

**PREDIKSI NILAI TUKAR RUPIAH INDONESIA TERHADAP  
DOLAR AMERIKA SERIKAT MENGGUNAKAN METODE  
*RECURRENT EXTREME LEARNING MACHINE NEURAL  
NETWORK***

**SKRIPSI**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:  
Daneswara Jauhari  
NIM: 135150200111002



PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2017

# PENGESAHAN

PREDIKSI NILAI TUKAR RUPIAH INDONESIA TERHADAP DOLAR AMERIKA SERIKAT  
MENGUNAKAN METODE *RECURRENT EXTREME LEARNING MACHINE NEURAL  
NETWORK*

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh :  
Daneswara Jauhari  
NIM: 135150200111002

Skripsi ini telah diuji dan dinyatakan lulus pada  
6 Juli 2017

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Imam Cholissodin, S.Si, M.Kom  
NIK: 201201 850719 1 001

Candra Dewi, S.Kom, M.Sc  
NIP: 19771114 200312 2 001

Mengetahui  
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D  
NIP: 19710518 200312 1 001

## **PERNYATAAN ORISINALITAS**

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 6 Juli 2017

Daneswara Jauhari

NIM: 135150200111002

## KATA PENGANTAR

Syukur Alhamdulillah penulis panjatkan kehadiran Allah SWT, yang telah senantiasa melimpahkan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Prediksi Nilai Tukar Rupiah Indonesia Terhadap Dolar Amerika Serikat Menggunakan Metode *Recurrent Extreme Learning Machine Neural Network*”

Shalawat serta salam semoga tetap tercurahkan kepada junjungan kita, Nabi kita Rasulullah SAW, beserta keluarganya, sahabatnya, Tabi’in, Tabiut’ Tabi’in dan juga para Ulama yang senantiasa mengikuti jalan dakwahnya. Semoga kelak di akhirat nanti kita mendapatkan syafa’atnya. Aamiin.

Dalam penulisan skripsi ini, penulis menyadari tidak terlepas dari berbagai kesulitan dalam menyelesaikannya. Namun, Alhamdulillah Robbil `Alamin berkat kemudahan yang diberikan Allah SWT lewat orang-orang yang dicintai-Nya, akhirnya skripsi ini dapat diselesaikan dengan baik. Oleh karena itu, penulis ingin mengucapkan terimakasih banyak kepada:

1. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D. selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya.
2. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya.
3. Bapak Agus Wahyu Widodo, S.T, M.Cs selaku Ketua Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya.
4. Bapak Imam Cholissodin, S.Si, M.Kom dan Ibu Candra Dewi, S.Kom, M.Sc selaku dosen pembimbing skripsi yang telah meluangkan waktu dan dengan sabar membimbing dan juga mengarahkan penulis sehingga dapat menyelesaikan skripsi ini.
5. Seluruh dosen Fakultas Ilmu Komputer Universitas Brawijaya atas ketersediaannya membagi ilmu kepada penulis selama masa perkuliahan.
6. Teman-teman LKI Al-Fatih Muslim Drenalin, Masbie dan Himpunan Mahasiswa Informatika yang telah banyak membantu penulis dan memberikan dukungan dalam menyelesaikan skripsi ini.
7. Teristimewa kepada kedua orang tua penulis yang telah membesarkan dan memperjuangkan penulis dengan selalu memberikan doa, dukungan, semangat yang tiada henti hingga penulis dapat menyelesaikan studi diperguruan tinggi negeri. Penulis mempersembahkan skripsi ini sebagai salah satu perwujudan rasa tanggung jawab dan terima kasih atas cinta dan pengorbanan yang selama ini telah diberikan kepada penulis.
8. Semua pihak yang telah membantu terselesainya skripsi ini yang tidak dapat penulis sebut satu persatu.

Atas jasa mereka, peneliti hanya dapat memohon doa semoga amal mereka mendapatkan balasan yang lebih baik serta kesuksesan baik di dunia maupun di akhirat. Aamiin. Penulis menyadari bahwa skripsi ini tidak lepas dari kekurangan

dan kesalahan. Oleh karena itu, peneliti dalam hal ini juga mengharapkan kritik dan saran yang membangun dari pembaca untuk perbaikan kedepannya. Akhir kata, semoga skripsi ini dapat memberikan manfaat bagi semua pihak yang terkait.

Malang, 6 Juli 2017

Penulis

[danesarajauhari@gmail.com](mailto:danesarajauhari@gmail.com)

## ABSTRAK

Nilai tukar uang oleh sebagian orang yang bekecimpung di dalam perekonomian khususnya perekonomian antar negara sangat 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%. Sementara waktu yang dibutuhkan RELMNN pada proses *training* tidak jauh berbeda dari metode ELM, sedangkan waktu yang dibutuhkan antara kedua metode dalam proses *testing* hampir sama. Nilai tersebut dicapai dengan menggunakan 12 *hidden neuron*, 4 fitur, jangka waktu 1 hari, dan 1 *context neuron*.

