlihat pada Gambar dibawah ini



Gambar 2. Model yang diusulkan

Pada Gambar 3.1. menunjukkan proses yang dilakukan dalam tahap modeling untuk menyelesaikan prediksi hasil pemilihan umum dengan menggunakan dua metode yaitu algoritma *neural network* dan algoritma *neural network* dengan *Particle swarm optimization*.

4. Eksperimen dan pengujian model

Untuk memilih arsitektur *neural network* yang tepat, agar menghasilkan nilai akurasi dan nilai AUC yang terbesar, diperlukan pengaturan (*adjustment*) untuk parameter-parameter *neural network* antara lain *training cycle*, *learning rate*, *momentu*, *hidden layer* dan *neuron size*. Untuk memilih arsitektur *neural network* yang tepat, agar menghasilkan nilai akurasi dan nilai AUC yang terbesar, diperlukan pengaturan (*adjustment*) untuk parameter-parameter *neural network*. Berikut ini adalah parameter-parameter yang membutuhkan *adjustment*:

a. *Training cycle*, *learning rate*, dan *momentum*

Training cycle adalah jumlah perulangan training yang perlu dilakukan untuk mendapatkan error yang terkecil. Nilai *training cycle* bervariasi mulai dari 1

sampai dengan tak terhingga. *Learning rate* adalah variabel yang digunakan oleh algoritma pembelajaran untuk menentukan bobot dari *neuron* (K & Deepa, 2011). Nilai yang besar menyebabkan pembelajaran lebih cepat tetapi ada osilasi bobot, sedangkan nilai yang kecil menyebabkan pembelajaran lebih lambat. Nilai *learning rate* harus berupa angka positif kurang dari 1. *Momentum* digunakan untuk meningkatkan *convergence*, mempercepat waktu pembelajaran dan mengurangi osilasi. Nilai momentum bervariasi dari 0 ke 1.

b. *Hidden Layer*

Ada 2 masalah dalam pengaturan *hidden layer*, yaitu penentuan jumlah *hidden layer* dan penentuan *size* atau jumlah *neuron* dari *hidden layer*. Saat ini tidak ada alasan teoritis untuk menggunakan *neural network* dengan lebih dari dua *hidden layer*. Bahkan, untuk banyak masalah praktis, tidak ada alasan untuk menggunakan lebih dari satu *hidden layer* (K & Deepa, 2011). Penentuan jumlah *neuron* yang terlalu sedikit akan mengakibatkan *underfitting*, yaitu jaringan kurang dapat mendeteksi

sinyal atau pola dalam set data. Jumlah *neuron* yang terlalu banyak akan mengakibatkan *overfitting*, yaitu jumlah informasi dalam *training set* yang terbatas, tidak cukup untuk melatih semua *neuron* dalam *hidden layer*.

c. *Arsitektur neural network*

Arsitektur neural network tersusun dari tiga buah lapisan (*layer*), yaitu *input layer*, *hidden layer*, dan *output layer*. *hidden layer* terletak diantara *input* dan *output*.

5. Evaluasi dan validasi hasil

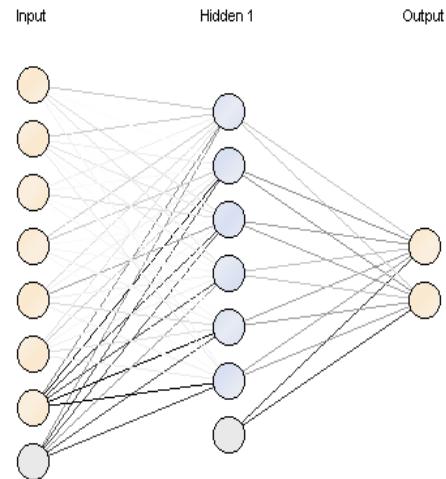
Setelah ditemukan nilai akurasi yang paling ideal dari parameter di atas akan terbentuk struktur algoritma yang ideal untuk pemecahan masalah tersebut. Model yang diusulkan pada penelitian tentang prediksi hasil pemilihan umum adalah dengan menerapkan *neural network* dan *neural network* berbasis *Particle swarm optimization*. Penerapan algoritma *neural network* dengan menentukan nilai *training cycle* terlebih dahulu. Setelah didapatkan nilai akurasi dan AUC terbesar, nilai *training cycle* tersebut akan dijadikan nilai yang akan digunakan untuk mencari nilai akurasi dan AUC tertinggi pada *learning rate* dan *momentum*. Setelah ditemukan nilai yang paling tinggi dari *training cycle*, *learning rate* dan *momentum* selanjutnya adalah menentukan ukuran (*size*) pada *hidden layer* tersebut. Sedangkan penerapan algoritma *neural network* berbasis *Particle swarm optimization* berdasarkan pada nilai *training cycle* pada algoritma tersebut. Setelah ditemukan nilai akurasi yang paling ideal dari parameter tersebut langkah selanjutnya adalah menentukan nilai *learning rate* dan *momentum*. Setelah ditemukan nilai akurasi yang paling ideal dari parameter tersebut langkah selanjutnya adalah menentukan *hidden layer* dan *neuron size* sehingga terbentuk struktur algoritma yang ideal untuk pemecahan masalah tersebut.

