

Peramalan Beban Listrik Jangka Pendek Menggunakan *Optimally Pruned Extreme Learning Machine* (OPELM) pada Sistem Kelistrikan Jawa Timur

Januar Adi Perdana¹⁾, Adi Soeprijanto²⁾ dan Rony Seto Wibowo³⁾

Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111

E-mail: adisup@ee.its.ac.id²⁾

Abstrak—Peramalan beban listrik jangka pendek merupakan faktor yang sangat penting dalam perencanaan dan pengoperasian sistem tenaga listrik. Tujuan dari peramalan beban listrik adalah agar permintaan listrik dan penyediaan listrik dapat seimbang. Karakteristik beban di wilayah Jawa Timur sangat fluktuatif sehingga pada penelitian ini digunakan metode *Optimally Pruned Extreme Learning Machine* (OPELM) untuk meramalkan beban listrik. Kelebihan OPELM ada pada *learning speed* yang cepat dan pemilihan model yang tepat meskipun datanya mempunyai pola non linier. Keakuratan metode OPELM dapat diketahui dengan menggunakan metode perbandingan yaitu metode ELM. Kriteria keakuratan yang digunakan adalah MAPE. Hasil dari perbandingan kriteria keakuratan menunjukkan bahwa hasil peramalan OPELM lebih baik dari ELM. Error rata-rata hasil pengujian peramalan paling minimum menunjukkan MAPE sebesar 1,3579% terjadi pada peramalan hari Jumat, sementara pada hari yang sama dengan metode ELM menghasilkan MAPE sebesar 2,2179%.

Kata kunci: Peramalan Beban Listrik Jangka Pendek, OPELM, ELM

I. PENDAHULUAN

Kebutuhan listrik di berbagai daerah dari waktu ke waktu selalu berbeda bergantung pada pemakaian listrik di daerah tersebut, sehingga penyediaan tenaga listrik dan alokasi pembangkit yang digunakan juga berbeda di daerah yang satu dengan lainnya. Pengalokasian pembangkit yang digunakan harus tepat agar kebutuhan listrik dari konsumen terpenuhi[1]. Kebutuhan listrik tersebut digunakan hampir di semua sektor, antara lain sektor rumah tangga, industri, usaha komersial, dan tempat layanan umum. Besarnya permintaan listrik pada suatu rentang waktu tidak dapat dihitung secara pasti. Akibatnya timbul permasalahan, yaitu bagaimana mengoperasikan suatu pembangkit sistem tenaga listrik secara kontinyu agar dapat memenuhi permintaan daya setiap saat. Apabila daya yang dikirim dari pembangkit jauh lebih besar daripada permintaan daya pada beban, maka akan timbul pemborosan biaya pembangkitan energi listrik pada perusahaan listrik. Sedangkan apabila daya yang dibangkitkan dan dikirimkan lebih rendah atau bahkan tidak memenuhi kebutuhan konsumen maka akan terjadi pemadaman lokal pada beban dan merugikan pihak konsumen. Dengan demikian diperlukan suatu usaha untuk memprediksi permintaan beban oleh konsumen melalui proses peramalan beban listrik yang pada akhirnya akan mempunyai peranan penting dalam hal ekonomi

dan keamanan operasi sistem tenaga[2]. Peramalan beban jangka pendek bertujuan untuk meramalkan beban listrik pada jangka waktu menit, jam, hari atau minggu.

Perusahaan penyedia listrik menggunakan suatu metode konvensional untuk meramalkan kebutuhan beban listrik di masa yang akan datang, dimana model dirancang berdasarkan hubungan antara beban listrik dengan faktor-faktor non linier yang mempengaruhi konsumsi beban seperti tingkat pertumbuhan penduduk, tingkat ekonomi masyarakat, cuaca pada periode tertentu, biaya pembangkitan energi listrik dan lain sebagainya. Dalam penelitian ini, diperkenalkan suatu metode *Optimally Pruned Extreme Learning Machine* (OPELM) sebagai solusi untuk permasalahan dari faktor-faktor non linier di atas yaitu peramalan beban listrik dilakukan berdasarkan pengenalan pola beban dalam periode jangka pendek pada semua hari di sistem kelistrikan di Jawa Timur.

II. PERAMALAN BEBAN LISTRIK

A. Pengenalan Pola Beban Listrik

Peramalan di bidang tenaga listrik dimaksudkan pada perkiraan kebutuhan beban listrik di masa yang akan datang. Beban yang diramalkan mempunyai jangka waktu tertentu yang disesuaikan dengan kebutuhan peramalan. Peramalan beban jangka pendek untuk meramalkan kebutuhan beban harian umumnya tersaji pada data permintaan beban harian per 30 menit selama 24 jam tersebut.

Peramalan beban listrik bertujuan untuk mengenali pola beban dengan mengolah data historis beban listrik yang direpresentasikan dalam kurva beban harian. Suatu model peramalan beban yang akurat sangat penting dalam perencanaan dan pengoperasian sistem tenaga listrik. Pemodelan yang sesuai berdasarkan pada pengenalan pola beban yang dilakukan sehingga diperoleh parameter-parameter yang diperlukan. Dalam penelitian ini digunakan pemodelan OPELM (*Optimally Pruned Extreme Learning Machine*) yang diaplikasikan dalam proses pelatihan untuk melatih weight dan bias.