**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%. While the time required by RELMNN in the training process is not much different from the ELM method, while the time required between the two methods in the testing process is almost the same. The value is achieved by using 12 hidden neurons, 4 features, a period of 1 day, and 1 context neuron.*

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

## DAFTAR ISI

PENGESAHAN .....	ii
PERNYATAAN ORISINALITAS .....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT .....	vii
DAFTAR ISI .....	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR .....	xiv
DAFTAR LAMPIRAN .....	xvi
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah .....	2
1.3 Tujuan .....	3
1.4 Manfaat.....	3
1.5 Batasan masalah .....	3
1.6 Sistematika pembahasan .....	3
BAB 2 LANDASAN KEPUSTAKAAN .....	5
2.1 Kajian Pustaka .....	5
2.2 Prediksi.....	8
2.3 Nilai Tukar Uang.....	8
2.4 Normalisasi dan Denormalisasi Data .....	10
2.5 Jaringan Syaraf Tiruan.....	10
2.5.1 Faktor Bobot.....	11
2.5.2 Fungsi Aktivasi.....	11
2.6 <i>Extreme Learning Machine</i> .....	11
2.6.1 Arsitektur Jaringan ELM .....	12
2.6.2 Proses <i>Training</i> .....	12
2.6.3 Proses <i>Testing</i> .....	14
2.7 Operasi Baris Elementer .....	14
2.8 <i>Recurrent Extreme Learning Machine Neural Network (RELMNN)</i> .....	15



2.8.1 Arsitektur Jaringan RELMNN .....	15
2.8.2 Proses <i>Training</i> .....	16
2.8.3 Proses <i>Testing</i> .....	18
2.9 Nilai Evaluasi .....	18
BAB 3 METODOLOGI .....	20
3.1 Tahapan Penelitian .....	20
3.2 Studi Literatur .....	20
3.3 Pengumpulan Data .....	21
3.4 Analisa Kebutuhan .....	21
3.5 Perancangan .....	21
3.6 Implementasi .....	22
3.7 Pengujian dan Analisis .....	22
3.8 Kesimpulan.....	22
BAB 4 PERANCANGAN.....	23
4.1 Formulasi Permasalahan.....	23
4.2 Perancangan Jaringan .....	24
4.3 Diagram Alir Program .....	25
4.3.1 Normalisasi.....	26
4.3.2 <i>Training</i> .....	29
4.3.3 <i>Testing</i> .....	50
4.4 Perhitungan Manual .....	53
4.5 Perancangan <i>User Interface</i> .....	64
4.5.1 Perancangan <i>Form Datasets</i> .....	65
4.5.2 Perancangan <i>Form Training</i> .....	65
4.5.3 Perancangan <i>Form Testing</i> .....	66
4.6 Perancangan Uji Coba dan Evaluasi.....	67
4.6.1 Pengujian Jumlah <i>Hidden Neuron</i> .....	68
4.6.2 Pengujian Jumlah Fitur.....	69
4.6.3 Pengujian Jangka Waktu Prediksi.....	69
4.6.4 Pengujian Jumlah <i>Context Neuron</i> .....	70
4.6.5 Pengujian Perbandingan Metode RELMNN dengan Metode ELM .....	71

BAB 5 IMPLEMENTASI .....	73
5.1 Implementasi Program .....	73
5.1.1 Implementasi Proses Normalisasi Data .....	73
5.1.2 Implementasi Proses Inisialisasi Matriks <i>Delay</i> .....	74
5.1.3 Implementasi Proses Membuat Nilai Random Bobot dan Bias ..	75
5.1.4 Implementasi Proses Menghitung Keluaran pada <i>Hidden Layer</i>	76
5.1.5 Implementasi Proses Menggabungkan Matriks Input dengan Matriks <i>Delay</i> .....	77
5.1.6 Implementasi Proses Memperbanyak Matriks Bias .....	77
5.1.7 Implementasi Proses <i>Transpose</i> Matriks .....	78
5.1.8 Implementasi Proses Perkalian Matriks.....	78
5.1.9 Implementasi Proses Penjumlahan Matriks .....	79
5.1.10 Implementasi Proses Menghitung Fungsi Aktivasi .....	79
5.1.11 Implementasi Proses Menghitung Bobot Keluaran .....	80
5.1.12 Implementasi Proses Menghitung Matriks <i>Moore-Penrose Pseudo Invers</i> .....	80
5.1.13 Implementasi Proses Menghitung <i>Invers</i> Matriks .....	81
5.1.14 Implementasi Proses Menghitung Hasil Prediksi.....	82
5.1.15 Implementasi Proses Denormalisasi .....	82
5.1.16 Implementasi Proses Menghitung Nilai Evaluasi .....	83
5.2 Implementasi <i>User Interface</i> .....	83
5.2.1 Implementasi <i>Form Datasets</i> .....	84
5.2.2 Implementasi <i>Form Training</i> .....	84
5.2.3 Implementasi <i>Form Testing</i> .....	85
BAB 6 PENGUJIAN DAN ANALISIS.....	87
6.1 Hasil dan Analisis Uji Coba Jumlah <i>Hidden Neuron</i> .....	87
6.2 Hasil dan Analisis Uji Coba Jumlah Fitur .....	88
6.3 Hasil dan Analisis Uji Coba Jangka Waktu Prediksi .....	90
6.4 Hasil dan Analisis Uji Coba Jumlah <i>Context Neuron</i> .....	91
6.5 Hasil dan Analisa Uji Coba Perbandingan Metode RELMNN dengan Metode ELM .....	93
BAB 7 PENUTUP .....	97
7.1 Kesimpulan.....	97

7.2 Saran .....	98
DAFTAR PUSTAKA.....	99
LAMPIRAN A DATA NILAI TUKAR RUPIAH INDONESIA TERHADAP DOLAR AMERIKA SERIKAT .....	102
LAMPIRAN B HASIL UJI COBA .....	128
LAMPIRAN C VISUALISASI HASIL UJI COBA .....	133

