

Prediksi Nilai Tukar Rupiah Indonesia Terhadap Dolar Amerika Serikat Menggunakan Metode *Recurrent Extreme Learning Machine Neural Network*

Daneswara Jauhari¹, Imam Cholissodin², Candra Dewi³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹danesarajauhari@gmail.com, ²imamcs@ub.ac.id, ³dewi_candra@ub.ac.id

Abstrak

Nilai tukar uang oleh sebagian orang yang bekecimpung didalam perekonomian khususnya perekonomian antar negara sangatlah diperhatikan, seringkali mempengaruhi keputusan seseorang dalam mengambil sebuah kebijakan. Namun, nilai tukar merupakan nilai yang sangat tidak stabil, memiliki banyak *noise* dan fluktuatif, hal ini menyebabkan sangat sulit untuk memprediksi nilai tukar uang. Penelitian mengenai prediksi nilai tukar telah menjadi penelitian yang paling menantang dikalangan peneliti, serta dianggap sebagai salah satu bidang penelitian yang penting dalam keuangan internasional. Oleh karena itu, diperlukan sebuah aplikasi yang dapat melakukan prediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat dengan lebih baik, pada penelitian ini penulis menggunakan metode *Recurrent Extreme Learning Machine Neural Network* (RELMNN), metode tersebut dapat menangani *datasets* yang skuensial berdasarkan waktu dan dapat meningkatkan kemampuan metode *Extreme Learning Machine* (ELM) dalam melatih dan beradaptasi. Setelah dilakukan pengujian dengan parameter yang optimal, dan dilakukan pengujian perbandingan dengan metode ELM, didapatkan hasil bahwa metode RELMNN lebih unggul dari pada metode ELM dengan nilai *Mean Absolute Percentage Error* (MAPE) sebesar 0,069502%, sementara metode ELM mendapatkan nilai MAPE 0,090423%.

Kata Kunci: *Prediksi, Nilai Tukar Uang, Extreme Learning Machine, Recurrent Extreme Learning Machine Neural Network.*

Abstract

The exchange rate of money by some people who are involved in the economy, especially the inter-state economy is very payed, often influencing one's decision in taking a policy. However, the exchange rate is a very unstable value, has a lot of noise and fluctuation, it is very difficult to predict the exchange rate. Research on exchange rate prediction has become the most challenging research among researchers, and that is considered one of the most important areas of research in international finance. Therefore, an application is needed, which can better predict the exchange rate of Indonesian Rupiah against the US Dollar. In this study the authors use the method of Recurrent Extreme Learning Machine Neural Network (RELMNN), the method can handle time-ordered datasets and can improve the ability of the Extreme Learning Machine (ELM) method in training and adapting. After testing with optimum parameters, and compared with ELM method, we found out that RELMNN method is superior to ELM method with Mean Absolute Percentage Error (MAPE) value of 0.069502%, while ELM method get MAPE 0.090423%.

Keywords: *Prediction, Exchange Rate, Extreme Learning Machine, Recurrent Extreme Learning Machine Neural Network.*

1. PENDAHULUAN

Nilai tukar merupakan variabel penting dalam penelitian keuangan internasional, tidak hanya mempengaruhi keseimbangan internal

suatu perekonomian negara, tapi menentukan keseimbangan negara asing juga (Huang et al., 2011). Naik turunnya nilai tukar menjadi masalah besar bagi investor, karena dalam berinvestasi mereka membutuhkan hasil prediksi nilai tukar uang yang akurat (Wang et al., 2011).

Ketidakpastian akan besarnya pergerakan nilai tukar juga mempengaruhi arus kas operasi perusahaan, yakni arus kas masuk yang diterima dari ekspor atau dari anak perusahaan dan arus kas keluar yang dibutuhkan untuk pembayaran impor (Mata dan Pangeran, 2016). Selain itu, penurunan nilai mata uang suatu negara dapat menyebabkan krisis mata uang dalam suatu negara, penurunan ini akan mempengaruhi perekonomian dengan terjadinya ketidakstabilan pada nilai tukar.

Sistem keuangan di tandai dengan ketidakpastian yang tinggi, *non-linear*, dan perilaku yang bervariasi, sehingga membuat sangat sulit untuk meramalkan variabel keuangan seperti nilai tukar, indeks saham, dan inflasi (Liu et al., 2009). Nilai tukar menjadi salah satu penelitian yang paling menantang di kalangan peneliti (Rehman et al., 2014), dan dianggap sebagai salah satu bidang penelitian yang paling penting di bidang keuangan internasional, selama nilai tukar masih dipengaruhi atas dasar informasi yang di kumpulkan setiap waktu (Nayakovit et al., 2010). Hal ini karena nilai tukar bersifat *non-linear* dan *non-stasioner* (Sermpinis et al., 2015). Selain itu data yang terkait dengan nilai tukar uang memiliki banyak *noise*, tidak stabil dan fluktuatif (Rehman et al., 2014). Oleh karena itu, kemampuan memahami serta memprediksi sistem keuangan akan menghasilkan keuntungan yang signifikan, lingkungan ekonomi yang stabil, dan perencanaan yang cermat (Liu et al., 2009).