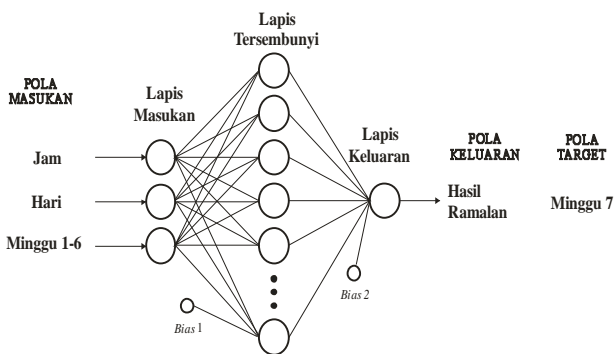
B. Pengumpulan dan Pengolahan Data

Data yang digunakan adalah data historis beban listrik harian per 30 menit selama 24 jam dalam satuan Megawatt (MW) pada sistem kelistrikan Jawa Timur pada 2 Januari 2012

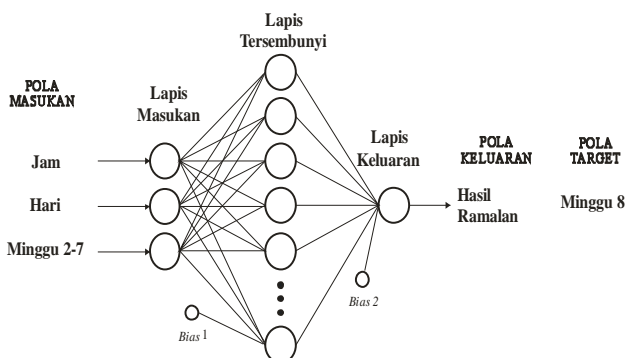
hingga 26 Februari 2012. Data diambil dari PT PLN (Persero) Area Pengatur Beban (APB) Jawa Timur.

Data yang digunakan sebagai masukan adalah data beban, hari, dan jam. Hari dan jam diubah terlebih dahulu ke bentuk desimal sebelum digunakan sebagai masukan.

Senin diinisialisasi menjadi 10, Selasa menjadi 20, seterusnya hingga kelipatan 10 (Minggu menjadi 70). Sedangkan jam diubah menjadi 1 hingga 48 karena data diambil per 30 menit selama 24 jam. Data beban dibagi menjadi dua yaitu data untuk training dan data untuk testing. Data pada awalnya melalui proses training mengenali pola beban, kemudian disimulasikan dengan proses testing untuk mengetahui keakuratan peramalan beban di masa yang akan datang. Pemodelan jaringan *feedforward* ANN untuk proses training maupun testing menjadi sebagai berikut.



Gambar 1 Arsitektur ANN untuk Proses Training



Gambar 2 Arsitektur ANN untuk Proses Testing

Berikut perumusan data masukan dan keluaran berdasarkan pemodelan di atas.

Data untuk proses *training* pengenalan pola beban adalah :
 $y_{hasil_training\ M-7} = f(y_{M-6}, y_{M-5}, y_{M-4}, y_{M-3}, y_{M-2}, y_{M-1}, y_{hari}, y_{jam})$ (1)

Data untuk proses *testing* peramalan beban adalah :
 $y_{hasil_ramal\ M-8} = f(y_{M-7}, y_{M-6}, y_{M-5}, y_{M-4}, y_{M-3}, y_{M-2}, y_{hari}, y_{jam})$ (2)

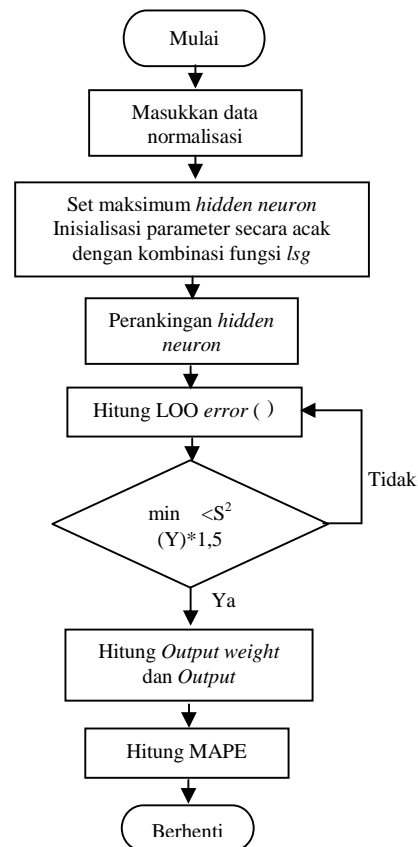
Peramalan ini berdasarkan data *similar day*, dimana pola beban hari Senin pada minggu ini akan mempunyai pola yang sama dengan hari Senin pada minggu yang akan datang.