## DAFTAR TABEL

Tabel 2.1 Perbandingan dengan penelitian sebelumnya.....	6
Tabel 4.1 Data nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat .....	23
Tabel 4.2 Nilai <i>Maximum</i> dan <i>Minimum</i> .....	53
Tabel 4.3 Normalisasi <i>Datasets</i> .....	54
Tabel 4.4 Matriks <i>Delay</i> Proses <i>Training</i> .....	54
Tabel 4.5 Matriks Bobot .....	55
Tabel 4.6 Matriks Bias .....	55
Tabel 4.7 Gabungan Matriks <i>X</i> Dengan Matriks <i>Delay</i> Proses <i>Training</i> .....	55
Tabel 4.8 Matriks Bias Proses <i>Training</i> .....	56
Tabel 4.9 Matriks Hasil Perkalian .....	57
Tabel 4.10 Matriks Hasil Penjumlahan.....	57
Tabel 4.11 Matriks keluaran <i>hidden layer</i> .....	58
Tabel 4.12 Matriks Hasil Perkalian .....	58
Tabel 4.13 Matriks <i>Invers</i> .....	59
Tabel 4.14 Matriks <i>Moore-Penrose Pseudo Invers</i> .....	59
Tabel 4.15 Matriks Bobot Keluaran.....	60
Tabel 4.16 Matriks Bobot Keluaran.....	60
Tabel 4.17 Matriks <i>Delay</i> Proses <i>Testing</i> .....	61
Tabel 4.18 Gabungan Matriks <i>X</i> Dengan Matriks <i>Delay</i> Proses <i>Testing</i> .....	61
Tabel 4.19 Matriks Bias Proses <i>Testing</i> .....	62
Tabel 4.20 Matriks Hasil Perkalian .....	62
Tabel 4.21 Matriks Hasil Penjumlahan.....	62
Tabel 4.22 Matriks keluaran <i>hidden layer</i> .....	63
Tabel 4.23 Matriks Bobot Keluaran.....	63
Tabel 4.24 Matriks Denormalisasi.....	64
Tabel 4.25 Rancangan Pengujian <i>Hidden Neuron</i> .....	68
Tabel 4.26 Rancangan Pengujian Jumlah Fitur .....	69
Tabel 4.27 Rancangan Pengujian Jangka Waktu Prediksi .....	70
Tabel 4.28 Rancangan Pengujian <i>Context Neuron</i> .....	70
Tabel 4.29 Rancangan Pengujian Perbandingan Metode Terhadap Nilai MAPE..	72

Tabel 4.30 Rancangan Pengujian Perbandingan Metode Terhadap Selisih Nilai Tukar Rupiah Indonesia terhadap Dolar Amerika.....	72
Tabel 4.31 Rancangan Pengujian Perbandingan Metode Terhadap Waktu yang Digunakan.....	72
Tabel 6.1 Hasil Uji Coba Jumlah <i>Hidden Neuron</i> .....	87
Tabel 6.2 Hasil Uji Coba Jumlah <i>Fitur</i> .....	89
Tabel 6.3 Hasil Uji Coba Jangka Waktu Prediksi.....	90
Tabel 6.4 Hasil Uji Coba Jumlah <i>Context Neuron</i> .....	92
Tabel 6.5 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Nilai MAPE .....	94
Tabel 6.6 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Selisih Nilai Tukar Rupiah Indonesia terhadap Dolar Amerika .....	94
Tabel 6.7 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Waktu yang Dibutuhkan Pada Proses <i>Training</i> dan <i>Testing</i> .....	94

## DAFTAR GAMBAR

Gambar 2.1 Jaringan <i>Extreme Learning Machine</i> .....	12
Gambar 2.2 Jaringan <i>Recurrent Extreme Learning Machine Neural Network</i> .....	16
Gambar 3.1 Tahapan-tahapan penelitian .....	20
Gambar 4.1 Arsitektur Jaringan .....	25
Gambar 4.2 Diagram Alir Program .....	26
Gambar 4.3 Diagram Alir Proses Normalisasi .....	28
Gambar 4.4 Diagram Alir Proses <i>Training</i> .....	29
Gambar 4.5 Diagram Alir Proses Inisialisasi Matriks <i>Delay</i> .....	31
Gambar 4.6 Diagram Alir Proses Membuat Nilai Random Untuk Bobot Masukan dan Bias .....	32
Gambar 4.7 Diagram Alir Proses Menghitung Matriks Keluaran Pada Hidden Layer .....	33
Gambar 4.8 Diagram Alir Proses Menggabungkan Matriks Input Dengan Matriks <i>Delay</i> .....	35
Gambar 4.9 Diagram Alir Proses Memperbanyak Matriks Bias.....	36
Gambar 4.10 Diagram Alir Proses Melakukan <i>Transpose</i> Matriks .....	37
Gambar 4.11 Diagram Alir Proses Menghitung Perkalian Matriks .....	39
Gambar 4.12 Diagram Alir Proses Menghitung Penjumlahan Matriks.....	40
Gambar 4.13 Diagram Alir Proses Menghitung Fungsi Aktivasi .....	41
Gambar 4.14 Diagram Alir Proses Menghitung Bobot Keluaran .....	42
Gambar 4.15 Diagram Alir Proses Menghitung Matriks <i>Moore-Penrose Pseudo Invers</i> .....	43
Gambar 4.16 Diagram Alir Proses Menghitung <i>Invers</i> Matriks .....	48
Gambar 4.17 Diagram Alir Proses Menghitung Hasil Prediksi .....	49
Gambar 4.18 Diagram Alir Proses <i>Testing</i> .....	50
Gambar 4.19 Diagram Alir Proses Denormalisasi .....	51
Gambar 4.20 Diagram Alir Proses Menghitung Nilai Evaluasi .....	52
Gambar 4.21 Perancangan <i>Form Datasets</i> .....	65
Gambar 4.22 Perancangan <i>Form Training</i> .....	66
Gambar 4.23 Perancangan <i>Form Testing</i> .....	67
Gambar 5.1 Implementasi <i>Form Datasets</i> .....	84