Banyak penelitian yang telah di publikasi mengenai prediksi nilai tukar uang, dan banyak dari penelitian tersebut yang menarik, seperti penelitian yang dilakukan oleh Rehman et al. (2014) yang melakukan prediksi nilai tukar uang dengan menggunakan metode *Cartesian Genetic Programming* dan *Recurrent Neural Network*, penelitian tersebut mendapatkan tingkat keakuratan 98,872%. Penelitian dengan metode lain dilakukan oleh Huang et al.(2011) juga melakukan penelitian untuk memprediksi nilai tukar, metode yang digunakan yaitu *Linear Moving Average (MA) Model* dan *non-linear Genetic Algorithm Backpropagation (GABP) Neural Network*, dengan menggunakan data uji didapatkan nilai *Mean Absolute Error (MAE)* sebesar 0,1208, *Mean Squared Error (MSE)* sebesar 0,0325, *Mean Absolute Percentage Error (MAPE)* sebesar 6,9615% dan *Root Mean Square Error (RMSE)* sebesar 0,1802.

Sedangkan penelitian mengenai metode *Extreme Learning Machine (ELM)* telah

dilakukan oleh Handika et al. (2016), untuk membandingkan antara metode ELM dengan metode *Particle Swarm Optimization Extreme Learning Machine (PSO-ELM)*, dalam meramalkan jumlah penjualan barang. Penelitian tersebut mendapatkan nilai MSE sekitar 0,01121 sampai 0,01161 untuk PSO-ELM, nilai tersebut lebih kecil dibandingkan nilai MSE dengan menggunakan ELM yaitu sekitar 0,01315 sampai 0,01419, namun metode PSO-ELM membutuhkan waktu proses yang jauh lebih lama jika di bandingkan dengan ELM. Penelitian lain telah dilakukan oleh Ertugrul (2016) untuk meramalkan beban listrik, dengan menggunakan metode *Recurrent Extreme Learning Machine (RELM)*, penelitian ini memberikan hasil nilai RMSE hampir dua kali lebih baik dari pada metode ELM, *Recurrent Neural Network (RNN)*, *Linear Regression (LR)*, *K-Smooth Regression (kSR)*, *K-Nearest Neighborhood Regression (kNNR)*, *Gaussian Process Regression (GPR)*, dan *Generalized Regression Neural Network (GRNN)*, dengan kecepatan yang hampir menyamai kecepatan ELM.

Oleh karena itu, perlu adanya penelitian untuk memprediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat dengan menggunakan metode *Recurrent Extreme Learning Machine Neural Network (RELMNN)*, diharapkan pada penelitian ini mendapat nilai evaluasi yang lebih baik, jika dibandingkan dengan metode yang pernah digunakan sebelumnya.

2. DATASETS

Data yang digunakan pada penelitian ini adalah data nilai tukar mulai dari tanggal 1 Januari 2009 sampai 11 Maret 2017. Sumber data yang digunakan dalam penelitian ini didapatkan dari website www.ofx.com, yang merupakan salah satu penyedia data nilai tukar uang.

3. RECURRENT EXTREME LEARNING MACHINE NEURAL NETWORK

Recurrent Extreme Learning Machine Neural Network (RELMNN) menambahkan mekanisme recurrent dalam jaringan ELM, maka satu atau lebih output akan dikembalikan ke sistem sebagai input baru. Tidak seperti jaringan yang lainnya, metode ini akan menghasilkan output jaringan yang tergantung dari nilai dari input sebelumnya (Rehman et al., 2014). RELMNN dibangun untuk menangani

datasets yang skuensial berdasarkan waktu, output yang dikembalikan ke sistem akan meningkatkan kemampuan jaringan dalam melatih dan beradaptasi (Ertugrul, 2016).

3.1. Proses Training

Tahapan-tahapan yang dilakukan ketika proses *training* dengan menggunakan metode RELMNN yaitu (Ertugrul, 2016):

1. Inisialisasi matriks *delay* δ dengan Persamaan (1).

$$\delta_{tr} = T(t - (n + r) + n) \quad (1)$$

Keterangan:

δ_{tr} = *delay* pada urutan data ke- t kolom ke- r

T = matriks target

n = jumlah *input neuron*

t = urutan data

2. Membuat nilai random untuk matriks $W'_{m(n+r)}$ sebagai bobot masukan dengan range $[-1,1]$, dalam bentuk *array* ukuran m (jumlah *hidden neuron*) $\times n+r$ (jumlah fitur + *context neurons*). Kemudian buat nilai random untuk matriks bias b dengan range $[0,1]$ dalam ukuran $1 \times$ (jumlah *hidden neuron*).
3. Menghitung nilai matriks keluaran pada *hidden layer* dengan menggunakan Persamaan (2). Perhitungan $b(\text{ones}(i, 1), :)$ akan memperbanyak matriks bias sebanyak jumlah data latih.

$$H' = \frac{1}{1 + \exp(-([x_{train}, \delta] W'^T + b(\text{ones}(i_{train}, 1), :)))} \quad (2)$$

Keterangan:

H' = matriks keluaran *hidden layer* pada proses *recurrent*

x_{train} = matriks input yang telah di normalisasi

δ = matriks *delay*

$[x_{train}, \delta]$ = matriks gabungan dari matriks x_{train} dan δ

W'^T = matriks *transpose* dari bobot pada proses *recurrent*

i_{train} = jumlah data latih

b = matriks bias

4. Menghitung $\hat{\beta}'$ sebagai bobot keluaran dengan menggunakan Persamaan (3), dimana H'^+ atau matriks *Moore-Penrose Pseudo Invers* dapat dihitung dengan Persamaan (4).

$$\hat{\beta}' = H'^+ T \quad (3)$$

$$H'^+ = (H'^T H')^{-1} H'^T \quad (4)$$

Keterangan:

