

EXTREME LEARNING MACHINE: APLIKASI PADA SHORT TERM LOAD FORECASTING

Hasmina¹, Tari Mokui²

¹Jurusan Teknik Elektro, ²Jurusan Teknik Informatika, Universitas Haluoleo Kendari

Masuk: 30 Oktober 2009, revisi masuk: 29 Desember 2009, diterima: 11 Januari 2010

ABSTRACT

Accurate load forecasting becomes an important task for operating and planning of a power system to maintain the security of power supply dispatched to the consumers. This paper proposes an advanced method, namely Extreme Machine Learning (ELM), to forecast load in short time period. It is observed that implementation of the ELM can reduce cost and time significantly. Comparison results with a well known algorithm, called the Back Propagation (BP), show that the ELM can converge a hundred times faster than BP. In addition, the ELM needs 100 hidden neurons while the BP needs 2 hidden neurons to achieve similar result. This reveals that the number of hidden neurons is not a problem for ELM as long as there is sufficient memory to perform its computation.

Keywords: *Short-term Load Forecasting, Extreme Learning Machine, Back Propagation*

INTISARI

Perkiraan beban listrik (*load forecasting*) yang akurat menjadi tugas yang sangat penting dalam pengoperasian dan perencanaan sistem ketenagalistrikan untuk mempertahankan sekuritas dan suplai daya ke konsumen. Paper ini menawarkan satu metode terbaru yang disebut *Extreme Learning Machine* (ELM) untuk memprediksi beban dalam durasi waktu yang singkat (*short term load forecasting*). Ditemukan bahwa pengimplementasian ELM dapat mengurangi biaya (*cost*) serta waktu secara signifikan. Dalam penelitian, performans ELM dibandingkan dengan *Back Propagation* (BP) merupakan algoritma yang sering diaplikasikan dalam *computational intelligence*. ELM membutuhkan waktu seratus detik lebih cepat untuk melakukan *perkiraan beban*. ELM membutuhkan 100 *hidden neuron* untuk memperoleh hasil yang sama dengan BP yang hanya membutuhkan 2 *hidden neuron*. Hal ini membuktikan bahwa jumlah *hidden neuron* yang lebih banyak tidak menjadi masalah bagi ELM sepanjang memori yang tersedia mencukupi.

Kata Kunci: *Perkiraan Beban Jangka Pendek, Extreme Learning Machine, Back Propagation*

PENDAHULUAN

Perkiraan beban listrik ini disebut dengan *load forecasting* dalam paper ini merupakan hal sangat penting dalam proses pembangkitan tenaga listrik terutama dalam sisi pengoperasian sistem, perencanaan, distribusi dan pengontrolan suplai energi listrik ke konsumen. Sangatlah perlu bagi suatu sistem ketenagalistrikan untuk beroperasi pada *rate* yang mendekati kapasitas maksimum untuk tujuan optimisasi serta efisiensi penggunaan secara komersial dengan alasan lingkungan, dan ekonomi serta alasan teknis lainnya. Pada sistem

dalam deregulated market seperti halnya Australia, *spinning reserve* telah menjadi satu *ancillary service* yang sangat penting. Oleh karenanya, *peak load forecasting* (PLF) yang akurat menjadi hal ini yang sangat penting dalam lingkungan industri sektor ketenagalistrikan yang terbuka untuk menentukan kapasitas pembangkitan, transmisi dan distribusi serta investasi yang dikeluarkan. *Load forecasting* yang dilakukan dan ditujukan untuk menentukan *reserve generate* dimasa yang akan datang dan dilaksanakan dalam waktu *real time*.