Gambar 5.2 Implementasi <i>Form Training</i> .....	85
Gambar 5.3 Implementasi <i>Form Testing</i> .....	86
Gambar 5.4 Implementasi <i>Chart</i> hasil prediksi.....	86
Gambar 6.1 Grafik Rata-rata MAPE Uji Coba Jumlah <i>Hidden Neuron</i> .....	88
Gambar 6.2 Grafik Rata-rata MAPE Uji Coba Jumlah Fitur .....	89
Gambar 6.3 Grafik Rata-rata MAPE Uji Coba Jangka Waktu Prediksi.....	91
Gambar 6.4 Grafik Rata-rata MAPE Uji Coba Jumlah <i>Context Neuron</i> .....	93
Gambar 6.5 Grafik Nilai MAPE Uji Coba Perbandingan Metode RELMNN dengan Metode ELM.....	95
Gambar 6.6 Grafik Waktu Uji Coba Perbandingan Metode RELMNN dengan Metode ELM.....	96

## DAFTAR LAMPIRAN

LAMPIRAN A DATA NILAI TUKAR RUPIAH INDONESIA TERHADAP DOLAR AMERIKA SERIKAT .....	102
LAMPIRAN B HASIL UJI COBA .....	128
B.1 Hasil Uji Coba Jumlah <i>Hidden Neuron</i> .....	128
B.2 Hasil Uji Coba Jumlah Fitur .....	129
B.3 Hasil Uji Coba Jangka Waktu Prediksi .....	129
B.4 Hasil Uji Coba Jumlah <i>Context Neuron</i> .....	130
B.5 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Nilai MAPE .....	131
B.6 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Selisih Nilai Tukar Rupiah Indonesia terhadap Dolar Amerika	131
B.7 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Waktu yang Dibutuhkan Pada Proses <i>Training</i> dan <i>Testing</i> ...	132
LAMPIRAN C VISUALISASI HASIL UJI COBA .....	133
C.1 Visualisasi Hasil Uji Coba Jangka Waktu 1 Bulan .....	133
C.2 Visualisasi Hasil Uji Coba Jangka Waktu 4 Bulan .....	134
C.3 Visualisasi Hasil Uji Coba Jangka Waktu 1 Tahun .....	135
C.4 Visualisasi Hasil Uji Coba Jumlah Data Latih dan Data Uji yang Sama. ....	136



# BAB 1 PENDAHULUAN

## 1.1 Latar belakang

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 di anggap 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).

Indonesia sebagai negara berkembang, telah melakukan sejumlah upaya untuk meningkatkan nilai tukar rupiah, sehingga saat ini, tanggal 09 Maret 2017 nilai satuan Dolar Amerika Serikat menjadi 13.398 Rupiah. Hal ini dilakukan semenjak nilai tukar terus melemah pada tahun 2015, misalnya pada tanggal 17 September 2015 nilai satuan Dolar Amerika Serikat mencapai 14.479 Rupiah. Padahal, pada tahun 2013 lebih tepatnya tanggal 12 Juli 2013 nilai satuan Dolar Amerika Serikat masih 9.990 Rupiah. Nilai tukar Rupiah Indonesia sampai tahun 2017 terus menunjukkan ketidakstabilannya, dan sangat sulit untuk diprediksi. Oleh karena itu, dibutuhkan sebuah program yang dapat memprediksi nilai tukar dengan akurat dan cepat.

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, skripsi ini mengusulkan penelitian yang melakukan prediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat, sedangkan metode yang digunakan pada penelitian ini yaitu *Recurrent Extreme Learning Machine Neural Network (RELMNN)*, diharapkan dengan menggunakan metode tersebut, penelitian ini mendapat nilai evaluasi yang lebih baik, jika dibandingkan dengan metode yang pernah digunakan sebelumnya. Berdasarkan latar belakang di atas judul dari skripsi ini adalah “Prediksi Nilai Tukar Rupiah Indonesia Terhadap Dolar Amerika Serikat Menggunakan Metode *Recurrent Extreme Learning Machine Neural Network*”.

## **1.2 Rumusan masalah**

Berdasarkan latar belakang masalah yang ada, maka dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana mengimplementasikan metode RELMNN untuk prediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat?
2. Berapa nilai evaluasi yang dihasilkan metode RELMNN dalam memprediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat?
3. Berapa perbandingan nilai evaluasi yang dihasilkan antara metode RELMNN dan ELM dalam memprediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat?

### 1.3 Tujuan

Tujuan dari penelitian ini dibedakan menjadi dua yaitu tujuan umum dan tujuan khusus:

Tujuan Umum:

Memprediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat dengan metode *Recurrent Extreme Learning Machine Neural Network*.

Tujuan Khusus:

1. Mengimplementasikan metode RELMNN untuk memprediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat.
2. Mengetahui nilai evaluasi yang dihasilkan metode RELMNN dalam memprediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat.
3. Mengetahui perbandingan nilai evaluasi yang dihasilkan antara metode RELMNN dan ELM dalam memprediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat.

### 1.4 Manfaat

Manfaat dari penelitian adalah mengetahui hasil prediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat, sehingga dapat bermanfaat bagi perusahaan dalam menyusun strategi bisnis, investor dalam menyusun strategi investasi, dan juga dapat membantu Indonesia dalam merencanakan kebijakan ekonomi dan keuangan untuk masa depan.

### 1.5 Batasan masalah

Terdapat beberapa batasan yang perlu diketahui dalam penggunaan layanan ini, adapun batasan sebagai berikut:

1. Prediksi yang dilakukan pada penelitian ini menggunakan analisis Teknikal.
2. Data yang digunakan didapatkan dari website [www.ofx.com](http://www.ofx.com), berupa nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat perhari mulai dari 1 Januari 2009 sampai 11 Maret 2017.
3. Hasil akhir dari prediksi dengan metode RELMNN hanya dibandingkan dengan metode ELM.

