

PERAMALAN BEBAN PUNCAK MENGGUNAKAN METODE *FEED FORWARD BACKPROPAGATION* DAN *GENERALIZED REGRESSION NEURAL NETWORK*

Yunus Alam Suryatna

S1 Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya
yunussuryatna@mhs.unesa.ac.id

Nur Kholis, Imam Agung, Widi Aribowo

S1 Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya
nurkholis@unesa.ac.id. ahmadimam@unesa.ac.id. widiaribowo@unesa.ac.id

Abstrak

Hal yang berpengaruh pada perencanaan operasi sistem tenaga listrik ialah prakiraan beban yang terjadi pada sistem tenaga listrik untuk meramalkan beban puncak pada hari esok agar tidak terjadi pemborosan daya pada suatu industry pembangkit listrik. Penelitian ini membahas beban puncak listrik dengan perbandingan metode prakiraan. Metode prakiraan yang digunakan adalah FFBNN fungsi pelatihan yaitu train CGF, train CGP, train GDM dan GRNN. Prakiraan akan kebutuhan untuk konsumsi energi listrik dilakukan berdasarkan perhitungan beban puncak pada waktu 16.00-23.00 pada tanggal 7-15 Maret 2016. Memanfaatkan *temperature* sebagai indikator pada prakiraan beban puncak. Hasil prakiraan beban puncak yang mendekati besar nilai beban *real* dengan hasil yang terbaik saat pukul 18.00-18.59 dimana beban *real*-nya menunjukkan nilai 52,2MW dengan prakiraan menggunakan metode FFBNN fungsi GDM diperoleh nilai sebesar 49,7MW memiliki selisih 2,5MW atau dengan persentase kesalahan 04,84%. Sedangkan untuk metode GRNN ditunjukkan pukul 16.00-16.59 dimana nilai beban *real* sebesar 49,3MW dan hasil prakiraan sebesar 47,3MW memiliki selisih 2 MW dengan persentase kesalahan 4,02%.

Kata Kunci: Prakiraan beban puncak, *Feed Forward Backpropagation*, *Generalized Regression*, *Neural Network*.

Abstract

The thing that affects the planning of the operation of the electric power system is the prediction of the load that occurs in the electric power system to predict the peak load on the next day so that there is no waste of power in a power generation industry. This study discusses the peak electrical load by comparing the forecast method used is the FFBNN training function, namely the CGF train, CGP train, GDM train and GRNN. The forecast of the need for electrical energy consumption is carried out based on the calculation of the peak load at 16.00-23.00 on march 7-15 2016. Using temperature as an indicator in the peak load forecast. The peak load forecast result are close to the real load value with the best result at 18.00-18.59 where the real load shows a value of 52.2MW with forecasts using the FFBNN method the GDM function obtained a value of 49.7MW having a difference of 2.5MW or a percentage error 04.84%. Meanwhile, the GRNN method is shown at 16.00-16.59 where the real load value is 49.3MW and the forecast result is 47.3MW which has a difference of 2MW with an error percentahe of 4.02%.

Keywords: Peak Load Forecasting, Feed Forward Backpropagation, Generalized Regression, NeuralNetwork.

PENDAHULUAN

Ketersediaan energi listrik saat dibutuhkan menyebabkan daya listrik bersifat fluktuatif seiring dengan berjalannya waktu. Disertai dengan permintaan konsumen, kualitas yang mumpuni dan harga yang tergolong murah. Bilamana daya yang dibangkitkan oleh sebuah pembangkit terlalu tinggi, maka akan berdampak munculnya pemborosan energi berlebih yang mampu mengakibatkan kerugian yang besar di sisi perusahaan listrik. Sedangkan apabila daya yang dibangkitkan lebih kecil dari permintaan konsumen, maka dapat menyebabkan terjadinya pemadaman tenaga listrik secara bergiliran yang sudah jelas merugikan konsumen.

Oleh sebab itu salah satu faktor yang menentukan pembuatan rancangan operasi sistem tenaga listrik aalah melakukan prakiraan beban puncak yang akan terjadi di sistem tenaga listrik tersebut. Sebuah permasalahan unik dalam pengooperasian sistem yakni: “Daya yang dibangkitkan / diproduksi harus selalu sama dengan daya yang dikonsumsi oleh pemakai tenaga listrik yang biasanya dikatakan sebagai beban sistem” (M.Djiteng, 1990). Pada era sekarang, di berbagai negara maju, metode di atas hampir jarang sekali diterapkan. Adapun metode yang diterapkan untuk melakukan prakiraan beban listrik sudah dirancang menggunakan teknologi analisis yang telah komputasi. keunggulan utama dari JST (Jaringan Syaraf Tiruan) ialah kemampuan mengomputasi secara paralel

berdasarkan dari berbagai pola yang diajarkan. Pada proses pembelajarannya, JST mampu mengaplikasikan *non-linier regretion* dari berbagai pola beban listrik untuk setiap jamnya pada satu hari. Sehingga membuat JST bisa untuk memprakirakan beban listrik pada keesokan harinya atau hari yang akan datang (Triwulan et al., 2013).

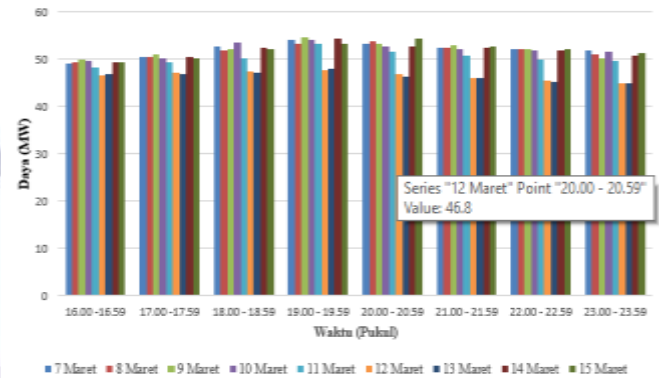
Prakiraan yang terdapat pada sistem tenaga listrik biasanya berupa prakiraan beban atau sering disebut *load forecasting* yang terdiri atas prakiraan beban puncak (MW) serta *demand forecasting* atau prakiraan permintaan energi listrik (MWh). Sedangkan prakiraan yang berdasarkan pada satuan periode waktu dapat dikelompokkan menjadi prakiraan jangka pendek, prakiraan jangka menengah dan prakiraan jangka panjang (Arya Hamidie, 2009).