$\hat{\beta}'$ = matriks bobot keluaran setelah

proses *recurrent*

H'^+ = matriks *Moore-Penrose Pseudo Invers* pada proses *recurrent*

T = matriks target

H' = matriks keluaran *hidden layer* pada proses *recurrent*

5. Menghitung hasil prediksi dengan menggunakan Persamaan (5).

$$\hat{Y}' = H' \hat{\beta}' \quad (5)$$

Keterangan:

\hat{Y}' = hasil prediksi pada proses *recurrent*

H' = keluaran *hidden layer* pada proses *recurrent*

$\hat{\beta}'$ = bobot keluaran pada proses *recurrent*

3.2. Proses Testing

Tahapan-tahapan yang dilakukan ketika proses *testing* dengan menggunakan metode RELMNN yaitu (Ertugrul, 2016):

1. Diketahui nilai $W'_{m(n+r)}$, b , dan $\hat{\beta}'$.
2. Inisialisasi nilai matriks *delay* δ' dengan melakukan perhitungan pada Persamaan (1).
3. Menghitung nilai matriks keluaran pada *hidden layer* dengan menggunakan Persamaan (6).

$$H' = \frac{1}{1 + \exp(-([x_{test}, \delta'] W'^T + b(\text{ones}(i_{test}, 1), :)))} \quad (6)$$

Keterangan:

H' = matriks keluaran *hidden layer* pada proses *recurrent*

x_{test} = matriks input yang telah di normalisasi

δ' = matriks *delay*

$[x_{test}, \delta']$ = matriks gabungan dari matriks x_{test} dan δ'

W'^T = matriks *transpose* dari bobot pada proses *recurrent*

i_{test} = jumlah data uji

b = matriks bias

4. Menghitung hasil prediksi dengan menggunakan Persamaan (5).

4. EVALUASI

Nilai *Mean Absolute Percentage Error* (MAPE) merupakan indikator yang sering digunakan dalam model prediksi (Worasuchep, 2015). MAPE digunakan ketika ukuran atau besar variabel prediksi dianggap penting dalam melakukan evaluasi hasil prediksi (Siwi, 2016). Persamaan (7) digunakan untuk menghitung nilai MAPE (Jauhari et al., 2016).

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|t_i - y_i|}{t_i} \times 100\% \quad (7)$$

Keterangan:

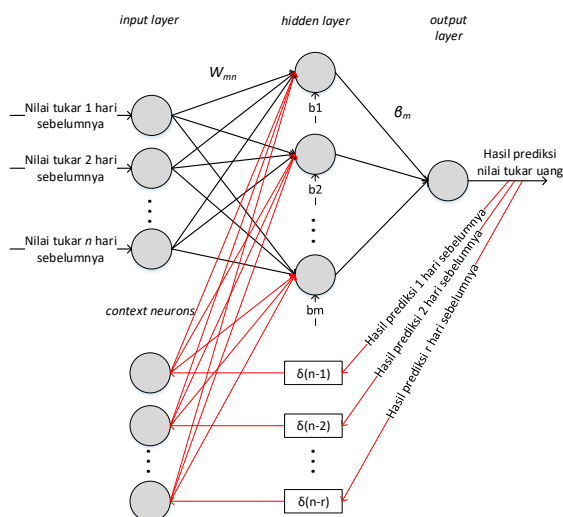
n = jumlah data

t_i = nilai aktual pada data ke- i

y_i = nilai hasil prediksi ke pada data ke- i

5. METODE

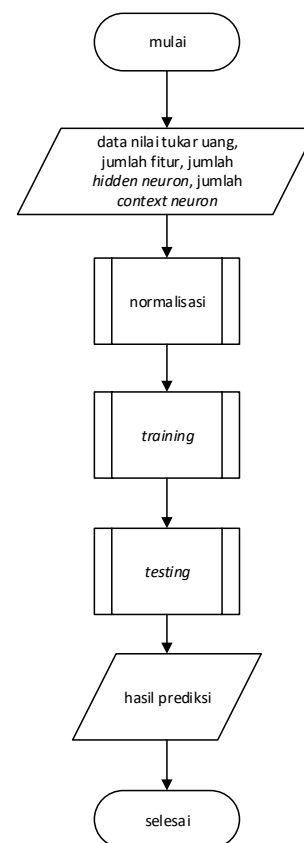
Arsitektur Jaringan yang digunakan pada penelitian ditunjukkan pada Gambar 2, dalam gambar tersebut terdapat beberapa komponen utama yaitu: *input layer*, *hidden layer*, *output layer* dan *context neuron*, dengan jumlah *neuron* sebanyak n untuk *input layer*, m untuk *hidden layer*, 1 untuk *output layer*, r untuk *context neuron*. Setiap *neuron* pada *input layer* terhubung dengan *hidden neuron* yang ada pada *hidden layer*. *Neuron* tersebut dihubungkan dengan *input weight* yang nilainya berbeda-beda. Masing-masing *neuron* yang ada pada *hidden neuron* juga terhubung dengan nilai bias yang nilainya berbeda-beda. *Neuron* yang ada pada *hidden neuron* terhubung dengan *output layer*. *Neuron* tersebut dihubungkan oleh *output weight*. Sedangkan nilai dari masing-masing *neuron* pada *context neuron* didapatkan dari nilai *delay* δ yang merupakan nilai tukar uang sebelumnya. Setiap *context neuron* terhubung dengan *hidden layer* melalui *input weight* yang nilainya berbeda-beda.