### 1.6 Sistematika pembahasan

Untuk memberikan gambaran secara menyeluruh mengenai penelitian ini, maka sistematika penulisan dibagi dalam tujuh bab sebagai berikut:

#### **BAB I PENDAHULUAN**

Bab ini akan menjelaskan mulai dari latar belakang masalah sampai sistematika penulisan, dalam bab ini akan menjadi alasan kenapa penelitian ini dilakukan.

## **BAB II LANDASAN KEPUSTAKAAN**

Bab ini berisi tentang kajian pustaka dan dasar teori yang berkaitan dengan objek penelitian dan metode penelitian, seperti pengetahuan dasar mengenai prediksi, nilai tukar uang, *Extreme Learning Machine*, *Recurrent Extreme Learning Machine Neural Network* dan Nilai Evaluasi.

## **BAB III METODOLOGI**

Bab ini menjelaskan tentang metodologi penelitian yang akan digunakan. Metodologi penelitian ini dapat memberikan gambaran secara menyeluruh mengenai tahapan yang akan dilakukan dalam penelitian.

## **BAB IV PERANCANGAN**

Bab ini membahas mengenai proses perancangan yang dilakukan dalam penelitian.

## **BAB V IMPLEMENTASI**

Bab ini menyajikan implementasi metode pada program untuk memprediksi nilai tukar uang, sesuai dengan rancangan pada bab perancangan.

## **BAB VI PENGUJIAN DAN ANALISIS**

Bab ini menyajikan hasil pengujian dan menganalisis hasil pengujian, yang dilakukan pada beberapa parameter dan data.

## **BAB VII PENUTUP**

Bab ini berisi kesimpulan dan saran dari seluruh penelitian, sehingga dapat digunakan untuk pengembangan penelitian selanjutnya.

## BAB 2 LANDASAN KEPUSTAKAAN

Bab ini berisi kajian pustaka mengenai penelitian yang pernah dilakukan dan dasar teori mengenai prediksi, nilai tukar uang, normalisasi dan denormalisasi data, Jaringan Syaraf Tiruan, ELM, Operasi Baris Elementer, RELMNN dan nilai evaluasi.

### 2.1 Kajian Pustaka

Penelitian pertama yaitu penelitian yang dilakukan Rehman et al. (2014) untuk memprediksi nilai tukar uang, penelitian tersebut menggunakan gabungan dari 2 metode yaitu *Cartesian Genetic Programming* dan *Recurrent Neural Network*, didapatkan tingkat keakuratan yang tinggi yaitu 98,872%. Detail lengkap dari penelitian ini dapat di lihat pada Tabel 2.1. Metode *Recurrent Neural Network* dalam penelitian ini memiliki beberapa kelebihan yaitu dapat menghilangkan koneksi yang berlebihan, efisiensi dari segi waktu dan pelaksanaan. Namun, kombinasi dari dua metode ini tidak terlalu memperhatikan nilai keluaran yang menjadi feedback, hal ini penting untuk diperhatikan, karena jika nilai keluaran awal yang dihasilkan tidak akurat, maka akan berdampak pada nilai keluaran selanjutnya, yang kemungkinan akan menjauh dari target yang diinginkan.

Penelitian kedua yaitu penelitian yang dilakukan Handika et al. (2016) untuk meramalkan jumlah penjualan barang, dalam penelitian tersebut melakukan perbandingan kinerja antara metode *Extreme Learning Machine* dan *Particle Swarm Optimization Extreme Learning Machine* (PSO-ELM), hasil dari penelitian ini didapatkan nilai MSE sekitar 0,01121 sampai 0,01161 untuk PSO-ELM, lebih kecil dibandingkan dengan metode ELM yang menghasilkan nilai MSE sekitar 0,01315 sampai 0,01419. Detail lengkap dari penelitian ini dapat di lihat pada Tabel 2.1. Penelitian ini menggunakan gabungan metode yang melibatkan ELM, dikarenakan metode ELM memberikan kecepatan yang lebih baik dari pada metode lainnya. Penelitian ini menunjukkan bahwa penggabungan antara metode *Particle Swarm Optimization* (PSO) dan ELM berhasil mendapatkan hasil yang lebih baik, terutama jika dibandingkan dengan ELM itu sendiri. Namun, PSO-ELM memerlukan waktu proses yang lebih lama dari pada ELM sendiri, hal ini dapat dibuktikan pada penelitian tersebut, dengan menggunakan laptop PSO-ELM membutuhkan waktu sekitar 20 detik, sementara ELM hanya membutuhkan waktu kurang lebih 0,04 detik.

Penelitian ketiga yaitu penelitian yang dilakukan Ertugrul (2016) untuk meramalkan beban listrik, dalam penelitian tersebut digunakan metode baru yaitu *Recurrent Extreme Learning Machine* (RELM), hasil dari penelitian ini didapatkan nilai evaluasi yang lebih baik dari pada metode ELM, 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, perbandingan kecepatan yang dibutuhkan antara RELM dan ELM yaitu 13:12. Detail lengkap dari

penelitian ini dapat di lihat pada Tabel 2.1. Metode RELM memberikan hasil nilai RMSE hampir dua kali lebih baik dari pada metode yang lain. Selain itu, waktu yang dibutuhkan dalam tahap pelatihan tidak beda jauh dengan ELM, keduanya sangat cepat dari pada metode yang lain. Oleh karena itu, kombinasi dari metode ini kemungkinan besar dapat menghasilkan hasil prediksi yang akurat dan sangat baik juga, jika digunakan untuk memprediksi nilai tukar uang.