Metode *Artificial Neural Network* (ANN) merupakan suatu metode yang mampu menyelesaikan hubungan non-linier antara faktor ekonomi yang beragam dengan beban, serta mampu melakukan penyesuaian terhadap perubahan yang timbul. Dengan metode *back propagation*, ANN mempunyai kemampuan untuk melakukan prakiraan dengan baik. Sampai saat ini telah banyak peneliti yang mempergunakan ANN guna melakukan prakiraan beban listrik jangka pendek dan luaran yang dihasilkan cukup baik. Hal ini dikarenakan prakiraan jangka pendek hanya terpengaruhi faktor cuaca (Muslimin, 2015).

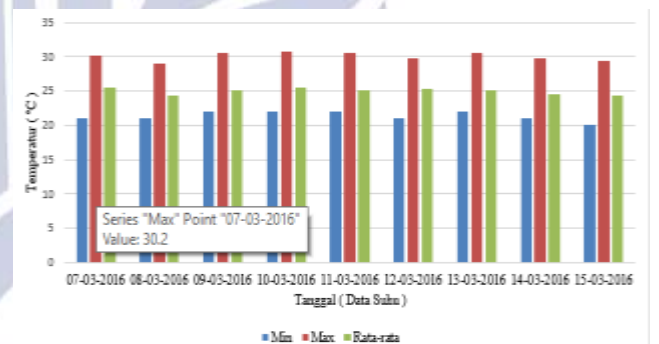
Salah satu diantara model dari jaringan radial basis yang banyak digunakan guna melakukan pendekatan suatu fungsi yakni dengan metode *General Regression Neural Network* (GRNN). Secara esensi operasi GRNN didasarkan pada kernel atau regresi *non-linear* dimana estimasi nilai luaran yang diharapkan telah ditentukan oleh inputannya. Meskipun luaran yang dihasilkan GRNN berupa *multivariate vector*, dengan tidak mengurangi keumuman dari deskripsi logika operasi GRNN. (Caraka et al., 2014)

Hasil dari penelitian ini meramalkan beban puncak harian dengan metode *Feed Forward Backpropagation Neuralnetwork* dengan membandingkan hasil dari berbagai fungsi pelatihan yaitu fungsi train CGF, train CGP, train GDM dan metode GRNN. Untuk data yang diaplikasikan ialah data beban puncak harian yang nantinya dianalisa kemudian mengomparasikan hasil dari perhitungan prakiraan dengan menghitung statistik dengan koefisien beban serta hasil ramalan menggunakan aplikasi Program *Matlab*, data yang dihasilkan disajikan dalam satuan MegaWatt (MW). Dan waktu dimana terjadi beban pada pukul 16:00 s/d 23:00. Untuk data beban listrik didasarkan pada referensi dari PT. PLN Rayon Kota Batu, sedangkan untuk data temperatur udara Kota Batu dari BMKG (Badan Meteorologi, Klimatologi dan Geofisika) Stasiun Klimatologi Malang. Kedua data diatas ialah data

harian yang tercatat dan terekam pada 07 - 15 Maret 2016 dimana pada artikel ini menggunakan data tanggal 7-8 Maret 2016 dan 10-11 Maret 2016 sebagai masukan latihan. Tanggal 9 Maret 2016 dan 12 Maret 2016 sebagai target uji latihan. Tanggal 13 - 14 Maret 2016 sebagai data masukan uji. Serta tanggal 15 Maret 2016 sebagai target pengujian. (Teknik et al., 2004). Data yang dijelaskan diatas dapat dilihat pada Grafik 1 dan 2.



Gambar 1. Grafik Beban Listrik Harian Tanggal 7 s/d 15 Maret 2016
(Sumber : Teknik et al., 2004)



Gambar 2. Grafik Temperatur Suhu Harian Tanggal 7 s/d 15 Maret 2016
(Sumber : BMKG kota Batu,2016)

KAJIAN PUSTAKA Prakiraan Beban

Prakiraan merupakan dugaan atas suatu fenomena tertentu di waktu yang akan datang. Prakiraan memiliki fungsi sebagai tindakan *preventif* terhadap hal-hal yang tidak diinginkan di waktu mendatang. Dalam melakukan prakiraan, pemilihan metodologi yang tepat dapat mengurangi persentase kesalahan (*error*). Hasil dari prakiraan tersebut dapat digunakan sebagai acuan guna menyiapkan tindakan yang akan dilakukan agar hasil aktual sesuai dengan yang diinginkan. (Indonesia et al., 2012)

Suatu pemodelan prakiraan beban yang tepat akan sangat berpengaruh pada perencanaan serta pengoperasian

sistem tenaga listrik. Prakiraan membantu perusahaan penyedia energi listrik ketika mengambil keputusan. Termasuk keputusan baik ketika pembangkitan listrik, pemutusan beban, maupun ketika pembangunan infrastruktur. Prakiraan beban listrik merupakan sebuah langkah pertama yang diambil dari Rencana Usaha Penyediaan Tenaga Listrik (RUPTL) yang disusun dan dirancang oleh PLN pusat. Apabila didasarkan pada jangka waktunya, prakiraan beban listrik diklasifikasikan menjadi tiga jenis. Pengelompokan tersebut antara lain. (Bahtiar, 2014)

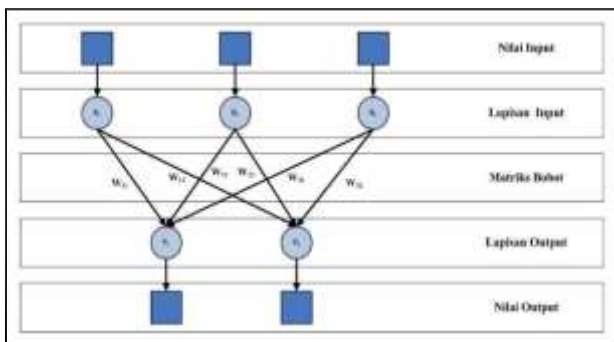
1. Prakiraan jangka panjang, digunakan untuk jangka waktu lebih dari satu tahun.
2. Prakiraan jangka menengah, dilakukan dalam rentang waktu satu hingga duabelas bulan.
3. Prakiraan jangka pendek, digunakan hanya dalam rentang waktu sekian jam dalam satu hari hingga satu minggu (168 jam).