¹tarimokui@gmail.com

Tujuan utama dalam *load forecasting* adalah untuk mendapatkan model ini yang representatif untuk menjelaskan mengenai perilaku beban atau biasa juga disebut *load behavior* (Maia et.al, 2007). Secara umum, model ini sangat kompleks karena ini sifatnya yang nonlinear maka akan sangat tergantung pada berbagai macam factor seperti cuaca, dampak ekonomi, perilaku pelanggan dan histories beban (*load* itu sendiri). Oleh karenanya, beberapa penyederhanaan ini dilakukan untuk memodelkan sistem yang kompleks

Teknik pemodelan berdasarkan angka statistik merupakan hal yang umum dilakukan dalam proses *load forecasting* (Keib et.al., 1995; Amjady, 2001; Espinoza et.al., 2005). Untuk meningkatkan keakuratan proses dengan cara forecasting serta mempersingkat waktu komputasi, PLF dilaksanakan dengan teknik-teknik komputasi cerdas seperti halnya *switched adaptive system* (Maia, et.al, 2007), *feed forward neural networks* berdasarkan pada metode *steepest descent*, *Bayesian regularization*, *resilient and adaptive back-propagation* (Saini, 2007), *Support Vector Machine* (Chen et.al, 2004) dan masih banyak lagi.

Paper ini menawarkan satu metodologi baru, *Extreme Learning Machine* (ELM), untuk melakukan *short term load forecasting*, yaitu prediksi beban listrik ini dalam jangka pendek selama satu jam. *Forecasting* dilakukan dengan mempertimbangkan konsumsi energi oleh pelanggan selama dua hari sebelumnya. Faktor-faktor lain seperti halnya temperatur, tidak menjadi bahan pertimbangan dalam pa-per ini. Tujuan utama yang akan dicapai adalah membangun suatu model/prediktor untuk memprediksi total energi listrik yang dibutuhkan oleh pelanggan. Prediksi pertama ini dilakukan dengan menggunakan algoritma ELM dan selanjutnya untuk kasus yang sama dilakukan pengujian dengan menggunakan algoritma *Back propagation* (BP) yang telah sering diaplikasikan pada bidang ini. Langkah selanjutnya adalah membandingkan performansi kedua algoritma ini dalam hal kecepatan serta keakuratan dalam melakukan proses *forecasting*.

ELM merupakan salah satu metode *data mining* yang diperkenalkan oleh

Guang Bin Huang, Qin Yu Zhu dan Chee-Kheong Siew (Han et.al, 2001; Huang et.al, 2006). ELM merupakan algoritma baru dalam *single hidden layer feed forward neural networks* (SLFNs) yang memilih *hidden neuron* secara random dan menentukan *output weight*-nya secara analitis.

Hasil dari *benchmarking* terhadap ELM ini untuk berbagai kasus riil yang kompleks seperti klasifikasi penyakit diabetes, *image segment*, *satellite image* dan lain-lain menunjukkan bahwa algoritma ini memiliki hasil yang sangat baik saat diaplikasikan untuk memprediksi beberapa kasus yang berbeda. (Huang et.al, 2006) mengklaim bahwa ELM ini memiliki performansi yang baik serta dapat memecahkan masalah dalam waktu yang sangat cepat pada beberapa dari bidang aplikasi yang sangat kompleks dibandingkan dengan beberapa algoritma lain seperti halnya *Back Propagation* (BP) dan *Support Vector Machine* (SVM). Hal ini karena algoritma ELM cenderung mencapai solusi secara langsung tanpa mempertimbangkan faktor *local minima*, *learning rate* yang tidak memadai, *over fitting* dan lain sebagainya, yang biasanya menjadi masalah pada algoritma-algoritma *gradient based learning*.

Selanjutnya juga telah dibuktikan bahwa tidak seperti halnya teori aproksimasi fungsi ini yang konvensional yang membutuhkan pengaturan *input weight* dan *hidden layer bias*, ELM dapat menentukan secara random kedua parameter tersebut jika fungsi aktivasinya adalah differensial tak terbatas. Pada algoritma-algoritma yang ditemukan sebelumnya, biasanya diperlukan pengaturan semua parameter SLFN, termasuk *input weight* w_i dan *hidden layer biases* b_i . Namun ELM ini, telah menunjukkan bahwa tidak-lah penting untuk mengatur semua parameter ini karena sesungguhnya *hidden layer output* matriks H dapat saja tidak berubah jika pada awal proses learning ke semua parameter ini telah diatur *random value*-nya (Huang et.al, 2006). Menurut Huang et.al (2006), metode learning ELM dapat disingkat sebagai berikut:

Langkah 1: Diberikan satu set *training data*,

$$\mathcal{S} = \left\{ (x_i, t_i) \mid x_i \in R^n, t_i \in R^m, i = 1 \dots N \right\} \text{ fungsi}$$

aktivasi atau fungsi kernel $g(x)$, dan *hidden neuron* atau kernel number N .