Gambar 1. Arsitektur Jaringan RELMNN

Proses dari prediksi nilai tukar uang melalui beberapa tahapan. Pertama menentukan jenis *datasets* nilai tukar dan memasukkan *datasets* tersebut kemudian melakukan normalisasi data, selanjutnya menentukan nilai parameter yang

dibutuhkan yaitu jumlah fitur, jumlah data uji, jumlah *hidden neuron*, dan jumlah *context neuron*, kemudian melakukan proses *training* dengan data latih yang didapatkan dari sebagian *datasets* nilai tukar yang telah dinormalisasi, dilanjutkan lagi dengan melakukan proses *testing* dengan data uji dan juga bobot masukan, bias dan bobot luaran dari proses *training*, didalam proses *testing* terdapat juga proses denormalisasi dan perhitungan nilai evaluasi, hingga menghasilkan hasil prediksi. Proses tersebut ditunjukkan pada Gambar 3.



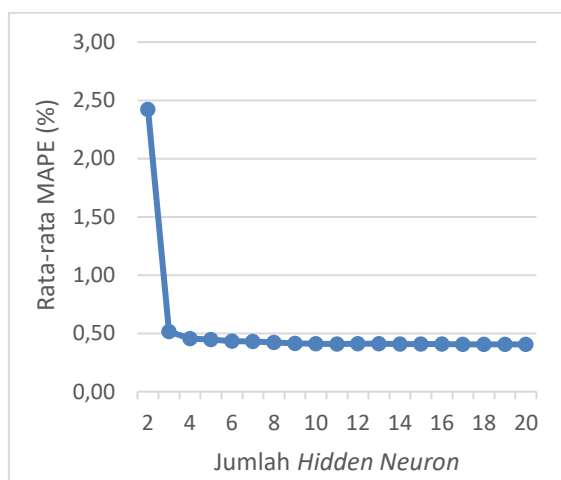
Gambar 2. Flowchart proses prediksi nilai tukar

6. HASIL DAN PEMBAHASAN

Pada bagian ini akan membahas mengenai hasil dan pembahasan dari beberapa pengujian yang telah dilakukan. Pengujian pada penelitian ini ada 4, yaitu pengujian pengaruh jumlah *hidden neuron* terhadap nilai MAPE, pengujian pengaruh jumlah fitur terhadap nilai MAPE, pengaruh jangka waktu prediksi terhadap nilai MAPE, pengaruh jumlah *context neurons* terhadap nilai MAPE dan perbandingan metode RELMNN dengan metode ELM.

6.1. Pengujian Jumlah *Hidden Neuron*

Pengujian berdasarkan *hidden neuron* digunakan untuk mengetahui jumlah *hidden neuron* yang memiliki nilai MAPE terbaik. Banyaknya *hidden neuron* yang di uji coba adalah dimulai dari 2 sampai 20. Inisialisasi parameter yang digunakan pada pengujian ini adalah bobot masukan dengan range $[-1,1]$, bias dengan range $[0,1]$, jumlah fitur sebanyak 4, jangka waktu 1 bulan yang terdiri dari 1890 data latih dan 30 data uji, dan jumlah *context neuron* sebanyak 2.



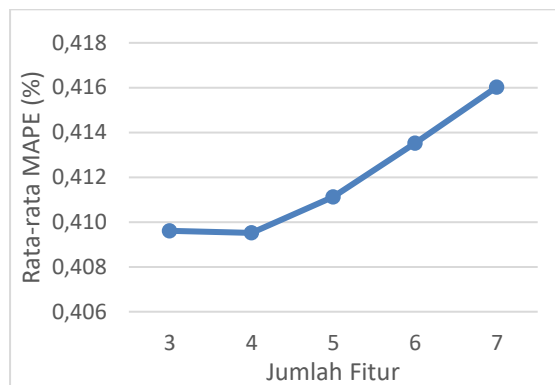
Gambar 3. Grafik Pengujian Jumlah *Hidden Neuron*

Grafik pada Gambar 3 merupakan grafik hasil dari pengujian jumlah *hidden neuron*. Berdasarkan grafik uji coba pada Gambar 3, terlihat bahwa nilai MAPE mulai stabil ketika jumlah *hidden neuron* berjumlah 4, untuk mendapatkan jumlah *hidden neuron* yang optimal maka diambil nilai tengah antara 4 dan 20 yaitu 12. Nilai *hidden neuron* yang semakin besar akan membentuk banyak penghubung antara *input layer* dan *output layer*. Kondisi tersebut membuat kemampuan dalam mengenali data semakin lebih baik karena semakin banyaknya pertimbangan keputusan yang bisa dilakukan *hidden node*.

6.2. Pengujian Jumlah Fitur

Pengujian berdasarkan jumlah fitur digunakan untuk mengetahui jumlah fitur yang memiliki nilai MAPE terbaik. Banyaknya jumlah fitur yang di uji coba adalah dimulai dari 3 sampai 7. Inisialisasi parameter yang digunakan pada pengujian ini adalah bobot masukan dengan range $[-1,1]$, bias dengan range $[0,1]$, jumlah *hidden neuron* menggunakan hasil

pengujian jumlah *hidden neuron* yang optimal, jangka waktu 1 bulan yang terdiri dari 1890 data latih dan 30 data uji, dan jumlah *context neuron* sebanyak 2.