III. OPTIMALLY PRUNED EXTREME LEARNING MACHINE (OPELM)

Optimally Pruned Extreme Learning Machine (OPELM) merupakan sebuah metode yang berdasarkan pada ELM

aslinya. OPELM ditujukan untuk mengatasi kelemahan yang ada pada ELM ketika terdapat variabel yang tidak relevan atau tidak berkorelasi. Untuk alasan tersebut, diperkenalkan metode OPELM untuk pemangkasan variabel yang tidak relevan dengan memangkas neuron tidak penting dari SLFN yang dibangun oleh ELM. Model OPELM dibangun dalam tiga tahap[3]. Langkah pertama dari metode OPELM adalah membangun struktur SLFN menggunakan algoritma ELM. Kemudian dilakukan perankingan neuron pada lapisan tersembunyi dengan algoritma MRSR (*Multiresponse Sparse Regression*), dan akhirnya penentu banyaknya neuron yang dipangkas dibuat berdasarkan metode estimasi *error Leave-One-Out* (LOO). Algoritma OPELM menggunakan kombinasi tiga jenis kernel, linear, sigmoid, dan gaussian. Sedangkan pada ELM hanya digunakan satu kernel saja, misalnya sigmoid.

Prosedur untuk tahapan *training* algoritma OPELM dapat digambarkan pada *flowchart* pada gambar 3.



Gambar 3 *flowchart* metode OPELM

Data diolah dan dikelompokkan berdasarkan pola *similar day* untuk masing-masing hari-hari, kemudian dilakukan normalisasi tiap data dengan persamaan berikut.

$$X_n = 2 [X_p - \min(X_p)] / [\max(X_p) - \min(X_p)] - 1 \quad (3)$$

Dimana :

- X_n = nilai hasil normalisasi dengan range [-1,1]
- X_p = nilai data asli yang belum dinormalisasi
- $\min(X_p)$ = nilai minimum pada data set
- $\max(X_p)$ = nilai maximum pada data set

Parameter yang digunakan pada penelitian ini antara lain.

- jumlah *hidden neuron* ditentukan pada maksimumnya sebanyak 25, mengingat OPELM tidak dilakukan *trial and error*
- Fungsi aktivasi menggunakan kombinasi fungsi *linear*, *sigmoid*, dan *gaussian*.

1) Pembentukan SLFNs dengan ELM

Langkah ini dilakukan menggunakan algoritma standar ELM dengan sejumlah neuron yang cukup besar N . Sementara ELM aslinya menggunakan kernel sigmoid dan model ELM ini biasanya dirumuskan berdasarkan satu jenis fungsi aktivasi atau kernel saja. Namun dalam metode OPELM digunakan kombinasi dari tiga jenis kernel: linear, sigmoid, dan gaussian. *Input weight* dan *bias* ditentukan secara acak. Matrik *output hidden layer* didapat dari kombinasi inialisasi parameter secara acak dari ketiga fungsi tersebut. Berikut ini merupakan model matematis algoritma ELM standar.

Sebanyak N dan *activation function* $g(x)$ dapat digambarkan secara matematis sebagai berikut :

$$\sum_{i=1}^N \beta_i g_i(x_j) = \sum_{i=1}^N \beta_i g(W_i \cdot x_j + b_i) = O_j \quad (4)$$

Dimana :

$$J = 1, 2, \dots, N$$

$w_i = (w_{i1}, w_{i2}, \dots, w_{in})^T$ = merupakan vektor dari weight yang menghubungkan i th *hidden nodes* dan *input nodes*

$i = (i_1, i_2, \dots, i_m)^T$ = merupakan *weight vector* yang menghubungkan i th *hidden nodes* dan *output nodes*

b_i bias dari i th *hidden nodes*

$w_i \cdot x_i$ Merupakan *inner produk* dari w_i dan x_i

SLFNs dengan N *hidden nodes* dan *activation function* $g(x)$ diasumsikan dapat meng-approximate dengan tingkat error 0 atau dapat dinotasikan sebagai berikut.

$$\sum_{j=1}^n \|O_j - t_j\| = 0 \text{ sehingga } O_j = t_j \quad (5)$$

$$\sum_{i=1}^N \beta_i g(W_i \cdot x_j + b_i) = t_j \quad (6)$$

Persamaan (6) dapat dituliskan secara sederhana sebagai.

$$H = T \quad (7)$$

$$H = \begin{pmatrix} w_1 & \dots & w_N & b_1 & \dots & b_N & x_1 & \dots & x_N \\ g(w_1 \cdot x_1 + b_1) & \dots & g(w_N \cdot x_1 + b_N) \\ \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ g(w_1 \cdot x_N + b_1) & \dots & g(w_N \cdot x_N + b_N) \end{pmatrix} \quad (8)$$

$$= \begin{pmatrix} 1 \\ \vdots \\ T \end{pmatrix} \quad (9)$$

$$T = \begin{pmatrix} t_1 \\ \vdots \\ t_N \end{pmatrix} \quad (10)$$

H pada persamaan (8) di atas adalah *hidden layer* $g(w_1 \cdot x_1 + b_1)$. *Output matrix* menunjukkan *output* dari *hidden neuron* yang berhubungan dengan *input* x_i . merupakan matrix dari *output weight* dan T matrix dari target atau *output*. Pada ELM *input weight* dan *hidden bias* ditentukan secara acak, maka *output weight* yang berhubungan dengan *hidden layer* dapat ditentukan dari persamaan (9).