Arsitektur Artificial Neural Network (ANN)

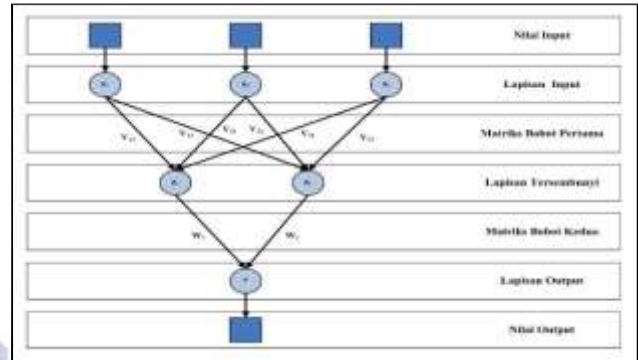
ANN ialah paradigma pemrosesan data dari suatu informasi yang terinspirasi oleh sistem sel syaraf seperti otak yang memproses suatu informasi. Elemen dasar dari paradigma tersebut yakni pembaruan struktur dari sistem pemrosesan informasi. ANN dilakukan guna memecahkan suatu masalah tertentu semacam pengenalan pola karena adanya proses pembelajaran. ANN berkembang pesat dalam satu dekade terakhir dan sudah dikembangkan sebelum adanya komputer konvensional yang modern, canggih dan tetap terus berkembang meskipun pernah mengalami masa suram selama beberapa tahun.

Arsitektur ANN dapat dilihat berdasarkan jumlah *node* atau lapisan dan jumlah *layer* yang ada di masing-masing lapisan. Lapisan penyusun ANN dapat diklasifikasikan menjadi 3, yaitu:

- (1) Lapisan input, yang menerima masukan dari luar sebagai pembelajaran dari problema.
- (2) Lapisan tersembunyi, yang mana *node* ini disebut unit tersembunyi yang tidak bisa diamati secara langsung
- (3) Lapisan output, dimana *node* merupakan hasil luaran JST terhadap suatu permasalahan.



Gambar 3. Arsitektur Artificial Neural Network (ANN) Dengan Lapisan Tunggal (*Single Layer Net*)
(Sumber: Muslimin, 2015)



Gambar 4. Arsitektur Artificial Neural Network (ANN) Dengan Lapisan Banyak (*Multi Layer Net*)
(Sumber: Muslimin, 2015)

Feed Forward Backpropagation Neural Network (FFBNN)

Backpropagation adalah satu dari sekian jenis algoritma yang lumrah diterapkan untuk menyelesaikan suatu permasalahan rumit. Hal tersebut didukung oleh karena jaringan yang menggunakan algoritma dengan jenis ini dilatih dengan menggunakan metode *training* yang terbimbing. Algoritma berikut terbagi atas propagasi mundur serta propagasi maju.

Propagasi maju dimulai dengan memasukkan pola *input* ke dalam lapisan *input*. Pola tersebut adalah nilai dari aktivasi berbagai unit masukan. Dengan dilakukannya propagasi maju, maka nilai aktivasi yang terdapat pada tiap unit di *layer* selanjutnya dihitung pula. Pada tiap lapisan, masing-masing unit pengolah akan melakukan proses penjumlahan berbobot yang kemudian menggunakan fungsi sigmoid guna menghitung outputannya.

Propagasi mundur yang dilakukan ialah dengan mencari nilai kesalahan dan bobot yang terdapat di semua *interconnection* diubah. Nilai *error* atau kesalahan dihitung dengan mencakup keseluruhan unit pengolah serta nilai dari bobot pada semua sambunganpun nantinya diubah. Perhitungan dimulai dari lapisan luaran atau *output layer* kemudian mundur hingga lapisan masukan. Hasil luaran propagasi maju kemudian akan dikomparasikan dengan nilai luaran yang diinginkan.

Langkah-langkah algoritma Pelatihan Propagasi Balik adalah sebagai berikut :

- 1) Inisialisasi bobot-bobot.
- 2) Kerjakan langkah-langkah berikut selama kondisi berhenti bernilai FALSE.
- 3) Untuk masing pasangan elemen nantinya akan masuk tahap pembelajaran, kerjakan: Feed-forward:

4) Tiap-tiap unit masukan (x_i , $i=1,2,3,\dots,n$) menerima sinyal masukan x_i (berupa 10 parameter data masukan), dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya (lapisan tersembunyi).

5) Tiap-tiap unit tersembunyi (z_j , $j=1,2,3,\dots,p$) menjumlahkan sinyal masukan terbobot:

$$z_in_j = v_{oj} + \sum_{i=1}^n x_i v_{ij} \quad (1)$$

Hitung sinyal luarannya dengan fungsi aktivasi:

$$z_j = f(z_in_j) \quad (2)$$

dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit output).

6) Tiap-tiap unit keluaran (y_k , $k=1,2,3,\dots,m$) menjumlahkan sinyal masukan terbobot:

$$y_in_k = w_{ok} + \sum_{j=1}^p z_j w_{jk} \quad (3)$$

gunakan fungsi aktivasi untuk menghitung sinyal keluarannya:

$$y_k = f(y_in_k) \quad (4)$$

dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit output).

Propagasi Balik *error* :

7) Tiap-tiap unit keluaran (y_k , $k=1,2,3,\dots,m$) menerima target referensi (berupa data target beban puncak), untuk dihitung informasi error-nya:

$$\delta_k = (t_k - y_k) f'(y_in_k) \quad (5)$$

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai w_{jk}):

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (6)$$

hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai w_{ok}):

$$\Delta w_{ok} = \alpha \delta_k \quad (7)$$

kirimkan δ_k ini ke unit-unit yang ada di lapisan bawahnya.