Langkah 2: Masukkan *input weight* w_i dan bias b_i secara *random*, dengan $i = 1, \dots, N$

Langkah 3: Lakukan perhitungan *hidden layer output* matriks H . Langkah 4: lakukan perhitungan *output weight* (β), dengan $\beta = H^+T$, dimana, $T = [t_1, \dots, t_N]^T$. H^+ merupakan *Moore-Penrose generalized invers* dari matriks H .

Beberapa penelitian yang telah dilakukan (Huang et.al, 2006), menunjukkan bahwa algoritma ELM memiliki sifat dan karakteristik penting sebagai berikut:

- *Minimum training error*. ELM memiliki *norm least solution* terkecil pada satu

fungsi linier umum, $\hat{\beta} = H^+T$ yang berarti bahwa *training error* terkecil dapat dicapai dengan solusi sebagai berikut:

$$\|H\hat{\beta} - T\| = \|HH^+T - T\| = \min_{\beta} \|H\beta - T\| \quad \dots(1)$$

- *Smallest norm of weights*. Disebutkan

bahwa solusi tertentu, $\hat{\beta} = H^+T$ memiliki *norm* terkecil diantara semua *least square solution* dari $H\beta = T$:

$$\|\hat{\beta}\| = \|H^+T\| \leq \|\beta\|, \quad \forall \beta \in \left\{ \beta : \|H\beta - T\| \leq \|H\hat{\beta} - T\|, \forall z \in \mathbb{R}^{N \times N} \right\} \dots\dots(2)$$

- Sistem linier $H\beta = T$ memiliki satu *norm least square solution* minimum yang

unik, yaitu $\hat{\beta} = H^+T$

Metodologi Penelitian: Data set, data yang dipergunakan merupakan data Australian National Electricity ini yang di-download dari website NEMCO (*National Electricity Market Company*). Secara spesifik, data ini merupakan pengukuran selama setengah jam aktivitas pada *electricity market* Queensland selama bulan April 2006. Data ini berupa *total demand* dan harga listrik per Kwh (*RRP-Regional Reference Price*). Namun demikian, riset ini hanya mempertimbangkan pada *total demand* saja. Setiap *data series* memiliki jumlah sebanyak 1440 titik. 1000 titik pertama akan dipergunakan sebagai *training set*. Sedangkan sisanya sebanyak 440 titik

menjadi *testing set*. Data selama dua hari yaitu sebanyak 48 titik akan digunakan untuk memprediksi *demand* satu jam di masa yang akan datang.

Algoritma yang digunakan, *data set* yang diperoleh digunakan untuk melatih dan mengetes performansi *Back-Propagation* (BP) dan ELM. Hasil penelitian yang dilakukan oleh Huang et.al. (2006) menunjukkan bahwa BP membutuhkan cost yang besar dalam hal waktu dan memori yang digunakan. Namun demikian, algoritma memiliki performansi yang cukup baik karena umumnya dapat diperoleh tingkat keakuratan yang tinggi sebab algoritma ini mengatur *weight* setiap koneksi secara iteratif sehingga outputnya akan mendekati nilai yang diharapkan. Pada umumnya dilakukan beberapa percobaan dengan pengaturan *weight* awal yang berbeda dan dipilih secara *random*. Sekumpulan *weight* awal mengalami proses penyesuaian yang berbeda yang tentu saja akan menghasilkan *output* yang berbeda pula.