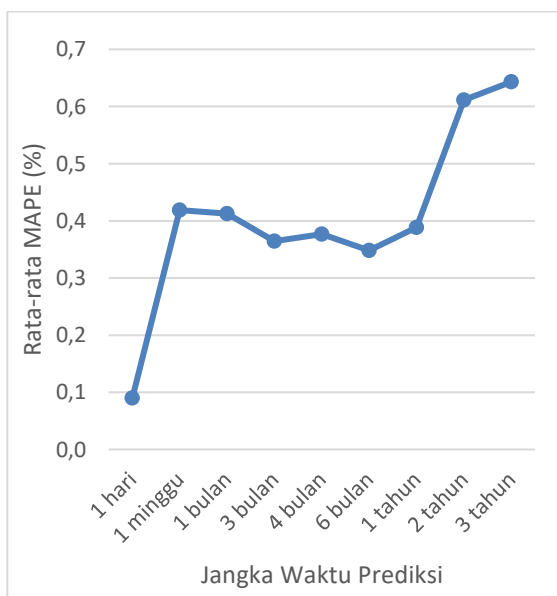
Gambar 4. Grafik Pengujian Jumlah Fitur

Grafik pada Gambar 4 merupakan grafik hasil dari pengujian jumlah fitur. Berdasarkan grafik uji coba pada Gambar 4, terlihat bahwa rata-rata nilai MAPE terkecil didapatkan ketika menggunakan 4 fitur. Hasil tersebut menunjukkan jumlah fitur yang sama dengan yang digunakan pada penelitian sebelumnya oleh Sumarto et al. (2016) dan Pramesti dan Mahmudy (2016). Grafik tersebut juga menunjukkan bahwa dengan fitur 5 sampai 7 nilai rata-rata MAPE semakin besar, yang artinya ketika jumlah fitur lebih dari 4 maka nilai tukar akan lebih sulit untuk diprediksi. Hal ini terjadi karena ketika fitur yang digunakan terlalu banyak, akan mengakibatkan jaringan kesulitan dalam mengenali pola data, memerlukan lebih banyak *hidden neuron*, dan memerlukan arsitektur jaringan yang lebih dinamis lagi, hal ini dikarenakan data yang menjadi fitur semakin banyak peluangnya dipengaruhi faktor luar yang tidak pasti kapan terjadinya, sedangkan ketika fitur yang digunakan terlalu sedikit, akan membuat jaringan kesulitan juga dalam mengenali pola data dikarenakan terlalu banyak pola yang sama.

6.3. Pengujian Jangka Waktu Prediksi

Pengujian berdasarkan jangka waktu prediksi digunakan untuk mengetahui akurasi prediksi pada jangka waktu tertentu. Terdapat 9 jangka waktu yang diuji, yaitu 1 hari, 1 minggu, 1 bulan, 3 bulan, 4 bulan, 6 bulan, 1 tahun, 2 tahun, dan 3 tahun. Inisialisasi parameter yang digunakan pada pengujian ini adalah bobot masukan dengan range $[-1,1]$, bias dengan range $[0,1]$, jumlah *hidden neuron* menggunakan hasil

pengujian jumlah *hidden neuron* yang optimal, jumlah fitur menggunakan hasil pengujian jumlah fitur yang optimal, jumlah *context neuron* sebanyak 2. Grafik pada Gambar 5 merupakan grafik hasil dari pengujian jangka waktu prediksi. Berdasarkan grafik uji coba pada Gambar 5, terlihat bahwa rata-rata nilai MAPE terkecil didapatkan ketika menggunakan jangka waktu 1 hari. Hal tersebut terjadi dikarenakan ketika memprediksi dengan jangka waktu 1 hari, jaringan dengan mudah dapat mengenali pola data dikarenakan jaringan telah banyak mengenali pola data dari data latih. Secara umum ketika melakukan prediksi dengan jangka waktu yang panjang, maka jaringan akan lebih kesulitan dalam mengenali pola-pola yang kemungkinan besar belum dikenali oleh jaringan, sedangkan terjadinya naik turun ketika melakukan prediksi 1 minggu hingga 1 tahun dikarenakan adanya perbedaan perbandingan banyaknya faktor luar yang terjadi setiap waktu dengan tidak adanya faktor luar yang mempengaruhi nilai tukar uang secara signifikan, misalnya ketika adanya bom yang terjadi di negara Indonesia atau Amerika Serikat, kondisi politik di Indonesia atau Amerika dan faktor-faktor yang lain, yang faktor-faktor tersebut sulit diprediksi kapan terjadinya dan seberapa banyak akan terjadi (Kornitasari, 2017).

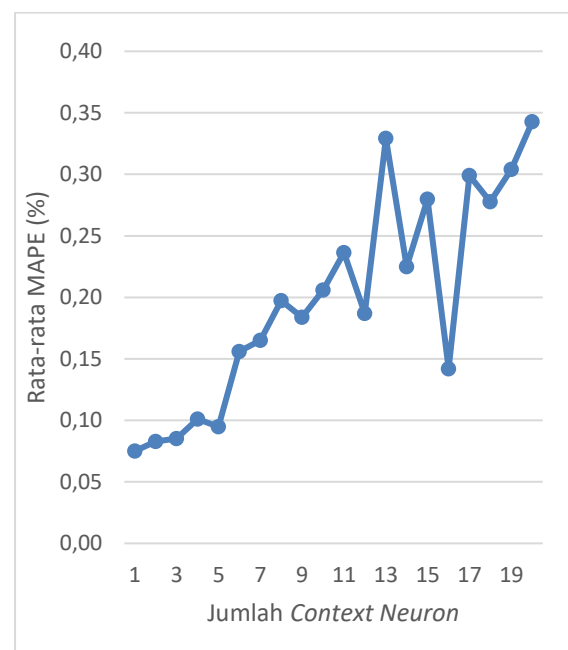


Gambar 5. Grafik Pengujian Jangka Waktu Prediksi

6.4. Pengujian Jumlah *Context Neuron*

Pengujian berdasarkan *context neuron* digunakan untuk mengetahui jumlah *context neuron* yang memiliki nilai MAPE terbaik.