8) Tiap-tiap unit tersembunyi (z_j , $j=1,2,3,\dots,p$) menjumlahkan delta masukan-nya (dari unit-unit yang berada pada lapisan atasnya):

$$\delta_in_j = \sum_{k=1}^m \delta_k w_{jk} \quad (8)$$

Kemudian dikalikan nilai diatas dengan turunan dari fungsi aktivasinya guna mencari informasi kesalahan:

$$\delta_j = \delta_in_j f'(z_in_j) \quad (9)$$

kemudian menghitung koreksi bobot (yang selanjutnya digunakan untuk memperbaiki v_{ij}):

$$\Delta v_{ij} = \alpha \delta_j x_i \quad (10)$$

hitung pula koreksi bias (yang selanjutnya digunakan untuk memperbaiki v_{oj}):

$$\Delta v_{oj} = \alpha \delta_j \quad (11)$$

9) Tiap unit luaran (y_k , $k=1,2,3,\dots,m$) memperbaiki bias beserta bobotnya ($j=1,2,3,\dots,p$):

$$w_{jk} (baru) = w_{jk} (lama) + \Delta w_{jk} \quad (12)$$

Tiap unit tersembunyi (z_j , $j=1,2,3,\dots,p$) memperbaiki bias beserta bobotnya ($i=0,1,2,3,\dots,n$):

$$v_{ij} (baru) = v_{ij} (lama) + \Delta v_{ij} \quad (13)$$

10) Tes keadaan berhenti. (Teknologi & No, 2005)

Keterangan:

x_i = sinyal masukan pembelajaran.

w_{jk} = bobot koneksi antara sel j ke sel k .

v_{ij} = bobot antara *input layer* dan *hidden layer*

z_in_j = jumlah masukan yang masuk ke neuron j

z_j = sinyal *output* unit tersembunyi.

f_k = fungsi aktivasi

y_k = sinyal *output* pembelajaran.

t_k = sinyal *output* target (referensi)

δ_k = sinyal *error*.

α = konstanta laju pembelajaran

q = iterasi ke- q .

Fungsi Pelatihan (*Train*)

Fungsi *Train* yang diterapkan metodo *Feed Forward Backpropagation Neural Network* pada penelitian berikut untuk memperoleh perbandingan antara lain:

1. Train CGF (*Gredien Conjugate-Metode Fletcher Reeves*) ialah fungsi pelatihan untuk memperbaiki bobot serta nilai bias yang berdasarkan konjugasi gradien *backpropagation* dengan menerapkan pembaruan Fletcher-Reeves.
2. Train CGP (*Gredien Conjugate- Polak-Ribière*) ialah fungsi pelatihan yang bertugas untuk memperbaiki nilai bobot serta nilai bias yang didasarkan pada hasil konjugasi gradien *backpropagation* dengan menggunakan pembaruan *Polak-Ribière*.
3. Train GDM (*Gradien Descent with Momentum*) ialah model pelatihan yang memiliki fungsi untuk memperbaiki nilai bobot serta nilai bias berdasarkan hasil turunan gradien dengan momentum.

General Regression Neural Network (GRNN)

GRNN (*General Regression Neural Network*) adalah satu dari berbagai model jaringan radial basis yang biasa diterapkan guna melakukan pendekatan terhadap suatu fungsi. Secara esensial, *regresi nonlinear* (kernel) merupakan dasar dari operasi GRNN. Dimana perkiraan dari nilai luaran yang diharapkan ditentukan oleh himpunan inputannya.

Apabila Z diberikan ke (x,y) yang diketahui sebagai regresi Z pada (x,y) ialah suatu jalan keluar guna meminimalkan nilai MSE. Bila $f(x,y,Z)$ merupakan suatu fungsi densitas dari probabilitas kontinu bersama.

Fungsi densitas $f(x,y,Z)$ dapat diperkirakan dari data yang ada dengan menerapkan suatu estimator konsisten non-parametrik :

$$f^{\sim}(x,y) = \frac{1}{2\pi^{(p+1)/2}\sigma^{(p+1)}} \frac{1}{n} \sum_{i=1}^n \exp\left[-\frac{(x-x^i)^T(x-x^i)}{2\sigma^2}\right] \exp\left[-\frac{(y-y_i)^2}{2\sigma^2}\right] \quad (14)$$

Dengan memasukkan estimasi probabilitas bersama ke dalam rerata atau *mean* bersyarat didapatkan estimator kernel sebagaimana berikut. (Caraka et al., 2014)

$$Z_m(X,Y) = \frac{\sum_{i=1}^n z_i \exp\left(-\frac{D_i^2}{2\sigma^2}\right)}{\sum_{i=1}^n \exp\left(-\frac{D_i^2}{2\sigma^2}\right)} \quad (15)$$

Keterangan :

$f^{\sim}(x,y)$ = estimasi probabilitas nonparametik

$Z_m(X,Y)$ = estimasi kernel

n = banyaknya pengukuran himpunan data training

D_i^2 = Jarak Metriks

σ = Parameter penghalusan (*bandwidth*)

p = dimensi vektor variabel x

Struktur dan Arsitektur GRNN

GRNN terdiri atas 4 susunan atau lapisan pemrosesan yakni: lapisan pertama disebut dengan *neuron* inputan, lapisan kedua = *neuron* pola, lapisan ketiga dikenal dengan *summation neuron* dan yang terakhir ialah *neuron output*. Masing-masing lapisan mempunyai peranan tersendiri.

Neuron input berfungsi sebagai penerima sinyal masuk atau bisa dikatakan menerima informasi. Kemudian informasi ini diteruskan ke lapisan kedua yaitu *neuron* pola. Pada *neuron* pola, informasi yang diperoleh dari lapisan pertama akan diproses dengan sistematis menggunakan suatu fungsi aktivasi. Hasil luaran dari proses tersebut akan dilanjutkan menuju *neuron summation*. Pada lapisan pemrosesan ini unit-unit pelatihan dibagi menjadi dua bagian guna menyelesaikan dua jenis *summation* yaitu : sumasi aritmatika sederhana dan *weighted summation*. Penjumlahan dari *neuron-neuron summation* akan dilanjutkan ke tahap yaitu lapisan terakhir atau *neuron output*. Pada tahap di lapisan terakhir ini hasil pembagian dari sumasi aritmatika sederhana dan *weighted summation* adalah *output* atau luaran GRNN yang disimbolkan dengan y . (Adnyani & Subanar, 2015)

Persamaan dalam metode ini berbeda dari algoritma *backpropagation*. Metode *Generalized Regression Neural Network* menggunakan fungsi peramalan secara langsung dari data pelatihan. *Generalized Regression Neural Network* adalah metode untuk memperkirakan fungsi probabilitas probabilitas gabungan dari x dan y , diberikan hanya satu set pelatihan. (Aribowo et al., 2020)

Mean Absolute Deviation (MAD)