Saat ini, satu metode baru telah diperkenalkan disebut ELM, hampir sama dengan BP, hanya saja ELM merupakan algoritma yang jauh lebih cepat dibandingkan dengan BP. ELM terdiri atas tiga lapisan (*layer*), yaitu *layer input*, *hidden* dan *output*. *Weight* yang menghubungkan antara *input* dan *hidden layer*, yang disebut *input weight*, ditentukan secara *random*. Sedangkan *weight* yang menghubungkan antara *layer hidden* dan *layer output* yang juga disebut *output layer* dikalkulasi dengan menggunakan Moore-Penrose pseudoinverse. Karena tidak dibutuhkan proses komputasi yang berulang, maka proses *training* ELM menjadi sangat cepat. Namun demikian, dibutuhkan memori yang sangat besar dalam melakukan kalkulasi dengan metode pseudoinverse, yang merupakan kendala umum pada proses *training* yang cepat. Dalam tulisan ini, sekumpulan eksperimen dilakukan, masing-masing dengan fungsi aktivasi sigmoid untuk *hidden layer* dan fungsi aktivasi linear untuk *output layer*.

Sebagai catatan, 48 titik yang merupakan *input*, disebut *attribute* pada ELM. Sedangkan *target* atau *output* hasil prediksi disebut juga *class* atau *category*. Is-

tilah-istilah ini akan digunakan pada bagian selanjutnya dari tulisan ini.

Normalisasi data, pada ELM dan BP, proses *preprocessing* data diawali dengan menormalisasi *training* dan *testing* data dengan menggunakan formula sebagai berikut:

$$\text{Data} = \frac{(\text{Data} - \text{min_Data})}{(\text{max_Data} - \text{min_Data})}$$

Satu hal yang perlu diingat, *class/category* tidak mengalami proses normalisasi. Performansi dalam tingkat keakuratan dalam memprediksi *testing* data ini diketahui dengan mengukur nilai *Root Mean Square Error* (RMSE) pada kedua algoritma, dalam BP dan ELM. RMSE merupakan ukuran yang menyatakan perbedaan antara nilai yang diprediksi oleh model atau satu estimator dan nilai sesuatu yang dimodelkan atau telah diobservasi. Formula untuk menghitung RMSE adalah sebagai berikut:

$$\text{RMSE} = \sqrt{\frac{\sum (\text{Error})^2}{n}}$$

Dengan: *Error* = *demand* yang diprediksi-kan *demand* aktualnya; *n*= jumlah *error* yang terjadi di sekian banyak data yang dikalkulasikan.

Pada bagian ini, didiskusikan mengenai hasil studi empirik dengan menggunakan BP dan ELM dalam kasus *load forecasting*. Evaluasi performansi yang ingin diketahui dengan menjalankan program ini adalah seberapa cepat algoritma ini dapat memprediksi keluaran dari masukan yang diberikan berdasarkan model yang dihasilkan dari *training set*, nilai RMSE dalam MW, serta kemudahan dalam mengoperasikannya. Mula-mula ditampilkan hasil pengukuran untuk masing-masing algoritma selanjutnya akan ditampilkan hasil perbandingan untuk kedua algoritma ini.

Satu hal yang perlu dicatat, bahwa *tools* kedua algoritma ini dijalankan pada program MATLAB dan sangat mudah untuk diaplikasikan. ELM memiliki perintah yang jelas dan jauh lebih mudah untuk dipahami jika dibandingkan dengan BP. Karena yang perlu dilakukan hanyalah menentukan nama data *testing* dan *training*, jumlah *hidden neuron* yang digunakan serta *transfer functionnya*.

Sedangkan untuk BP, dibutuhkan beberapa perintah untuk memanggil fungsinya satu persatu. Detail mengenai parameter pengujian yang digunakan untuk BP ditampilkan pada Tabel 1.