Banyaknya *context neuron* yang di uji coba adalah dimulai dari 1 sampai 20. Inisialisasi parameter yang digunakan pada pengujian ini adalah bobot masukan dengan range $[-1,1]$, bias dengan range $[0,1]$, jumlah *hidden neuron* menggunakan hasil pengujian jumlah *hidden neuron* yang optimal, jumlah fitur menggunakan hasil pengujian jumlah fitur yang optimal dan jangka waktu dengan data latih dan data uji menggunakan hasil pengujian jangka waktu yang optimal.



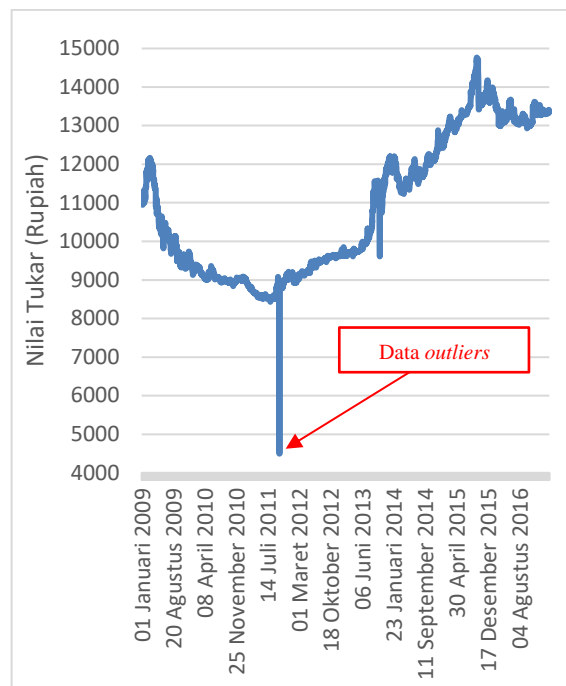
Gambar 6. Grafik Pengujian Jumlah *Context Neuron*

Grafik pada Gambar 6 merupakan grafik hasil dari pengujian jumlah *context neuron*. Berdasarkan grafik uji coba pada Gambar 6, terlihat bahwa rata-rata nilai MAPE terkecil didapatkan ketika menggunakan 1 *context neuron*, sedangkan jumlah *context neuron* lebih dari 1 menghasilkan nilai MAPE yang semakin besar dan tidak stabil, hal ini terjadi karena memang jumlah context neuron tergantung dari jumlah jangka waktu yang akan diprediksi. Sesuai dengan yang dikatakan Ertugrul (2016) bahwa jumlah *context neuron* akan meningkatkan kemampuan melatih dan beradaptasi dari jaringan, namun jumlah dari *context neuron* ini juga tergantung dari jumlah jangka waktu yang akan diprediksi. Pada pengujian ini dengan jangka waktu 1 hari didapatkan jumlah *context neuron* yang optimal adalah 1. Sama seperti apa yang didapatkan dari penelitian yang dilakukan Ertugrul (2016) yang menguji jumlah *context neuron* untuk

memprediksi 1 hari maka didapatkan jumlah *context neuron* yang optimal adalah 1, sementara ketika menguji jumlah *context neuron* untuk memprediksi 1 tahun maka didapatkan jumlah *context neuron* yang optimal sebanyak 3.

6.5. Pengujian Perbandingan Metode

Pengujian perbandingan metode RELMNN dengan metode ELM digunakan untuk mengetahui metode terbaik dalam memprediksi nilai tukar. Inisialisasi parameter yang digunakan pada pengujian ini adalah bobot masukan dengan range $[-1,1]$, bias dengan range $[0,1]$, jumlah *hidden neuron* menggunakan hasil pengujian jumlah *hidden neuron* yang optimal, jumlah fitur menggunakan hasil pengujian jumlah fitur yang optimal, jangka waktu menggunakan hasil pengujian jangka waktu yang optimal dan pada metode RELMNN jumlah *context neuron* yang digunakan merupakan jumlah *context neuron* pada hasil pengujian *context neuron* yang optimal. Hasil uji coba perbandingan metode RELMNN dan metode ELM dengan jenis *datasets* yang berbeda terhadap nilai MAPE dapat dilihat pada Tabel 5. Hasil uji coba perbandingan metode RELMNN dengan metode ELM terhadap selisih nilai tukar Rupiah Indonesia terhadap Dolar Amerika menggunakan *datasets* tanpa *outliers* dapat dilihat pada Tabel 6. Sedangkan, hasil uji coba perbandingan metode RELMNN dengan metode ELM terhadap waktu yang dibutuhkan pada proses *training* dan *testing* menggunakan *datasets* tanpa *outliers* dapat dilihat pada Tabel 7. *Datasets outliers* pada penelitian ini didapatkan dengan cara manual, yaitu dengan melihat grafik keseluruhan *datasets*. Grafik dari keseluruhan *datasets* dapat dilihat pada Gambar 7.