Dari paparan kajian pustaka tersebut, maka diusulkan penelitian untuk melakukan prediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat, dengan menggunakan metode *Recurrent Extreme Learning Machine Neural Network*, diharapkan dalam penelitian ini, metode tersebut juga dapat digunakan dalam memprediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat dengan akurat dan sangat baik.

**Tabel 2.1 Perbandingan dengan penelitian sebelumnya**

No	Judul	Objek	Metode	Keluaran
		Masukan dan Parameter	Proses	Hasil Penelitian
1	<i>Foreign Currency Exchange Rates Prediction using CGP and Recurrent Neural Network</i> (Rehman et al., 2014)	Nilai tukar uang	<i>Cartesian Genetic Programming</i> dan <i>Recurrent Neural Network</i>	- Hasil berupa akurasi dan MAPE
		<ul style="list-style-type: none"> <li>- Jumlah Feedback</li> <li>- Input berupa 5 mata uang yang berbeda, Yen Jepang, Dolar Selandia Baru, Dolar Kanada, Won Korea, Rupiah Indonesia.</li> <li>- Jumlah Node</li> </ul>	<ul style="list-style-type: none"> <li>- Di latih dengan data historis dan data feedback dari pelatihan sebelumnya</li> <li>- Membagi data menjadi 5 bagian</li> <li>- Memasukkan 10 nilai tukar uang ke jaringan</li> <li>- Proses RCGPANN</li> <li>- Melakukan fungsi aktivasi sigmoid</li> <li>- mengukur kinerjanya dengan MAPE</li> </ul>	Menghasilkan model yang efisien secara komputasi dan akurat untuk prediksi nilai tukar uang, dengan tingkat keakuratan 98,872% untuk jangka waktu 1000 hari
2	Perbandingan Metode <i>Extreme Learning Machine</i> dan <i>Particle Swarm Optimization</i> <i>Extreme Learning Machine</i> untuk Peramalan Jumlah	Jumlah penjualan barang	<i>Extreme Learning Machine</i> dan <i>Particle Swarm Optimization</i> <i>Extreme Learning Machine</i>	<ul style="list-style-type: none"> <li>- <i>mean square error</i></li> <li>- waktu proses</li> </ul>
		<ul style="list-style-type: none"> <li>- Maksimum iterasi</li> <li>- Jumlah partikel</li> </ul>	<ul style="list-style-type: none"> <li>- Input masukan maksimum iterasi dan jumlah partiel.</li> <li>- Normalisasi data latih</li> <li>- Buat populasi PSO</li> </ul>	Dari penelitian ini dapat dihasilkan bahwa rata-rata MSE yang dihasilkan PSO-ELM lebih kecil dibandingkan ELM, sekitar 0,01121 sampai 0,01161

**Tabel 2.1 Perbandingan dengan penelitian sebelumnya**

No	Judul	Objek	Metode	Keluaran
		Masukan dan Parameter	Proses	Hasil Penelitian
	Penjualan Barang (Handika et al., 2016)		<ul style="list-style-type: none"> <li>- Inisialisasi jumlah <i>hidden nodes</i>, bobot dan bias</li> <li>- Hitung output</li> <li>- Hitung vektor bobot output</li> <li>- Hitung fungsi aktivasi</li> <li>- Tetapkan Gbest</li> <li>- Update posisi global terbaik</li> <li>- Update kecepatan setiap partikel</li> <li>- Evaluasi nilai fitness menggunakan posisi baru</li> </ul>	untuk PSO-ELM, dan sekitar 0,01315 sampai 0,01419 untuk ELM. Oleh karena itu dapat dikatakan metode PSO mampu mengoptimasi jumlah <i>hidden nodes</i> dari ELM untuk meramalkan jumlah penjualan barang.
3	Forecasting electricity load by a novel recurrent extreme learning machines approach (Ertugrul, 2016)	Beban listrik	- <i>Recurrent Extreme Learning Machine</i>	- Nilai RMSE
		<ul style="list-style-type: none"> <li>- Input data berupa beban listrik</li> <li>- Jumlah hidden neuron</li> <li>- Jumlah <i>context neurons</i></li> </ul>	<ul style="list-style-type: none"> <li>- Menentukan parameter jaringan yang optimal.</li> <li>- Proses peramalan dengan RELM</li> <li>- Menghitung nilai evaluasi dengan RMSE</li> </ul>	Mendapatkan nilai evaluasi yang lebih tinggi dari pada metode ELM, RNN, LR, kSR, kNNR, GPR, dan GRNN, dengan kecepatan yang relatif sama dengan ELM.
4	Prediksi Nilai Tukar Rupiah Indonesia terhadap Dolar Amerika Serikat Menggunakan Metode <i>Recurrent Extreme Learning Machine Neural Network</i> (Usulan)	Nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat	- <i>Recurrent Extreme Learning Machine Neural Network</i>	<ul style="list-style-type: none"> <li>- Nilai MAPE</li> <li>- Nilai tukar uang Rupiah Indonesia terhadap Dolar Amerika Serikat yang akan datang</li> </ul>
		<ul style="list-style-type: none"> <li>- Input data berupa nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat</li> <li>- Jumlah hidden neuron</li> <li>- Jumlah <i>context neurons</i></li> </ul>	<ul style="list-style-type: none"> <li>- Normalisasi data</li> <li>- Proses prediksi dengan metode <i>Recurrent Extreme Learning Machine Neural Network</i></li> <li>- Menghitung nilai evaluasi dengan MAPE</li> </ul>	Dari penelitian ini diharapkan metode <i>Recurrent Extreme Learning Machine Neural Network</i> dapat digunakan dalam memprediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat dengan sangat baik.