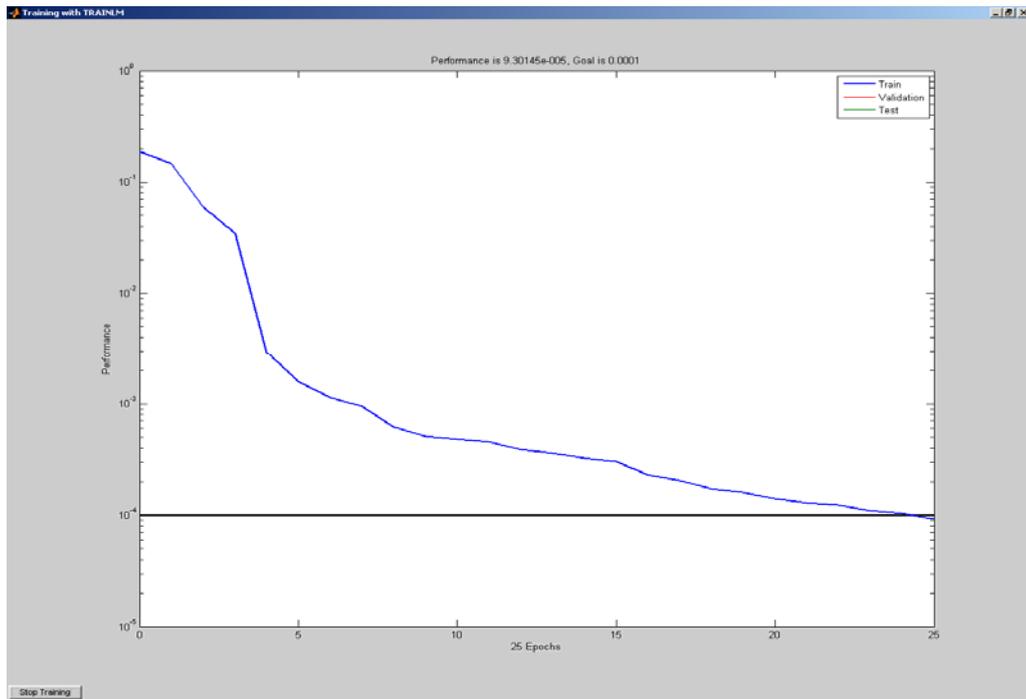
PEMBAHASAN

Tabel 1. Parameter pengujian dengan BP

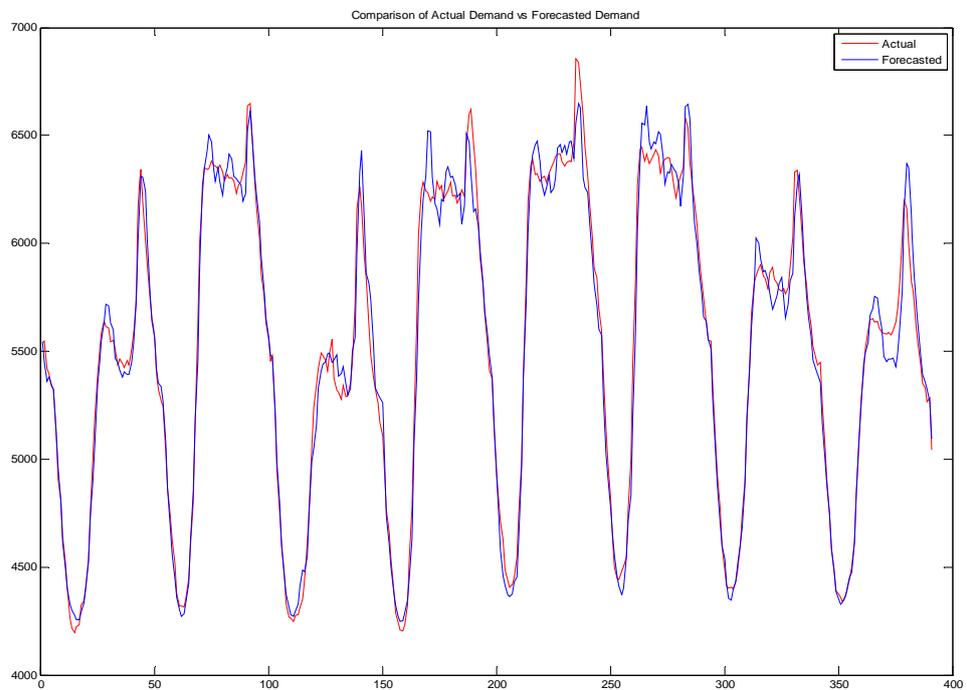
Hidden Neuron	=	2
Activation Function	=	Sigmoid , $Y^{sigmoid} = \frac{1}{1 + e^{-x}}$, for neural net this function is presented as logsig
Network training function	=	Trainlm, which updates weight and bias values according to Levenberg-Marquardt optimization
Stopping criteria	=	Goal = 10^{-4} , i.e. when the goal is reached then the model is converged.