MAD (*Mean Absolute Deviation*) merupakan metode untuk mengevaluasi prakiraan dengan menggunakan penambahan dari nilai-nilai *error* atau kesalahan yang absolut. MAD mengukur ketepatan prakiraan dengan menghitung rerata kesalahan dugaan. MAD berguna saat mengukur tingkat kesalahan prakiraan dalam suatu unit yang sejenis sebagai deret asli. Nilai MAD bisa diperoleh menggunakan rumus sebagai berikut : (Darmawan et al., 2017)

$$MAD = \frac{1}{N} \sum_{i=1}^N |R_i - F_i| \quad (16)$$

Keterangan :

N = Banyaknya observasi

F_i = Prakiraan

R_i = Data aktual

MAD = Rata rata persen error per unit

Mean Square Error (MSE)

Mean Squared Error (MSE) merupakan salah satu metode untuk mengevaluasi metode prakiraan yang ada. Setiap *error* nantinya akan dikuadratkan. Setelah itu kemudian dijumlahkan dengan banyaknya jumlah observasi yang dilakukan.

Menghitung MSE yaitu dengan cara membandingkan nilai *output* ($y_i(x)$) dan nilai target ($T_i(x)$). Pendekatan ini mengatur agar menghasilkan kesalahan-kesalahan tingkat sedang yang memiliki kemungkinan untuk diubah lebih baik menjadi kesalahan kecil, akan tetapi terkadang menghasilkan suatu perbedaan yang besar. *Mean Squared Error* merupakan rerata dari *error forecast* dikuadratkan, atau bila disajikan dalam bentuk persamaan adalah : (Darmawan et al., 2017)

$$MSE = \frac{1}{n} \sum_{i=1}^n (T_i(x) - y_i(x))^2 \quad (17)$$

Keterangan :

T_i = nilai keluaran yang diharapkan

y_i = nilai keluaran jaringan syaraf

n = jumlah penelitian.

Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) atau persentase kesalahan absolut rata-rata dihitung dengan menentukan kesalahan atau *error* absolut tiap periode, kemudian dibagi dengan nilai observasi pada periode tersebut, dan akhirnya menentukan rerata persentase absolut tersebut.

Pendekatan ini berguna bila variabel prakiraan merupakan salah satu faktor penting pada proses mengevaluasi akurasi prakiraan tersebut.

MAPE memberikan petunjuk mengenai tingkat *error* atau kesalahan prakiraan dikomparasikan dengan nilai aktual dari seri tersebut. MAPE juga mampu digunakan untuk membandingkan tingkat akurasi dari teknik yang sama atau bahkan berbeda sekalipun pada dua seri yang berbeda.

Mean absolute percentage error (MAPE) merupakan cara yang akurat untuk menghitung kesalahan atau *error*, hal ini dikarenakan menyatakan persentase *error* hasil prakiraan terhadap kondisi aktual selama periode waktu tertentu yang berupa informasi terlalu tinggi atau rendahnya persentase. (Ikhtari, 2010)

$$MAPE = \frac{1}{N} \sum_{i=1}^N \frac{|R_i - F_i|}{R_i} \times 100 \tag{18}$$

Keterangan :

N = Banyaknya observasi

F_i = Prakiraan

R_i = Data aktual

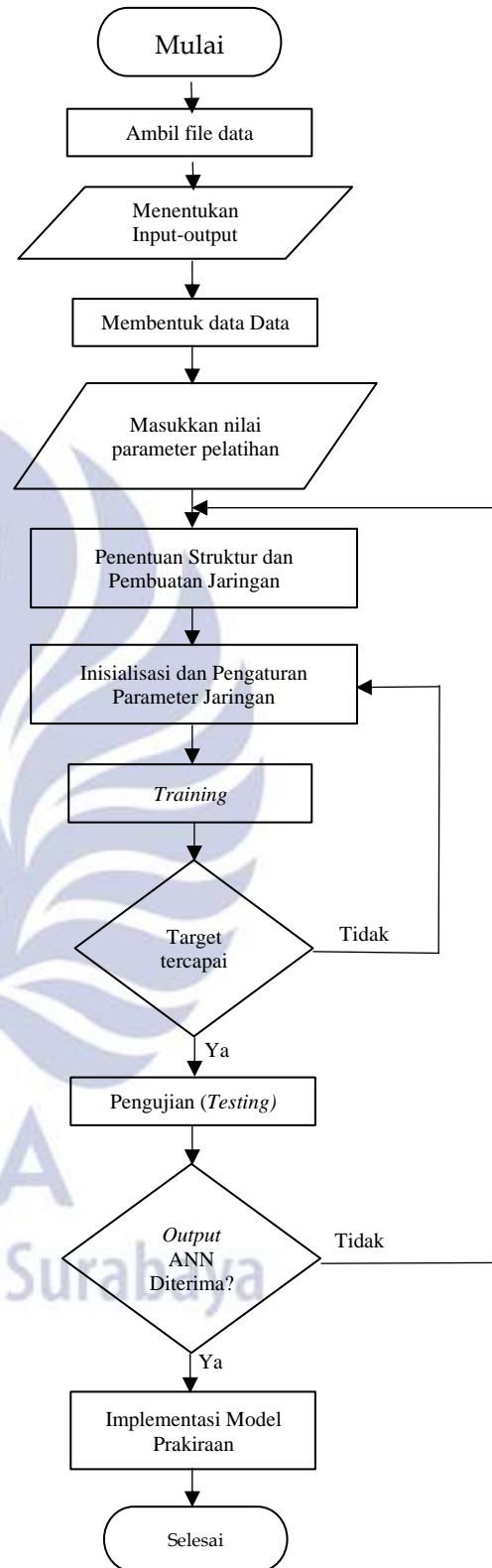
MAPE = Rata rata persen eror keseluruhan

METODE

Prakiraan beban puncak ini menerapkan metode FBNN dengan menggunakan beberapa fungsi dari pelatihannya yaitu CGF, CGP, GDM dan metode GRNN. Data input yang digunakan ialah data beban listrik tiap jam dalam sehari (24 jam) pada tanggal 7 – 15 maret 2016 dan suhu udara dalam satu hari yang mencakup suhu minimum, maksimum dan rata rata.