Gambar 7. Grafik *Datasets* Terdapat *Outliers*

Tabel 5. Hasil Pengujian Perbandingan Metode Terhadap Nilai MAPE

Metode (Jenis <i>Datasets</i>)	Rata-rata MAPE (%)
RELMNN (<i>datasets</i> terdapat <i>outliers</i>)	0,128711
ELM (<i>datasets</i> terdapat <i>outliers</i>)	0,122319
RELMNN (<i>datasets</i> tanpa <i>outliers</i>)	0,069502
ELM (<i>datasets</i> tanpa <i>outliers</i>)	0,090423

Tabel 6. Hasil Pengujian Perbandingan Metode Terhadap Selisih Nilai Tukar Rupiah

Metode	Rata-rata Selisih (IDR)	Pembulatan (IDR)
RELMNN	7,963540	8
ELM	10,360598	10

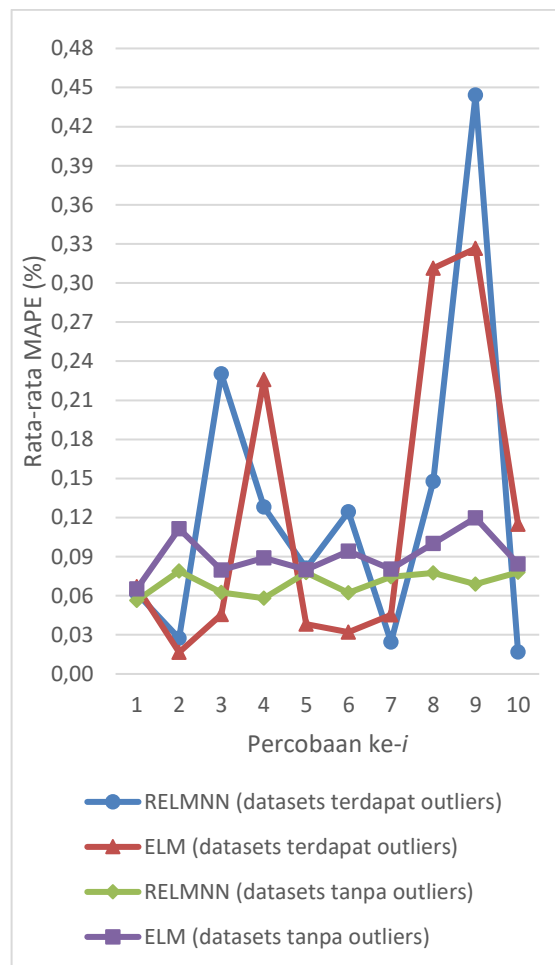
Tabel 7. Hasil Pengujian Perbandingan Metode Terhadap Waktu yang Digunakan

Metode	Rata-rata Waktu (detik)
RELMNN (<i>training</i>)	0,015907
ELM (<i>training</i>)	0,014400
RELMNN (<i>testing</i>)	0,001359
ELM (<i>testing</i>)	0,001849

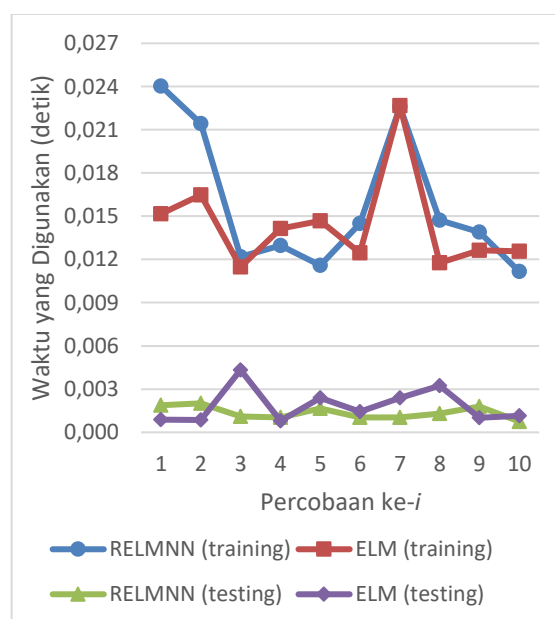
Grafik pada Gambar 8 dibuat untuk mempermudah dalam menganalisis metode terbaik dengan nilai MAPE terkecil berdasarkan

Tabel 5. Berdasarkan grafik uji coba pada Gambar 8, terlihat bahwa pada setiap percobaan perbandingan metode dengan jenis *datasets* tanpa *outliers* nilai MAPE terkecil didapatkan ketika menggunakan metode RELMNN, jika dibandingkan dengan metode ELM dengan jenis *datasets* tanpa *outliers* perbandingannya sebesar 0,069502:0,090423 (RELMNN:ELM). Hasil tersebut membuktikan bahwa memang benar adanya proses *recurrent* pada jaringan akan meningkatkan kemampuan melatih dan beradaptasi dari jaringan. Sementara itu ketika kedua metode menggunakan *datasets outliers* didapatkan nilai MAPE yang tidak stabil, ELM lebih unggul sedikit dari pada RELMNN. Hasil tersebut membuktikan bahwa kedua metode yang digunakan masih belum bisa menangani data yang *outliers*. Berdasarkan Tabel 6 yang didapatkan dengan menggunakan hasil perbandingan metode terhadap nilai MAPE dengan jenis *datasets* tanpa *outliers*, terlihat bahwa prediksi nilai tukar dengan metode RELMNN mendapatkan rata-rata selisih antara nilai target dengan nilai prediksi mencapai 8 rupiah, sementara ELM mencapai 10 rupiah.