$$= H^T T \quad (11)$$

2) Perankingan *hidden neuron* dengan MRSR

Sebagai langkah kedua dalam metode OPELM, *Multiresponse Sparse Regression* (MRSR) diterapkan untuk perankingan *hidden neuron* berdasarkan keakurasiannya. MRSR diperkenalkan oleh Similä dan Tikka[4]. Ide utama dari algoritma ini adalah menambahkan setiap kolom dari matriks regressor satu per satu ke dalam model $Y^k = X W^k$, dimana $Y^k = [y_1^k \dots y_p^k]$ adalah pendekatan target model T . Dimana $X = [x_1 \dots x_m]$ merupakan $n \times m$ matrik regressor, $T = [t_1 \dots t_p]$ $n \times p$ matrik target dan W^k *weight matrix* memiliki k baris tak nol dan sebuah kolom baru pada matriks regressor ditambahkan ke model. Sebagai catatan bahwa MRSR adalah perluasan dari algoritma *Least Angle Regression* (LARS) yang sebenarnya merupakan teknik perankingan variabel. Solusi yang dihasilkan sangat tepat jika permasalahan berbentuk linier. *Neural network* dibangun pada tahap sebelumnya, *hidden layer neuron* diranking oleh algoritma LARS. Karena bagian antara *hidden* dan *output layer* dari jaringan *neural network* adalah linear, LARS akan menemukan perankingan terbaik (*best ranking*).

Dengan $k=0$, inialisasi Y^0 dan W^0 ke nol, dan normalisasi T dan X ke rata nol. Definisikan *cummulative correlation* c_j^k antara regressor x_j dan residual, serta maksimum *cummulative correlation* c_{max}^k , maka didapat kumpulan regressor yang memenuhi korelasi maksimumnya, dinotasikan dengan A dan membentuk matrik $n \times |A|$ $X_A = [x_{j_1} \dots x_{j_{|A|}}]$.

$$c_j^k = \|(T - Y^k)^T x_j\| = \sum_{i=1}^p |(t_i - y_i^k)^T x_j| \quad (12)$$

$$c_{max}^k = \max_j \{c_j^k\}, A = \{j | c_j^k = c_{max}^k\} \quad (13)$$

Kemudian hitung parameter OLS \bar{W}^{k+1} dan estimasi OLS untuk target \bar{Y}^{k+1}

$$\bar{W}^{k+1} = (X_A^T X_A)^{-1} X_A^T T \quad (14)$$

$$\bar{Y}^{k+1} = X_A (X_A^T X_A)^{-1} X_A^T T \quad (15)$$

Greedy forward selection menambahkan regressor berdasarkan persamaan (12) dengan menggunakan estimasi OLS (15). Algoritmanya dapat didefinisikan dengan perpindahan dari estimasi MRSR Y^k menuju estimasi OLS \bar{Y}^{k+1} , $U^k = \bar{Y}^{k+1} - Y^k$ namun cara tersebut tidak diijinkan. Langkah yang mungkin diambil ke arah U^k untuk regressor x_j dimana $j \notin A$ memiliki *cummulative correlation* besar dengan residual yang siap ditambahkan ke regressor.

Untuk update estimasi MRSR Y^{k+1} perlu dihitung nilai step size γ^k yang tepat. Ini membuat *cummulative correlation* pada langkah selanjutnya sebagai sebuah fungsi

$$C_j^{k+1}(\gamma) = |1 - \gamma| C_{max}^k \text{ untuk } j \notin A \quad (16)$$

$$C_j^{k+1}(\gamma) = \|a_j^k - \gamma b_j^k\| \text{ untuk } j \in A \quad (17)$$

Dimana $a_j^k = (T - Y^k)^T x_j$ dan $b_j^k = (\bar{Y}^{k+1} - Y^k)^T x_j$. Regressor baru dengan index $j \notin A$ akan dimasukkan ke model ketika (13) dan (14) sama. Ini terjadi jika *step size* diambil dari

$$\Gamma_j = \left\{ \frac{C_j^k + s^T a_j^k}{C_{max}^k + s^T b_j^k} \right\}_{s \in \mathcal{S}} \quad (18)$$

Dimana \mathcal{S} adalah *sign vector* 2^p , matrik $p \times 1$ dengan element \mathcal{S} bisa bernilai 1 atau -1. Pilihan terbaik adalah *step size* terkecil dengan nilai positive yang menghasilkan regressor baru

$$\gamma^k = \min \{ \gamma | \gamma \geq 0 \text{ dan } \gamma \in \Gamma_j \text{ untuk } j \in A \} \quad (19)$$

didapat *update* estimasi target sebagai berikut.

$$Y^{k+1} = Y^k + \gamma^k (\bar{Y}^{k+1} - Y^k) \quad (20)$$

Dan *weight matrix* yang memenuhi (17) dan (12) diupdate menjadi

$$W^{k+1} = (1 - \gamma^k) W^k + \gamma^k \bar{W}^{k+1} \quad (21)$$

Parameter dari regressor terpilih disusutkan berdasarkan persamaan (21). Pemilihan akhir model dari m kemungkinan didasarkan pada akurasi prediksi untuk data baru.