Hasil simulasi dengan BP ditampilkan pada Tabel 1. Hasil simulasi yang terdiri atas proses *testing* hingga mencapai *goal* atau juga disebut *learning curve* ditampilkan pada Gambar 1 di bawah ini. *Learning curve* menunjukkan seberapa cepat suatu *network* dalam melakukan proses *learning*. Kasus *load forecasting network* dapat *terconverge*, diperlukan 25 *epoch* dengan *stopping*

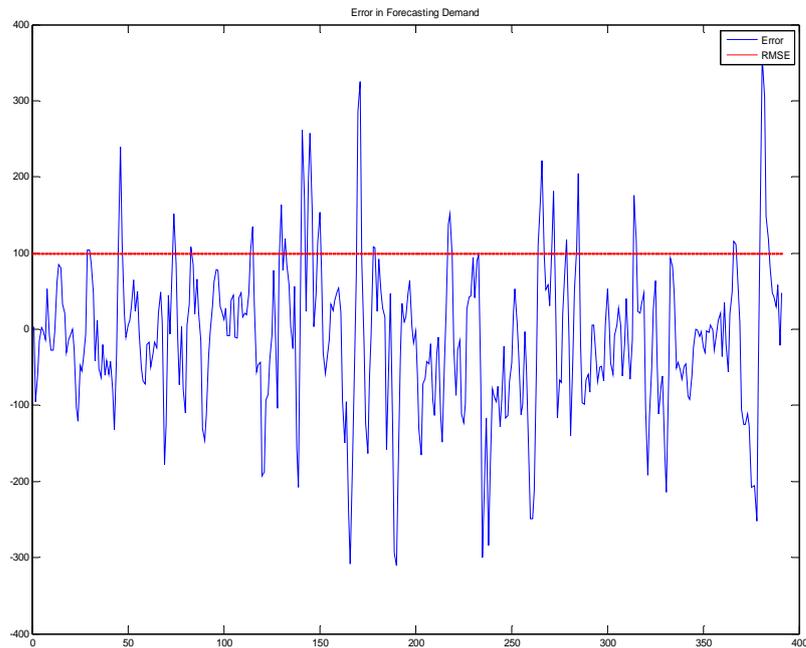
criteria (goal) 0.0001. Waktu yang diperlukan untuk *training* selama 77,4764 detik sedangkan *testing time*-nya selama 0.0288 detik dengan nilai RMSE ini sebesar 99.8721 MW. Pada Gambar 2 menampilkan hasil perbandingan antara *actual demand versus forecasted demand* sedangkan estimasi error dari hasil pengujian dengan BP ditampilkan pada Gambar 3.



Gambar 1 Hasil simulasi yang terdiri dari proses *testing* hingga mencapai *goal*



Gambar 2. *Actual Demand Versus Forecasted Demand* pada simulasi dengan metode *backpropagation* (dalam MW)



Gambar 3. Estimasi *error* pada simulasi dengan BP (dalam MW)

Simulasi dengan menggunakan metode ELM ini menerapkan parameter-parameter seperti yang ditampilkan pada Tabel 2. Pada kasus *load forecasting* ini, tipe ELM yang diaplikasikan adalah

regression yang dilambangkan dengan angka 0 pada *syntax*-nya. *Transfer function* yang diaplikasikan adalah Sigmoid seperti halnya pada *Back Propagation*.