IV. Pembahasan

4.1 Metode Neural Network

Algoritma *neural network* adalah algoritma untuk pelatihan *supervised* dan didesain untuk operasi pada *feed forward* multilapis. Algoritma *neural network* bisa didekripsikan sebagai berikut: ketika jaringan diberikan pola masukan sebagai pola pelatihan maka pola tersebut menuju ke unit-unit pada lapisan tersembunyi untuk diteruskan ke unit-unit lapisan terluar. Hasil terbaik pada *eksperiment* adalah dengan *accuracy* yang dihasilkan sebesar 98.50 dan AUCnya 0.982.

Dari eksperimen terbaik di atas maka didapat *arsitektur neural network* dengan menghasilkan enam *hidden layer* dengan tujuh atribut *input layer* dan dua *output layer*. Gambar *arsitektur neural network* terlihat pada gambar 3 seperti di bawah ini



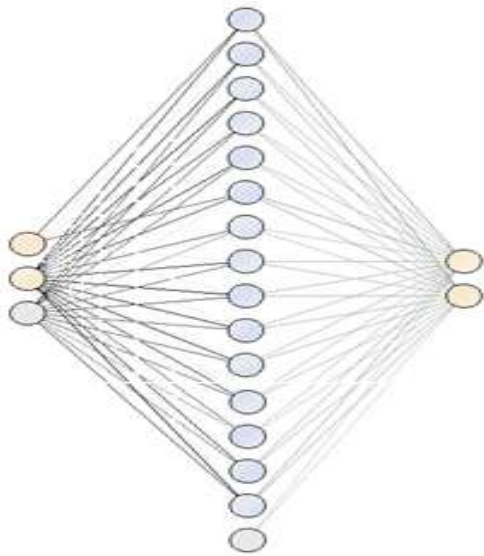
Gambar 3. *Arsitektur neural network*

4.2 Neural Network berbasis PSO

Particle Swarm Optimization memiliki perbandingan lebih atau bahkan pencarian kinerja lebih unggul untuk banyak masalah optimasi dengan lebih cepat dan tingkat *konvergensi* yang lebih stabil. Untuk menemukan solusi yang optimal, masing-masing partikel bergerak ke arah posisi yang terbaik sebelumnya dan posisi terbaik secara global. Hasil terbaik pada *eksperiment* diatas adalah dengan *accuracy* yang dihasilkan sebesar 98.85 dan AUCnya 0.996.

Langkah selanjutnya adalah menyeleksi atribut yang digunakan yaitu jenis kelamin, no. urut parpol, suara sah partai, jumlah perolehan kursi, daerah pemilihan, no. urut caleg, suara sah caleg dan 1 atribut sebagai label yaitu hasil pemilu. Dari hasil eksperimen dengan menggunakan algoritma *neural network* berbasis *particle swarm optimization* diperoleh beberapa atribut atribut yang berpengaruh terhadap bobot atribut yaitu : Juml. Perolehan kursi dengan bobot 0.143, no. urut caleg dengan bobot 0.344 dan suara sah caleg dengan bobot 1. Sedangkan atribut lainnya seperti: jenis kelamin, nomor urut partai, suara sah partai, daerah pemilihan dan suara sah caleg tidak berpengaruh terhadap bobot atribut.