3) Seleksi neuron dengan Leave-One-Out

Setelah perankingan neuron dari *hidden layer* telah diperoleh jumlah neuron terbaik untuk model yang dipilih, digunakan LOO untuk memvalidasi. Menghitung LOO error bisa sangat memakan waktu ketika kumpulan data cenderung memiliki sampel neuron penting. Untungnya, PERSS statistik (or *PREDiction Sum of Squares*) memberikan formula langsung dan tepat untuk perhitungan kesalahan ini pada model linier:

$$PRESS = \frac{y_i - h_i b}{1 - h_i P h_i^T} \quad (28)$$

Dimana i dinotasikan sebagai *hidden node* ke- i , P didefinisikan sebagai $P = (H^T H)^{-1}$, H adalah *hidden layer output matrix* yang didefinisikan sebelumnya, dan h_i adalah kolom pada keluaran matrik lapisan tersembunyi.

Jumlah neuron yang optimal didapat dari estimasi LOO error pada jumlah node-node (yang telah diranking berdasarkan akurasi) dan menyeleksi jumlah neuron dari minimum errornya.

Kemudian neuron hasil pemangkasan tersebut digunakan untuk menghitung *Output weight* yang didapat dari hasil invers dari matrik *hidden layer* dan target *Output*. Untuk mengetahui performa algoritma dari tiap metode maka dihitung nilai *fitness* yang menunjukkan MAPE (*Mean Absolute Percentage Error*) dari tiap perhitungan fungsi objektif *training* dan *testing*, yaitu error rata-rata dari pemodelan OPELM dan ELM.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{Y_{prediksi} - Y_{target}}{Y_{target}} \right| \cdot 100\% \quad (29)$$

jika nilai MAPE makin mendekati nol maka kinerja hasil peramalan semakin baik.

IV. SIMULASI, HASIL, DAN PEMBAHASAN

A. Analisa Data

Sistem peramalan ini berdasarkan *similar day*, misalnya pola beban hari Senin pada minggu ini akan mempunyai pola

yang sama dengan hari Senin pada minggu yang akan datang. Pada Tugas Akhir ini digunakan data beban listrik harian pada hari kerja (*week day*), Senin sampai dengan Jumat dan data beban listrik harian untuk akhir minggu (*week end*), hari Sabtu dan Minggu.

Pengidentifikasi karakteristik atau pola konsumsi beban perlu dilakukan sebelum melakukan peramalan. Karena kecenderungan pola aktivitas konsumen listrik akan berulang dengan tren yang meningkat. Maka hendaknya data akhir minggu yang dipilih memperhatikan korelasinya terhadap data aktual yang diramalkan sehingga proses pengenalan pola beban memperoleh hasil yang akurat. Proses *testing* dari hasil *training* terbaik tidak selalu memberikan hasil yang baik pula, tergantung pada korelasi data yang digunakan sebagai masukan[2]. Jika data yang digunakan untuk *testing* mempunyai korelasi yang hampir sama antara data-data masukan yang digunakan maka hasil *testing* akan memberikan hasil yang bagus, dan sebaliknya.

B. Analisis Hasil

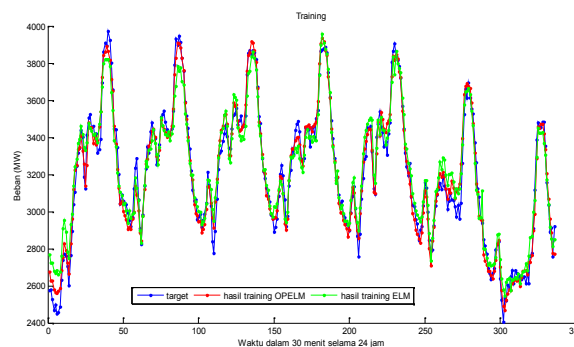
Berikut ini merupakan hasil peramalan untuk satu minggu ke depan menggunakan OPELM dan ELM. Fungsi aktivasi yang digunakan adalah kombinasi antara *linear*, *sigmoid*, dan *gaussian* dengan maksimum jumlah *hidden neuron* ditetapkan sebanyak 25. Hasil optimasi menghasilkan jumlah *hidden neuron* yang lebih kecil, yaitu sebanyak 8 *hidden neuron*. Pada ELM digunakan fungsi *linear* karena data yang diramalkan bersifat stationer dengan jumlah *hidden neuron* sama dengan yang digunakan OPELM. Nilai keakuratan pada data *training*