Tahap pertama dalam penelitian ini adalah melakukan praproses pengambilan data yang akan digunakan untuk melakukan peramalan. kemudian menentukan *input* dan *output* untuk membentuk grafik atau tabel pada data yang akan digunakan sehingga membentuk nilai parameter pelatihan, lalu menentukan struktur dan pembuatan jaringan agar bisa dilakukan pengujian hingga dapat menghasilkan *output* peramalan dari ANN, setelah output data diterima oleh ANN maka akan di implementasikan peramalan dalam bentuk model tabel atau grafik.

Flowchart untuk prakiraan beban puncak ditampilkan pada Gambar 5.



Gambar 5. Diagram Alir Prakiraan Beban

Pre-processing Data

1. Data beban listrik dan temperatur dibagi 2 bagian, 2 pola sebagai data untuk melakukan *training* model ANN, 1 pola data digunakan untuk pengujian model.
2. Fungsi aktivasi yang diterapkan pada jaringan adalah fungsi sigmoid biner yang memiliki nilai dengan rentang antara 0-1, sehingga data *input* perlu dilakukan normalisasi dengan persamaan sebagai berikut:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} (BA - BB) + BB \quad (19)$$

Keterangan :

- X' = Nilai *input* hasil normalisasi
- X = Nilai *input*
- X_{min} = Nilai *input* paling rendah
- X_{max} = Nilai *input* paling tinggi
- BA = Batas atas (1)
- BB = Batas bawah (0)

Penentuan Pola Input dan Pola Output

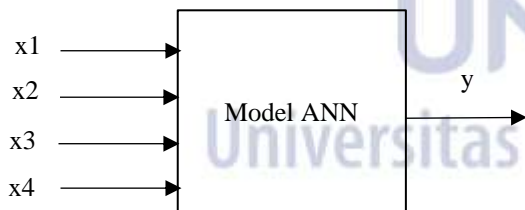
Pola Input

Pola *input* terdiri dari:

1. Data beban puncak harian dan temperatur udara pada tanggal 7-8 Maret, 10-11 Maret sebagai data *input* latih dan 13-14 Maret 2016 sebagai data *input* uji.
2. Data beban listrik per hari dan temperatur udara pada tanggal, 9 Maret dan 12 Maret sebagai data target latih.

Pola Output

Pola *Output* adalah keluaran yang diharapkan atau target uji dari model prakiraan, yaitu berupa hasil prakiraan beban puncak listrik dari jam ke jam pada tanggal 15 Maret 2016. Diagram blok dari model ANN akan ditunjukkan pada Gambar 6.



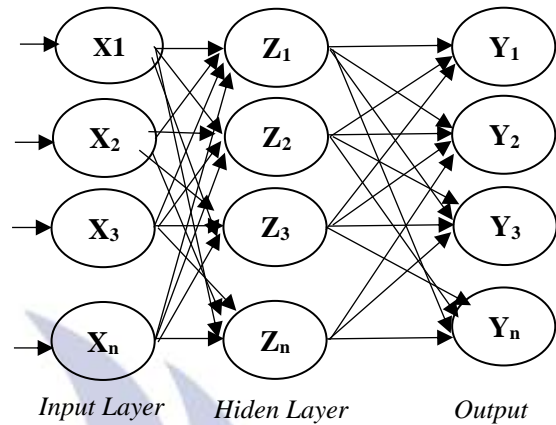
Gambar 6. Diagram Blok Model ANN
(Sumber: Elektro et al., 2015)

Keterangan:

- x1: Beban tanggal 7 Maret, 10 Maret, 13 Maret 2016
- x2: Beban tanggal 8 Maret, 11 Maret, 14 Maret 2016
- x3: Suhu tanggal 7 Maret, 10 Maret, 13 Maret 2016
- x4: Suhu tanggal 8 Maret, 11 Maret, 14 Maret 2016
- y: Beban hasil prediksi tanggal 15 Maret 2016

Rancangan Arsitektur Artificial Neural Network

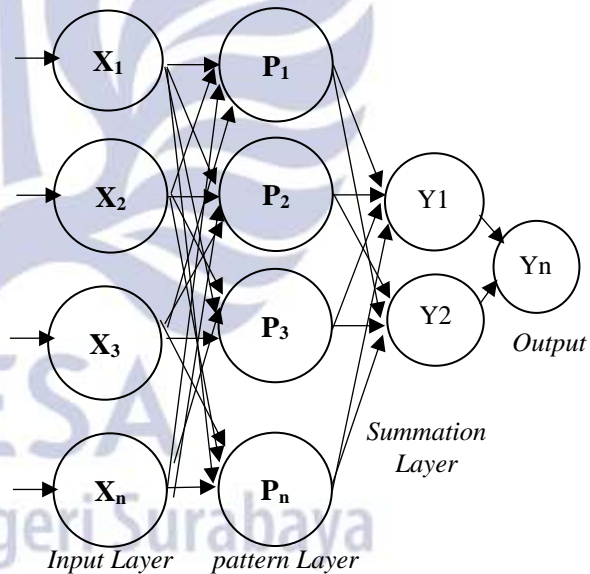
Model arsitektur yang diperbarui dari metode *Feed Forward Backpropagation* pada JST ditunjukkan pada Gambar 7.



Gambar 7. Arsitektur *Feed Forward Backpropagation Neural Network*

(Sumber: Elektro et al., 2015)

Arsitektur yang dikembangkan dari metode GRNN ditunjukkan pada Gambar 8.



Gambar 8. Arsitektur *Generalized Regression Neural Network*

(Sumber: Elektro et al., 2015)

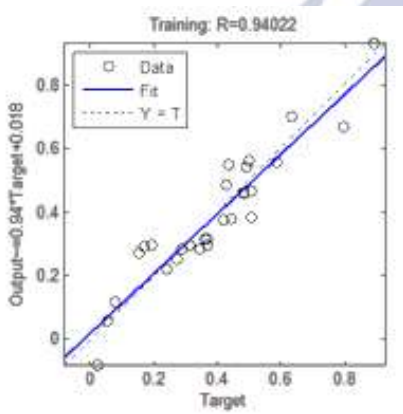
Processing Data

Setelah melewati tahap *preprocessing data*, selanjutnya menghitung tiap neuron pada masing-masing lapisan pada metode *Feed Forward Backpropagation Neuralnetwork* dengan menerapkan fungsi pelatihan CGF, CGP, GDM dan GRNN, yang kemudian dilanjut dengan mencari besar nilai *error*.