Dari eksperimen terbaik di atas maka didapat *arsitektur neural network* dengan menghasilkan lima belas *hidden layer* dengan dua atribut *input layer* dan dua *output layer*. Gambar *arsitektur neural network* terlihat pada gambar 4 dibawah ini

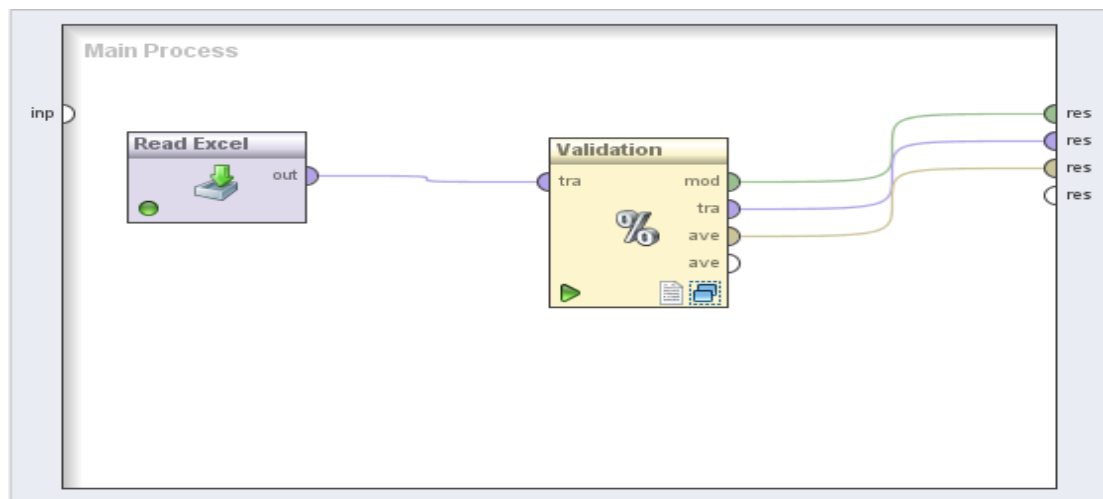


Gambar 4. Arsitektur *neural network* berbasis PSO

4.3 Analisa Evaluasi dan Validasi Model

Hasil dari pengujian model yang dilakukan adalah memprediksi hasil pemilu legislatif DKI Jakarta 2009 dengan *neural network* dan *neural network* berbasis *particle swarm optimization* untuk menentukan nilai *accuracy* dan *AUC*.

Dalam menentukan nilai tingkat keakurasian dalam model *neural network* dan algoritma *neural network* berbasis *particle swarm optimization*. Metode pengujianya menggunakan *cross validation* dengan desain modelnya sebagai berikut.



Gambar 5. Pengujian cross validation

Dari hasil pengujian diatas, baik evaluasi menggunakan *confusion matrix* maupun *ROC curve* terbukti bahwa hasil pengujian algoritma *neural network* berbasis PSO memiliki nilai akurasi yang lebih tinggi dibandingkan dengan algoritma *neural network*.

Nilai akurasi untuk model algoritma *neural network* sebesar 98.50 % dan nilai akurasi untuk model algoritma *neural network* berbasis PSO sebesar 98.85 % dengan selisih akurasi 0.35 %.

Sedangkan evaluasi menggunakan *ROC curve* sehingga menghasilkan nilai *AUC* (*Area Under Curve*) untuk model algoritma *neural network* mengasilkan nilai 0.982 dengan nilai diagnosa *Excellent Classification*, sedangkan untuk algoritma *neural network* berbasis PSO (*Particle Swarm Optimization*) menghasilkan nilai 0.994 dengan nilai diagnosa *Excellent*

Classification, dan selisih nilai keduanya sebesar 0.12.