Grafik pada Gambar 9 dibuat untuk mempermudah dalam menganalisis metode terbaik dengan waktu yang digunakan paling sedikit berdasarkan Tabel 7. Berdasarkan grafik uji coba pada Gambar 9, terlihat bahwa waktu yang digunakan metode ELM pada proses *training* rata-rata lebih unggul dari pada metode RELMNN hal ini memang wajar karena metode RELMNN terdapat tambahan proses *recurrent*. Sedangkan pada proses *testing* metode RELMNN dan ELM terlihat membutuhkan waktu yang hampir sama, hal ini dikarenakan memang proses yang dijalankan pada proses *testing* antara kedua metode memang sama.



Gambar 8. Grafik Pengujian Perbandingan Metode Terhadap Nilai MAPE



Gambar 9. Grafik Pengujian Perbandingan Metode Terhadap Waktu yang Digunakan

7. KESIMPULAN

Berdasarkan hasil pengujian dari prediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika menggunakan metode *Recurrent Extreme Learning Machine Neural Network* maka diperoleh kesimpulan bahwa nilai evaluasi berupa rata-rata nilai MAPE terbaik yang didapatkan dengan menggunakan metode RELMNN sebesar 0,069502%. Nilai MAPE tersebut didapatkan dengan menggunakan 12 *hidden neuron*, 4 fitur, jangka waktu 1 hari dan 1 *context neuron*. Berdasarkan penelitian yang telah dilakukan, didapatkan beberapa saran yang dapat dikembangkan untuk penelitian selanjutnya yaitu dengan menambahkan proses khusus untuk menangani *datasets outliers* secara otomatis, sehingga hasil dari prediksi bisa tetap stabil walaupun terdapat *datasets outliers*.

DAFTAR PUSTAKA

- Ertugrul, Ö.F., 2016. Forecasting electricity load by a novel recurrent extreme learning machines approach. *International Journal of Electrical Power and Energy Systems*, 78, pp.429–435.
- Handika, I.P.S., Giriantari, I.A. dan Dharma, A., 2016. Perbandingan Metode Extreme Learning Machine dan Particle Swarm Optimization Extreme Learning Machine untuk Peramalan Jumlah Penjualan Barang. 15(1), pp.84–90.
- Huang, Z., Zheng, G. dan Jia, Y., 2011. Forecasting exchange rate volatility with linear MA model and nonlinear GABP neural network. *Proceedings - 2011 4th International Conference on Business Intelligence and Financial Engineering, BIFE 2011*, pp.22–26.
- Jauhari, D., Hanafi, A., Y, M.F.A., Satria, A.R. dan H, L.H., 2016. Prediksi Nilai Tukar Rupiah Terhadap US Dollar Menggunakan Metode Genetic Programming. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, 3(4), pp.285–291.
- Liu, Z., Zheng, Z., Liu, X. dan Wang, G., 2009. Modelling and Prediction of the CNY Exchange Rate Using RBF Neural Network. *2009 International Conference on Business Intelligence and Financial Engineering*, (2), pp.38–41.
- Mata, V.W.T. dan Pangeran, P., 2016. Eksposur Nilai Tukar Dolar (US\$) Pada Perusahaan Multinasional di Indonesia. *KOMPETENSI - JURNAL MANAJEMEN BISNIS*, 11(2), pp.79–94.
- Nayakovit, S., Jaritngam, U. dan Khantanapoka, K., 2010. Prediction exchange rate of USD/GBP with intelligence cyberspace experimental. *ICEIE 2010 - 2010 International Conference on Electronics and Information Engineering, Proceedings*, 2(Iceie), pp.15–19.
- Pramesti, R.A. dan Mahmudy, W.F., 2016. Optimasi Fuzzy Inference System Mamdani Untuk Memprediksi Nilai Tukar Rupiah Terhadap Dollar Amerika Menggunakan Algoritma Genetika. Universitas Brawijaya.
- Rehman, M., Khan, G.M. dan Mahmud, S.A., 2014. Foreign Currency Exchange Rates Prediction Using CGP and Recurrent Neural Network. *IERI Procedia*, 10, pp.239–244.
- Sermpinis, G., Stasinakis, C., Theofilatos, K. dan Karathanasopoulos, A., 2015. Modeling, forecasting and trading the EUR exchange rates with hybrid rolling genetic algorithms - Support vector regression forecast combinations. *European Journal of Operational Research*, [online] 247(3), pp.831–846. Available at: <<http://dx.doi.org/10.1016/j.ejor.2015.06.052>>.
- Siwi, I.P., 2016. Peramalan Produksi Gula Pasir Menggunakan Extreme Learning Machine (ELM) Pada PG Candi Baru Sidoarjo. Universitas Brawijaya.
- Sumarto, K.C., Cholissodin, I. dan Data, M., 2016. Peramalan Nilai Tukar Rupiah (IDR) Terhadap Dollar Amerika (USD) Menggunakan Metode Support Vector Regression (SVR) dengan Simulated Annealing - Genetic Algorithm (SA-GA). Universitas Brawijaya.
- Wang, T.C., Kuo, S.H. dan Chen, H.C., 2011. Forecasting the exchange rate between ASEAN currencies and USD. *IEEE International Conference on Industrial Engineering and Engineering Management*, pp.699–703.
- Worasuchep, C., 2015. Forecasting currency

exchange rates with an artificial bee colony-optimized neural network. *2015 IEEE Congress on Evolutionary Computation, CEC 2015 - Proceedings*, 0(c), pp.3319–3326.