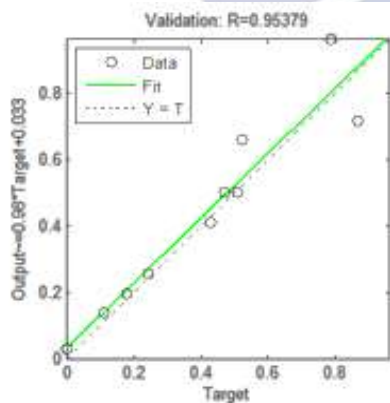
HASIL DAN PEMBAHASAN

Berdasarkan pengolahan data dari bulan Maret, dapat dilihat prakiraan beban puncak listrik tanggal 7 s/d 15 Maret tahun 2016 dengan menerapkan metode *Feed Forward Backpropagation Neuralnetwork* menggunakan fungsi pelatihan CGF, CGP, GDM dan *Generalized Regression Neural Network*.

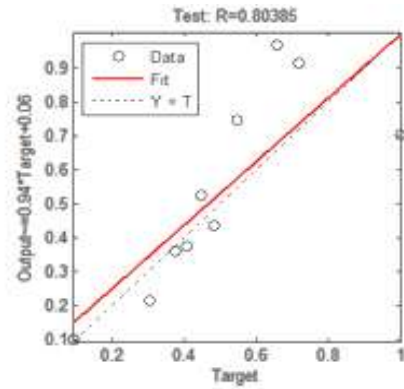
Pada proses pengolahan data metode *FeedForward Backpropagation NeuralNetwork* terdapat proses yang dinamakan *Regression* yang terdiri dari *Training*, *Validation*, *Test* dan hasil keseluruhan dimana hasil yang menunjukkan yang terbaik adalah yang mendekati nilai 1, Berikut adalah proses *Regression* dari metode *FeedForward Backpropagation NeuralNetwork* :



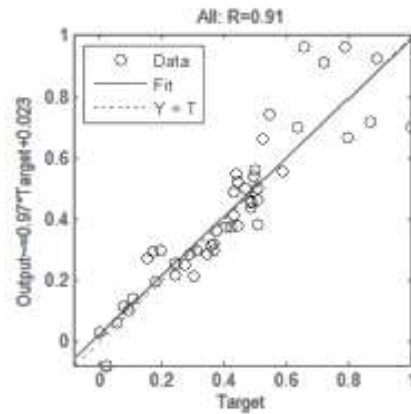
Gambar 9. Training Metode *FeedForward Backpropagation NeuralNetwork* Fungsi CGP.



Gambar 10. Validation Metode *FeedForward Backpropagation NeuralNetwork* Fungsi CGP.



Gambar 11. Test Metode *FeedForward Backpropagation NeuralNetwork* Fungsi CGP.



Gambar 12. Hasil Keseluruhan Metode *FeedForward Backpropagation NeuralNetwork* Fungsi CGP.

Hasil peramalan beban puncak listrik dengan data real pada tanggal 15 Maret 2016 dengan metode *Feed Forward Backpropagation Neuralnetwork* yang menerapkan beberapa fungsi pelatihan CGF, CGP, GDM dan metode *GRNN* ditunjukkan pada tabel 1.

Tabel 1. Hasil Prakiraan Beban Puncak Listrik Dari Data Tanggal 15 Maret 2016

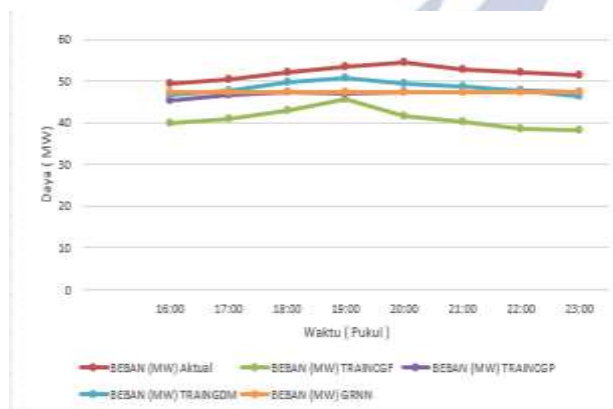
Waktu	Beban Real (MW)	Metode FFBNN			Metode GRNN (MW)
		Fungsi Pelatihan			
		CGF (MW)	CGP (MW)	GDM (MW)	
16:00-16.59	49.3	39.8388	45.4486	46.6145	47.319
17:00-17.59	50.3	40.9988	46.5132	47.7585	47.3262
18:00-18.59	52.2	43.0997	47.4112	49.6758	47.3396
19:00-19.59	53.4	45.6522	47.1035	50.7094	47.353
20:00-20.59	54.4	41.4884	47.3978	49.3098	47.3377
21:00-21.59	52.7	40.3085	47.4355	48.7302	47.3336
22:00-22.59	52.1	38.6405	47.5104	47.5045	47.3261
23:00-23.59	51.4	38.2842	47.4593	46.404	47.3184

Nilai MAPE untuk setiap metode dapat dilihat pada Tabel 2.

Tabel 2. Nilai MAPE setiap metode

Metode		MAD (%)	MAPE (%)
FFBNN	TRAIN CGF	4.45855	21.02798
	TRAIN CGP	2.23761	9.45539
	TRAIN GDM	1.64730	6.97555
GRNN		2.15714	21.02798

Perbandingan antara beban *real* dengan metode *Feed Forward Backpropagation NeuralNetwork* menggunakan beberapa pelatihan serta metode GRNN dapat ditunjukkan pada Gambar 12.



Gambar 12. Grafik Perbandingan Beban *Real*, metode *Feed Forward Backpropagation NeuralNetwork* dengan beberapa fungsi dan metode GRNN (*Generalized Regression NeuralNetwork*)

PENUTUP

Kesimpulan dan Saran

Pembahasan pada penelitian ini adalah prakiraan beban puncak listrik PT. PLN (Persero) Rayon Kota Batu dengan menerapkan metode FFBNN dengan fungsi CGF, CGP, GDM dan metode GRNN, dimana beban prakiraan dengan target uji tanggal 15 Maret 2016. Hasil prakiraan beban puncak yang mendekati besar nilai beban *real* dengan hasil yang terbaik saat pukul 18.00-18.59 dimana beban *real*-nya menunjukkan nilai 52,2MW dengan prakiraan menggunakan metode FFBNN fungsi GDM diperoleh nilai sebesar 49,7MW memiliki selisih 2,5MW atau dengan persentase kesalahan 04,84%. Sedangkan untuk metode GRNN ditunjukkan pukul 16.00-16.59 dimana nilai beban *real* sebesar 49,3 MW dan hasil dari prakiraan sebesar 47,3MW memiliki selisih 2 MW dengan persentase kesalahan 4,02%.