V. Kesimpulan

Berdasarkan hasil eksperimen yang dilakukan dari hasil analisis optimasi model algoritma *neural network* berbasis *particle swarm optimization* memberikan nilai akurasi yang lebih tinggi yaitu 98.85 % dibandingkan dengan model algoritma *neural network* 98.50 %. Dari hasil tersebut didapatkan selisih antara kedua model yaitu 0.35 %. Sementara untuk evaluasi menggunakan *ROC curve* untuk kedua model yaitu, untuk model algoritma *neural network* nilai *AUC* adalah 0.982 dengan tingkat diagnosa *Excellent Classification*, dan untuk model algoritma *neural network* berbasis *particle swarm optimization* nilai *AUC* adalah

0.996 dengan tingkat diagnosa *Excellent Classification*. Dari evaluasi ROC curve tersebut terlihat bahwa model *neural network* berbasis *particle swarm optimization* lebih tinggi jika dibandingkan dengan algoritma *neural network*. Dari hasil tersebut didapatkan selisih antara kedua model yaitu 0.14. Sehingga dapat disimpulkan bahwa algoritma *neural network* berbasis *particle swarm optimization* lebih akurat dalam memprediksi hasil pemilu legislatif.

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.
- Borisyuk, R., Borisyuk, G., Rallings, C., & Thrasher, M. , 2005, Forecasting the 2005 General Election: A Neural Network Approach. *The British Journal of Politics & International Relations Volume 7, Issue 2* , 145-299.
- Choi, J. H., & Han, S. T. , 1999, Prediction of Elections Result using Discrimination of Non-Respondents: The Case of the 1997 Korea Presidential Election.
- Dawson, C. W. , 2009, *Projects in Computing and Information System A Student's Guide*. England: Addison-Wesley.
- Gill, G. S. , 2005, Election Result Forecasting Using two layer Perceptron Network. *Journal of Theoretical and Applied Information Technology Volume.4 No.11* , 144-146.
- Gorunescu, F. , 2011, *Data Mining Concept Model Technique*. India: Springer.
- Gray, D. E. , 2004, *Doing Research in the Real World*. New Delhi: SAGE.
- Han, J., & Kamber, M. , 2007, *Data Mining Concepts and Technique*. Morgan Kaufmann publisher.
- 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.
- Kothari, C. R. , 2004, *Research Methodology methodes and Technique*. India: New Age Interntional.
- 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.
- Maimon, O., & Rokach, L. , 2010, *Data Mining and Knowledge Discovery Handbook*. London: Springer.
- Moscato, P., Mathieson, L., Mendes, A., & Berreta, R. , 2005, The Electronic Primaries: Prediction The U.S. Presidential Using Feature Selection with safe data. *ACSC '05 Proceeding of the twenty-eighth Australian conference on Computer Science Volume 38* , 371-379.
- 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.
- Rigdon, S. E., Jacobson, S. H., Sewell, E. C., & Rigdon, C. J. , 2009, A Bayesian Prediction Model For the United State Presidential Election. *American Politics Research volume.37* , 700-724.
- Salappa, A., Doumpos, M., & Zopounidis, C. , 2007, Feature Selection Algorithms in

- Classification Problems: An Experimental Evaluation. *Systems Analysis, Optimization and Data Mining in Biomedicine* , 199-212.
- Santoso, T. , 2004, Pelanggaran pemilu 2004 dan penanganannya. *Jurnal demokrasi dan Ham* , 9-29.
- Sardini, N. H. , 2011, *Restorasi penyelenggaraan pemilu di Indonesia*. Yogyakarta: Fajar Media Press.
- 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.
- Undang-Undang RI No.10 , 2008.
- 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.
- Mohammad Badrul, M.Kom adalah Staff Pengajar di STMIK Nusa Mandiri Jakarta. Penulis menyelesaikan Study Strata 1 (S1) di Kampus STMIK Nusa Mandiri dengan Jurusan Sistem Informasi dengan gelar S.Kom dan menyelesaikan program Srata 2 (S2) di Kampus yang sama dengan jurusan ilmu Komputer dengan gelar M.Kom. Selain mengajar, Penulis juga aktif dalam membimbing mahasiswa yang sedang melakukan penelitian khususnya di tingkat strata 1 dan penulis juga terlibat dalam tim konsorsium di Jurusan Teknik Informatika STMIK Nusa Mandiri untuk penyusunan bahan ajar. Penulis tertarik dalam bidang keilmuan Data mining, Jaringan , Operating system khususnya open source , Database, Software engineering dan Research Metode.