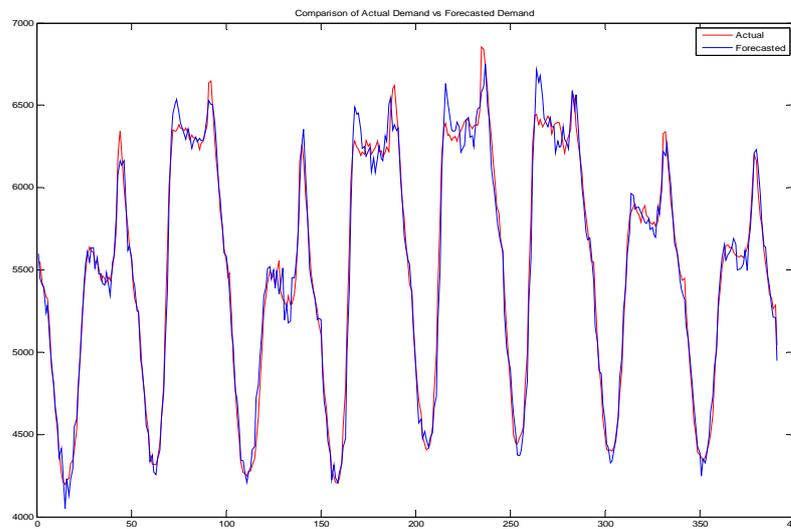
Tabel 2. Parameter pengujian dengan ELM

ELM type	:	0 (regression)
Number of hidden neuron	:	100
Transfer Function	:	Sigmoid, $Y^{sigmoid} = \frac{1}{1 + e^{-x}}$

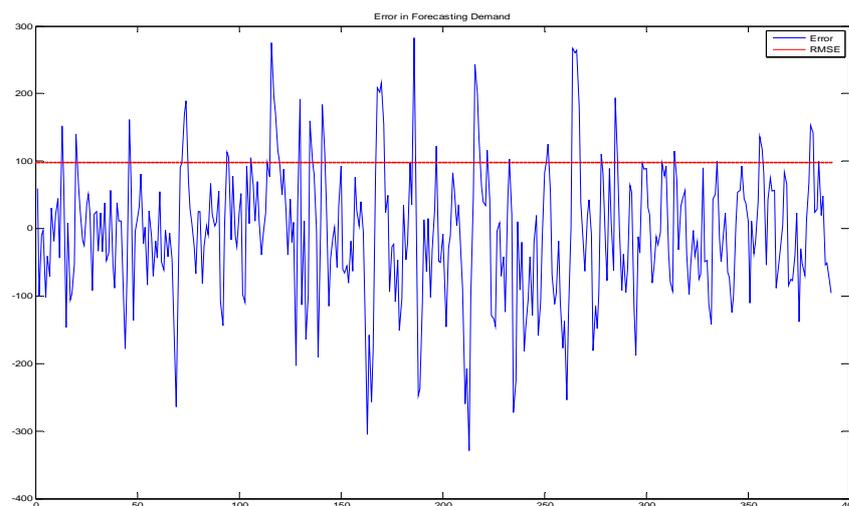
Simulasi ELM ini menunjukkan bahwa waktu *training* yang diperlukan adalah selama 0.0938 detik dan waktu *testing* selama 0.0156 detik. Akurasi, nilai RMSE-nya sebesar 75,2930 MW untuk *training* process-nya dan 98,4609 untuk *testing* proses-nya. Pada *testing data*, perbandingan antara *forecasted demand* dan *actual demand* ditampilkan pada Gambar 4 sedangkan estimasi *error* yang merupakan perbedaan antara

nilai aktual dan nilai *forecast*-nya ditunjukkan oleh Gambar 5.

Table 3 menunjukkan hasil perbandingan performansi antara BP dan ELM. Tampak bahwa baik ELM maupun BP menunjukkan performansi yang sangat baik. Namun demikian tampak bahwa kedua algoritma memiliki perbedaan yang cukup mendasar pada nilai RMSE yang berbeda serta waktu *training* dan *testing*-nya.