Sumber: Rehman et al.(2014), Handika et al. (2016), dan Ertugrul (2016)

## 2.2 Prediksi

Prediksi merupakan proses memperkirakan sesuatu secara sistematis, mengenai sesuatu yang mungkin terjadi di masa depan, didasari oleh informasi masa lalu sampai saat ini. Hasil prediksi tidak harus mendapatkan jawaban secara pasti, akan tetapi berusaha untuk mencari jawaban yang kemungkinan besar akan terjadi (Herdianto, 2013).

Prediksi dapat dikategorikan ke dalam dua bentuk analisis yaitu analisis fundamental dan analisis teknikal. Analisis fundamental menggunakan informasi keuangan secara mendalam pada masing-masing negara, seperti statistik ekonomi, permintaan mata uang dan lain-lain, dalam memprediksi nilai tukar uang di masa depan. Analisis teknikal mempelajari tren nilai tukar mata uang di masa lampau dan mencoba untuk memprediksi prilakunya di masa depan (He dan Shen, 2007). Dalam segi jangka waktu prediksi secara umum dibedakan menjadi 3 yaitu (Siwi, 2016):

1. Prediksi jangka panjang, yang merupakan prediksi yang dilakukan dalam jangka waktu lebih dari 24 bulan.
2. Prediksi jangka menengah, yang merupakan prediksi yang dilakukan dalam jangka waktu 3-24 bulan.
3. Prediksi jangka pendek, yang merupakan prediksi yang dilakukan dalam jangka waktu kurang dari 3 bulan.

Sedangkan, prediksi nilai tukar uang dapat dibagi menjadi tiga jangka waktu yaitu (Bailliu dan King, 2005):

1. Prediksi jangka panjang, yang merupakan prediksi yang dilakukan dalam jangka waktu beberapa tahun sampai puluhan tahun.
2. Prediksi jangka menengah, yang merupakan prediksi yang dilakukan dalam jangka waktu 1 bulan sampai beberapa tahun.
3. Prediksi jangka pendek, yang merupakan prediksi yang dilakukan dalam jangka waktu 1 hari sampai 1 bulan.

## 2.3 Nilai Tukar Uang

Nilai tukar atau biasa di sebut kurs adalah suatu perbandingan nilai uang, diantara dua negara (Jauhari et al., 2016a). Nilai tukar juga bisa diartikan sebagai harga satuan mata uang asing dalam mata uang dalam negeri dan sebaliknya. Formulasi dari nilai Rupiah yang ditukar dengan mata uang asing adalah (Simorangkir dan Suseno, 2004):

Nilai Tukar IDR/USD = Rupiah yang dibutuhkan untuk membeli satu Dolar Amerika Serikat

Nilai Tukar IDR/YEN = Rupiah yang dibutuhkan untuk membeli satu Yen Jepang



Ketika Nilai Tukar IDR/USD atau IDR/YEN meningkat maka Rupiah mengalami depresiasi, sedangkan ketika menurun maka Rupiah mengalami apresiasi. Contoh berikut akan memperjelas pengertian tersebut. Misalnya, nilai tukar 1 Dolar Amerika Serikat terhadap Rupiah sebesar Rp.13.000. Apabila nilai tukar berubah menjadi Rp.14.000, maka nilai tukar Rupiah mengalami depresiasi. Sebaliknya, apabila nilai tukar berubah menjadi Rp.12.000, maka nilai tukar Rupiah mengalami apresiasi (Simorangkir dan Suseno, 2004).

Sistem nilai tukar adalah sistem yang digunakan oleh suatu negara, untuk mendukung neraca pembayaran dan membantu efektivitas kebijakan moneter. Apresiasi nilai tukar yang berlebihan dapat mengakibatkan harga barang-barang ekspor menjadi lebih mahal di luar negeri dan barang-barang import menjadi lebih murah, hal ini mengakibatkan neraca perdagangan menjadi memburuk. Sementara itu, depresiasi nilai tukar yang berlebihan dapat mengakibatkan tingginya laju inflasi, hal ini akan mengganggu tujuan akhir kebijakan moneter untuk memelihara stabilitas negara. Sistem nilai tukar dapat diklasifikasikan menjadi tiga kelompok yaitu (Simorangkir dan Suseno, 2004):

1. Sistem nilai tukar tetap murni

Pada sistem nilai tukar tetap, mata uang suatu negara ditetapkan secara tetap dengan mata uang asing tertentu, dengan penetapan nilai tukar secara tetap terdapat kemungkinan nilai tukar yang ditetapkan terlalu tinggi atau terlalu rendah dari nilai sebenarnya. Banyak negara yang sudah meninggalkan sistem nilai tukar tetap, sebab sistem ini dapat mengganggu neraca perdagangan suatu negara dan banyak negara yang tidak memiliki cadangan devisa yang cukup untuk mempertahankan sistem ini. Sementara itu, beberapa negara yang masih menggunakan sistem ini dikarenakan sistem ini dapat digunakan sebagai alat pengendali inflasi.

2. Sistem nilai tukar mengambang murni

Dalam sistem nilai tukar mengambang, penetapan nilai mata uang domestik terhadap mata uang asing ditentukan oleh mekanisme pasar. Oleh karena itu, pada sistem ini nilai mata uang akan berubah setiap saat tergantung dari permintaan dan penawaran mata uang domestik dan perilaku spekulasi. Argumentasi negara-negara yang menggunakan sistem nilai tukar mengambang ini ada dua yaitu, sistem ini memungkinkan suatu negara mencegah kebijakan ekonomi makronya dari dampak kebijakan luar dan sistem ini tidak memerlukan cadangan devisa yang besar. Namun sistem ini juga mengakibatkan nilai tukar berfluktuasi dan dapat memicu inflasi di dalam negeri.