Tabel 1.
Keakuratan Model Peramalan Data *Training* Menggunakan Metode OPELM dan ELM untuk Hari Satu Minggu ke Depan

| Metode | Fungsi Aktivasi | Jumlah <i>hidden neuron</i> | MAPE <i>training</i> (%) |
|--------|-----------------|-----------------------------|--------------------------|
| OPELM | <i>lsg</i> | 8 | 1,5391 |
| ELM | <i>linear</i> | | 2,1570 |

^a*lsg* – *linear, sigmoid, gaussian*

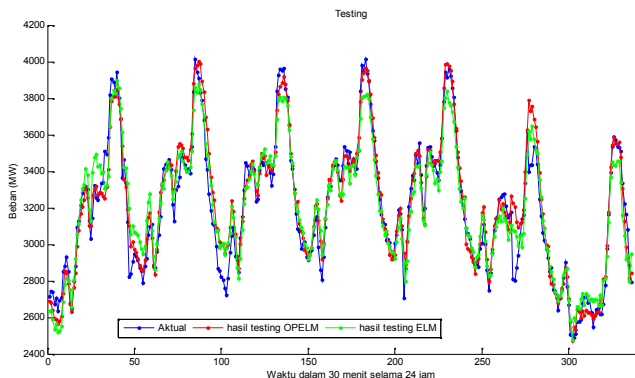
menggunakan metode OPELM dan ELM untuk satu minggu ke depan dapat dilihat pada Tabel 1. Berdasarkan Tabel diketahui bahwa metode OPELM memiliki nilai MAPE *training* sebesar 1,5391% sedangkan pada metode ELM sebesar 2,1570%. Nilai MAPE OPELM lebih kecil dibandingkan metode ELM. Grafik perbandingan hasil *training* antara OPELM dan ELM disajikan pada Gambar 4.



Gambar 4. Plot Perbandingan Hasil *Training* Peramalan Beban Listrik untuk Satu Minggu ke Depan

Untuk melihat seberapa baik model yang terbentuk, dilakukan validasi model dengan data *testing*. Tabel 2 merupakan perbandingan keakuratan hasil ramalan dari metode OPELM dan ELM. Berdasarkan tabel diketahui bahwa metode OPELM memiliki nilai MAPE *testing* sebesar 2,1935% dan ELM 2,6470%. Nilai MAPE OPELM lebih kecil dibandingkan dengan ELM. Hasil *testing* menggunakan metode OPELM dan ELM ditunjukkan pada Gambar 5. Pada gambar menunjukkan bahwa hasil peramalan dengan metode OPELM memiliki pola beban yang sama dengan pola beban data aktual tetapi memiliki *error* lebih kecil dibandingkan hasil peramalan dengan metode ELM.

Jika hasil pengujian peramalan direpresentasikan dalam jumlah beban (MW) per harinya dan ditunjukkan MAPE *error*, maka dapat ditampilkan pada Tabel 3. Pada tabel didapatkan besar nilai MAPE dari proses *testing* dari metode OPELM dan ELM. MAPE *testing* terkecil dengan metode OPELM yang menunjukkan bahwa hasil peramalan terbaik ditunjukkan pada peramalan pada hari Jumat sebesar 1,357%, sedangkan MAPE *testing* terbesar dengan metode OPELM yang menunjukkan hasil peramalan paling tidak akurat terjadi pada peramalan hari Sabtu sebesar 3,173%. Secara keseluruhan dari hasil *error* MAPE yang didapat, peramalan menggunakan metode OPELM memiliki nilai keakuratan lebih baik dari metode ELM.



Gambar 5. Plot Hasil *Testing* Menggunakan Metode OPELM dan ELM untuk Satu Minggu ke Depan

Tabel 3. Hasil keseluruhan Pengujian Peramalan pada Seluruh Hari Menggunakan Metode OPELM dan ELM

| Metode | Senin | Selasa | Rabu | Kamis | Jumat | Sabtu | Minggu |
|--------|-------|--------|-------|--------|-------|-------|--------|
| OPELM | 2,573 | 2,612 | 2,466 | 1,553 | 1,357 | 3,173 | 1,617 |
| ELM | 3,663 | 2,592 | 2,840 | 2,1667 | 2,217 | 2,512 | 2,534 |

Pada pembahasan berikut ini akan dipaparkan hasil pengujian peramalan pada hari jumat dengan prosentase nilai *error*, perbandingan hasil dengan metode ELM, dan representasi hasil ramalan per 30 menit. Perbandingan nilai keakuratan hasil ramalan dari metode OPELM dan ELM untuk hari Jumat dapat dilihat pada Tabel 4. Hasil pengujian peramalan direpresentasikan per 30 menit ditunjukkan pada Tabel 5 dan hasil *testing* ditunjukkan pada Gambar 6.