Untuk hasil daripada prakiraan beban puncak yang memiliki selisih kesalahan paling besar ketika pukul 22.00-20.59 pada metode FFBNN fungsi pelatihan CGF yang mana besar nilai pada beban *real* 52,1MW dengan hasil

prakiraan senilai 38,6 MW dengan selisih 13,5MW atau dengan persentase kesalahan 25,83%. Sedangkan pada metode GRNN pada pukul 20.00-20.59 dimana nilai dari beban *real* 54,4MW, dan hasil dari prakiraan 47,3 MW dengan selisih 7,1 MW atau dengan persentase kesalahan 12,98%.

Untuk nilai MAPE pada FFBNN dengan fungsi pelatihan CGF sebesar 21%, CGP sebesar 9,4%, dan GDM sebesar 6,9% sedangkan pada GRNN sebesar 21%. Maka dari hasil nilai MAPE dari metode-metode tersebut dapat disimpulkan bahwa hasil prakiraan terbaik dimiliki oleh metode FFBNN dengan fungsi GDM dimana nilai MAPE-nya mendekati nilai beban *real* yaitu sebesar 6,9%.

Harapan untuk penelitian selanjutnya ialah dapat memperbaiki kekurangan sehingga didapatkan hasil luaran atau output prakiraan beban puncak listrik dengan persentase *error* yang lebih kecil. Oleh sebab itu untuk penelitian selanjutnya disarankan agar melakukan perbandingan dengan membedakan beban puncak antara hari biasa dengan hari libur, serta dengan mencermati parameter lainnya yang berdampak pada tingkat konsumsi beban listrik seperti suhu udara, kegiatan sosial masyarakat pertumbuhan ekonomi, pertumbuhan penduduk atau lain sebagainya.

DAFTAR PUSTAKA

- Adnyani, L. P. W., & Subanar. (2015). General Regression Neural Network (GRNN) Pada Peramalan Kurs Dolar Dan Indeks Harga Saham Gabungan (IHSG). *Factor Exacta*, 8(2), 137–144.
- Aribowo, W., Muslim, S., & Basuki, I. (2020). Generalized Regression Neural Network for Long-Term Electricity Load Forecasting. *Proceeding - ICoSTA 2020: 2020 International Conference on Smart Technology and Applications: Empowering Industrial IoT by Implementing Green Technology for Sustainable Development*, April. <https://doi.org/10.1109/ICoSTA48221.2020.1570611361>
- Arya Hamidie, K. (2009). *Metode koefisien energi untuk peramalan beban jangka pendek pada jaringan jawa madura bali*. 1–10.
- Bahtiar, S. M. (2014). Peramalan Beban Dengan Menggunakan Metode Time Series untuk Kebutuhan Tenaga Listrik Di Gardu Induk Sungai Raya. *Universitas Tanjungpura*.
- BMKG, 2016, "Data Iklim Harian Temperatur Suhu" BMKG Batu, http://dataonline.bmkg.go.id/data_iklim,1Juli2020pukul09.58.
- Caraka, R. E., Yasin, H., & Prahutama, A. (2014). Pemodelan General Regression Neural Network (Grnn) Dengan Pemodelan General Regression Neural Network (Grnn) Dengan Peubah Input Data Return Untuk. *Trusted Digital Identity and Intelligent System*, June 2015, 283–288. https://www.researchgate.net/publication/277712485_Pemodelan_General_Regression_Neural_Networ

k_GRNN_Dengan_Peubah_Input_Data_Return_untuk_Peramalan_Indeks_Hangseng?ev=prf_pub

- Darmawan, D. R., Aspiranti, T., & Koesdiningsih, N. (2017). Analisis Peramalan Penjualan dengan Menggunakan Metode Single Moving Average , Weighted Moving Average dan Exponential Smoothing Sebagai Dasar Perencanaan Produksi Polo Shirt Pria (Studi Kasus pada PT . Amanah Garment Bandung). *Prosiding Manajemen*, 3(2), 703–708.
- Elektro, T., Teknik, F., Surabaya, U. N., & Aribowo, W. (2015). *PERAMALAN BEBAN LISTRIK JANGKA PENDEK MENGGUNAKAN METODE FEED FORWARD BACKPROPAGATION NEURAL NETWORK* Muhammad Permana Setya Gunawan Abstrak.
- Ikhtari, H. (2010). *DENGAN MENGGUNAKAN METODE ADAPTIVE NEURO FUZZY INFERENCE SYSTEM (ANFIS)* Oleh : Ikhtari Haimi.
- Indonesia, U., Dwiantoro, B., Teknik, F., Studi, P., & Elektro, T. (2012). *Berdasarkan Data Historis Menggunakan Metode Generalized Autoregressive Conditional Heteroskedasticity (Garch) Metode Generalized Autoregressive Conditional*.
- M. Djiteng. (1990). *Operasi Sistem Tenaga Listrik*, Balai Penerbit dan Humas ISTN, Jakarta, hal 13.
- Teknik, J., Fakultas, S., & Maret, U. S. (2004). *Lembar persetujuan*. 1–12.
- Teknologi, J., & No, E. (2005). Aplikasi Jaringan Syaraf Tiruan Untuk Peramalan Beban Tenaga Listrik Jangka Panjang Pada Sistem Kelistrikan Di Indonesia. *Tahun XIX*, 3, 211–217. http://seminar1.te.ugm.ac.id/pdf/05920_Paper3.pdf
- Triwulan, Y., Hariyanto, N., & Anwari, S. (2013). Peramalan Beban Puncak Listrik Jangka Pendek Menggunakan Metode Jaringan Syaraf Tiruan. *Jurnal Reka Elkomika ©TeknikElektro | Itenas Jurnal Online Institut Teknologi Nasional Jurnal Reka Elkomika*, 1(4), 2337–2439.