Gambar 4. *Actual demand Versus Forecasted Demand* untuk ELM (dalam MW)



Gambar 5. *Error pada load forecasting* dengan ELM (dalam MW)

Tabel 3. Perbandingan Performansi antara ELM dan BP

Algorithm	Time (Seconds)		RMSE (MW)	
	Training	Testing	Training	Testing
BP	77.4764	0.0288	Not calculated	99.8721
ELM	0.0938	0.0156	75.2930	98.4609

Seperti yang terlihat di Tabel 3, waktu konvergensi yang diperlukan oleh ELM jauh lebih cepat dibandingkan oleh BP. Dalam hal membangun model untuk proses *testing*-nya, ELM hanya membutuhkan waktu selama 0.0938 detik sedangkan BP membutuhkan ratusan de-

tik lebih lama. *Testing time* ELM ini juga dua kali lebih cepat dari pada BP ini. Namun demikian, dalam proses memprediksi *target testing*, RMSE yang diperoleh kedua algoritma ini hampir sama, yaitu 99.8721 MW untuk BP dan 98.4609 MW untuk ELM. Akan tetapi, untuk

memperoleh hasil yang hampir sama ini, ada beberapa parameter ini yang harus diterapkan, yaitu ELM perlu *single layer* dengan 100 *hidden neuron* sedangkan BP membutuhkan dua *layer* untuk *hidden neuron*-nya, yaitu 30 dan 1. Meskipun ELM membutuhkan *hidden neuron* yang lebih banyak, waktu komputasi tidaklah menjadi masalah sepanjang tersedia memori yang cukup besar untuk menyimpannya. Di sisi lain, BP dengan dua *hidden neuron* memberikan kecepatan konvergensi yang cukup lambat. Hal ini memperkuat fakta bahwa ELM dapat diaplikasikan pada daerah *Load forecasting* yang umumnya diperlukan memori yang besar terlebih jika banyak aspek yang menjadi pertimbangannya. Berdasarkan analisis tersebut, dapat dikatakan bahwa ELM memiliki performansi yang jauh lebih baik daripada BP dalam melakukan *demand* atau *load forecasting* untuk *short term period*.

KESIMPULAN

DAFTAR PUSTAKA

- Amjady, N., 2001 "Short-term Hourly Load Forecasting using Time Series Modelling," *Power Systems, IEEE Transactions on*, pp. 798-805.
- Anonim, 2007, "National Electricity Market Company" Website: http://www.nemweb.com.au/mms.GRAPHS/data/DATA200604_QLD1.csv. Download on 21 September
- Chen, B. J., Chang, M.W.; Lin, C.J, 2004, "Load Forecasting using Support Vector Machines: A Study on EUNITE Competition " *IEEE Transactions on Power Systems*, pp. 1821-1830,
- Espinoza, M.C., Joye, Belmans, R. and Moor, B.D, 2005, Shortterm Load Forecasting, Profile Identification, and Customer Segmentation: a Methodology Based on Periodic Time Series, *IEEE Transactions on Power Systems*, pp. 1622-1630,
- Han J. and Kamber M, 2001, *Data Mining: Concepts and Techniques*. San Fransisco: Morgan Kaufmann.
- Huang G.-B., Zhu Q.-Y., and Siew C.-K., 2006 "Extreme Learning Machine: A New Learning Scheme of Feedforward Neural Network," in *International Joint Conference on Neural Networks (IJCN2004)*, Budapest, Hungary.
- Huang, G.-B., Zhu, Q.-Y., and Siew, C.-K., 2006, "Extreme Learning Machine: Theory and Applications," *Neuro Computing*, vol. 70, pp. 489 - 501, 16 May 2006.
- Maia, C. A. and El-Keib, A.A, Ma and H. Ma, 1995, Advancement of Statistical based modelling techniques for short term load forecasting, *Electric Power Systems Research*, vol.35, pp.51-58.
- Saini, L. M., 2007 "peak Load Forecasting using Bayesian Regularization, Resilient, Adaptive Back-propagation Learning based Artificial Neural Networks," *Electric Power Systems Research*.