Berdasarkan Tabel 5, untuk beban puncak pada peramalan OPELM yaitu sebesar 3990,56 MW pada pukul 19.00 dan pada peramalan ELM sebesar 3840,88 MW pada pukul

Tabel 4. Perbandingan Keakuratan Hasil *Testing* Menggunakan Metode OPELM dan ELM untuk Hari Jumat

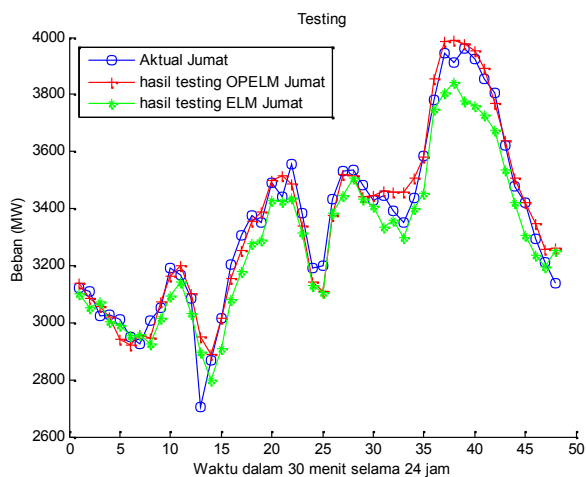
| Metode | Fungsi Aktivasi | Jumlah <i>hidden neuron</i> | MAPE <i>testing</i> (%) |
|--------|-----------------|-----------------------------|-------------------------|
| OPELM | <i>lsg</i> | 8 | 1,3579 |
| ELM | <i>linear</i> | | 2,2179 |

¹*lsg* – *linear, sigmoid, gaussian*

Tabel 5. Hasil *Testing* Peramalan Beban untuk Hari Jumat, 24 Februari 2012

| Jam | Data Aktual (MW) | OPELM (MW) | Error OPELM (MW) | ELM (MW) | Error ELM (MW) |
|-----------------|------------------|------------|------------------|----------|----------------|
| 00.30 | 3120,84 | 3139,84 | 0,6089 | 3101,41 | 0,6226 |
| 01.00 | 3111,64 | 3086,22 | 0,8170 | 3052,85 | 1,8895 |
| 01.30 | 3022,26 | 3054,84 | 1,0779 | 3071,86 | 1,6411 |
| 02.00 | 3028,36 | 3018,97 | 0,3102 | 3004,24 | 0,7963 |
| 02.30 | 3010,56 | 2942,74 | 2,2528 | 2989,08 | 0,7135 |
| 03.00 | 2950,36 | 2920,83 | 1,0009 | 2955,52 | 0,1748 |
| 03.30 | 2924,94 | 2959,58 | 1,1845 | 2957,33 | 1,1074 |
| 04.00 | 3008,00 | 2945,51 | 2,0776 | 2924,57 | 2,7737 |
| 04.30 | 3051,38 | 3073,65 | 0,7300 | 3014,48 | 1,2094 |
| 05.00 | 3190,00 | 3161,91 | 0,8805 | 3094,30 | 3,0001 |
| 05.30 | 3168,64 | 3198,06 | 0,9284 | 3142,48 | 0,8256 |
| 06.00 | 3083,10 | 3101,07 | 0,5828 | 3033,31 | 1,6150 |
| 06.30 | 2706,38 | 2951,40 | 9,0534 | 2898,37 | 7,0941 |
| 07.00 | 2867,96 | 2888,51 | 0,7166 | 2799,29 | 2,3946 |
| 07.30 | 3013,98 | 3016,81 | 0,0939 | 2909,04 | 3,4818 |
| 08.00 | 3205,44 | 3154,23 | 1,5977 | 3081,16 | 3,8770 |
| 08.30 | 3307,98 | 3253,84 | 1,6367 | 3180,83 | 3,8437 |
| 09.00 | 3376,28 | 3356,19 | 0,5951 | 3276,18 | 2,9647 |
| 09.30 | 3350,86 | 3389,88 | 1,1643 | 3288,66 | 1,8561 |
| 10.00 | 3488,88 | 3498,69 | 0,2812 | 3428,88 | 1,7198 |
| 10.30 | 3442,98 | 3514,00 | 2,0628 | 3425,70 | 0,5018 |
| 11.00 | 3554,98 | 3486,16 | 1,9359 | 3437,49 | 3,3048 |
| 11.30 | 3383,28 | 3339,37 | 1,2978 | 3312,32 | 2,0974 |
| 12.00 | 3191,06 | 3142,77 | 1,5133 | 3132,00 | 1,8509 |
| 12.30 | 3197,78 | 3109,26 | 2,7683 | 3105,18 | 2,8957 |
| 13.00 | 3432,02 | 3374,97 | 1,6624 | 3382,38 | 1,4463 |
| 13.30 | 3532,28 | 3518,89 | 0,3790 | 3446,53 | 2,4277 |
| 14.00 | 3535,44 | 3514,49 | 0,5926 | 3507,21 | 0,7985 |
| 14.30 | 3482,44 | 3443,19 | 1,1271 | 3434,59 | 1,3740 |
| 15.00 | 3429,38 | 3445,57 | 0,4721 | 3406,58 | 0,6648 |
| 15.30 | 3443,44 | 3461,23 | 0,5166 | 3335,21 | 3,1430 |
| 16.00 | 3392,76 | 3458,38 | 1,9342 | 3357,78 | 1,0310 |
| 16.30 | 3351,46 | 3456,35 | 3,1296 | 3296,74 | 1,6328 |
| 17.00 | 3435,76 | 3505,50 | 2,0300 | 3398,33 | 1,0895 |
| 17.30 | 3582,98 | 3581,69 | 0,0360 | 3454,31 | 3,5911 |
| 18.00 | 3779,28 | 3855,69 | 2,0219 | 3748,31 | 0,8195 |
| 18.30 | 3943,28 | 3985,23 | 1,0639 | 3804,17 | 3,5279 |
| 19.00 | 3913,16 | 3990,56 | 1,9780 | 3840,88 | 1,8470 |
| 19.30 | 3961,28 | 3976,05 | 0,3729 | 3775,96 | 4,6784 |
| 20.00 | 3924,58 | 3953,71 | 0,7422 | 3760,32 | 4,1855 |
| 20.30 | 3856,16 | 3891,76 | 0,9232 | 3728,68 | 3,3059 |
| 21.00 | 3805,34 | 3768,00 | 0,9812 | 3673,19 | 3,4729 |
| 21.30 | 3622,76 | 3636,79 | 0,3872 | 3535,04 | 2,4214 |
| 22.00 | 3477,28 | 3505,31 | 0,8062 | 3421,95 | 1,5913 |
| 22.30 | 3419,70 | 3418,94 | 0,0223 | 3305,50 | 3,3395 |
| 23.00 | 3292,46 | 3346,09 | 1,6289 | 3237,16 | 1,6796 |
| 23.30 | 3211,70 | 3255,51 | 1,3639 | 3195,98 | 0,4895 |
| 24.00 | 3139,30 | 3259,83 | 3,8395 | 3253,96 | 3,6524 |
| Error Maksimum | | | 9,0534 | | 7,0941 |
| Error Minimum | | | 0,0223 | | 0,1748 |
| Error Rata-rata | | | 1,3579 | | 2,2179 |

19.00. Sedangkan dari data aktual beban puncak terjadi pada pukul 19.30 sebesar 3961,28 MW. Hasil OPELM memiliki *error* maksimum 9,0534% pada beban pukul 06.30, *error* minimum 0,0223% pada pukul 22.30, dan *error* rata-rata 1,3579%. Sedangkan hasil dari ELM menunjukkan *error* maksimum 7,0941% pada pukul 06.30, *error* minimum 0,1748% pada pukul 03.00, dan *error* rata-rata 2,2179%.



Gambar 6. Plot Hasil *Testing* Menggunakan Metode OPELM dan ELM untuk Hari Jumat

Gambar 6 menunjukkan bahwa hasil peramalan dengan metode OPELM memiliki pola beban yang sama dengan pola beban data aktual tetapi memiliki *error* lebih kecil dibandingkan hasil peramalan dengan metode ELM.

Secara keseluruhan dari tabel dan grafik, peramalan OPELM memberikan hasil yang lebih baik dibanding dengan metode ELM dilihat dari errornya.

V. KESIMPULAN

Kesimpulan yang dapat diambil dari penelitian ini adalah sebagai berikut.

- 1) Peramalan beban listrik jangka pendek menggunakan metode *Optimally Pruned Extreme Learning* (OPELM) memperoleh hasil peramalan yang lebih akurat dibandingkan dengan metode *Extreme Learning Machine* (ELM).
- 2) Hasil terbaik OPELM ditunjukkan pada peramalan beban pada hari Jumat dengan MAPE *testing* sebesar 1,3579%, sementara MAPE *testing* ELM sebesar 2,2179%.
- 3) Pengidentifikasi karakteristik atau pola beban perlu dilakukan sebelum melakukan peramalan. Hal ini terkait dengan korelasi antara data yang menjadi masukan dengan data aktual. Semakin besar nilai korelasinya (kemiripan pola) maka *error* yang didapat semakin kecil. Terdapat juga faktor-faktor lain yang mempengaruhi pola konsumsi listrik, yaitu pengaruh cuaca dan tingkat perekonomian masyarakat setempat.

DAFTAR PUSTAKA

- [1] Suswanto, D., "*Sistem Distribusi Tenaga Listrik : Untuk Mahasiswa Elektro (Edisi Pertama)*", Jurusan Teknik Elektro Fakultas Teknik Universitas Negeri Padang, Padang, (2009).
- [2] Shayeghi, H., Shayanfar, H.A., and Azimi, G., "A Hybrid Particle Swarm Optimization Back Propagation Algorithm for Short Term Load Forecasting". *International Journal on Technical and Physical Problems of Engineering*, Vol.1, Issue 3, (2010) 12-22.
- [3] A. Sorjamaa, Y. Miche, R. Weiss and A. Lendasse, "Long Term Prediction of Time Series Using NNE-based Projection and OP-ELM". In *Proc. Inter. Jt. Conf. Artif. Neural Netw*, Hong Kong, China, (2008) 2675-2681.

- [4] T. Similä and Tikka, "Multiresponse sparse regression with application to multidimensional scaling", in *Proc. Int. Conf. Artif. Neural Netw.*, vol. 3697, (2005) 97-102.