3. Sistem nilai tukar tetap tetapi dapat disesuaikan

Sistem ini merupakan kombinasi dari sistem tukar tetap dan mengambang, dalam sistem ini besarnya nilai tukar ditetapkan oleh pembuat kebijakan, bank sentral, dan dipertahankan melalui intervensi langsung di pasar valuta asing. Sistem ini dapat mendorong terciptanya kebijakan moneter dan kebijakan nilai tukar yang independen.

## 2.4 Normalisasi dan Denormalisasi Data

Proses normalisasi bertujuan untuk standarisasi semua data yang digunakan, sehingga data berada pada jarak tertentu (Siwi, 2016). Data di normalisasi sebelum diolah, dan akan kembali dilakukan denormalisasi setelah digunakan. Data akan di normalisasi pada interval  $[0,1]$ , tapi karena nilai dari fungsi aktivasi sigmoid merupakan fungsi asimtotik, yang nilainya tidak pernah mencapai 0 ataupun 1, maka digunakan interval yang lebih kecil yaitu  $[0.1,0.9]$  (Irawan, 2015). Persamaan normalisasi dan denormalisasi yang akan digunakan, didefinisikan berturut-turut pada persamaan (2.1) dan persamaan (2.2).

$$y = \frac{x - \min}{\max - \min} (0,8) + 0,1 \quad (2.1)$$

$$y' = \frac{y - (0,1)}{0,8} (\max - \min) + \min \quad (2.2)$$

Keterangan:

$x$  = nilai data asli yang belum dinormalisasi

$\min$  = nilai minimum dari seluruh data asli

$\max$  = nilai maksimum dari seluruh data asli

$y$  = nilai data setelah proses normalisasi

$y'$  = nilai data setelah proses denormalisasi

## 2.5 Jaringan Syaraf Tiruan

Pertama *Artificial Neural Network* diciptakan pada tahun 1958 oleh psikolog Frank Rosenblatt, saat itu *Artificial Neural Network* disebut Perceptron, dengan tujuan untuk memodelkan bagaimana cara otak manusia memproses data visual dan belajar mengenali obyek (Nayakovit et al., 2010). *Artificial Neural Network* yang biasa disebut dengan Jaringan Syaraf Tiruan (JST), merupakan sebuah sistem yang biasa digunakan untuk memproses informasi, dan memiliki beberapa karakteristik yang hampir sama dengan jaringan syaraf biologis. Kemiripannya terletak pada saat pemrosesan informasi, pemrosesan informasi yang terjadi didalam otak manusia memiliki sifat adaptif, yaitu hubungan yang terjadi antar neuron dilakukan secara dinamis, yang memiliki kemampuan dalam mempelajari informasi-informasi yang belum diketahui (Nikmah, 2014).

Kebanyakan JST berisi tiga *layer* yaitu *input layer*, *output layer* dan *hidden layer*, setiap layer terdiri dari sejumlah unit yang disebut neuron dan fungsi aktivasi (Nayakovit et al., 2010). JST memiliki kelebihan yaitu bersifat fleksibel dibandingkan dengan model statistik, dapat menangani permasalahan dengan data linear atau non-linear, dan mampu mempelajari sistem yang kompleks, dimana ada keberadaan data yang tidak lengkap (Liu et al., 2009). Kemampuannya dalam belajar, memungkinkan JST untuk mengatasi banyak masalah yang sulit, atau tidak mungkin untuk dipecahkan dengan metode komputasi dan statistik

standar (Nayakovit et al., 2010). Dari kemampuan yang dimiliki, JST memiliki fungsi untuk belajar atau mempelajari, menghasilkan suatu aturan atau operasi, dan dapat menghasilkan suatu *output* yang sempurna dari *input* yang dimasukkan.

### 2.5.1 Faktor Bobot

Bobot adalah nilai tingkat atau kepentingan hubungan antara suatu neuron dengan neuron yang lain. Jadi besar bobot suatu koneksi, dapat menandakan semakin pentingnya hubungan kedua neuron tersebut. Bobot dapat berupa bilangan real maupun integer, dan bobot tersebut bisa di inisialisasi untuk berada di dalam nilai tertentu. Pada Jaringan Syaraf Tiruan, jaringan otomatis akan memperbaiki bobot secara terus-menerus, karena adanya kemampuan untuk belajar atau pelatihan. Setiap ada suatu masalah baru atau data baru, jaringan dapat belajar lagi dengan menambahkan masalah/data baru tersebut (Puspaningrum, 2006).

### 2.5.2 Fungsi Aktivasi

Fungsi aktivasi berguna untuk memproses *output* jaringan, sehingga sesuai dengan nilai target. Dalam Jaringan Syaraf Tiruan terdapat dua fungsi aktivasi yaitu fungsi *sigmoid biner*, dan fungsi *sigmoid bipolar*. Fungsi-fungsi aktivasi tersebut harus bersifat *continue*, diferensiabel, dan tidak menurun secara monoton. Selain itu fungsi aktivasi juga diharapkan dapat mendekati nilai-nilai maksimum dan minimum dengan baik (Puspaningrum, 2006). Untuk menghitung fungsi aktivasi *sigmoid biner* dapat menggunakan Persamaan 2.3 (Jauhari et al., 2016b).

$$y = f(y'_j) = \frac{1}{1+\exp(-y'_j)} \quad (2.3)$$

Keterangan:

$y$  = nilai output data setelah dilakukan proses aktivasi

$y'_j$  = nilai output jaringan pada data ke- $j$  sebelum dilakukan proses aktivasi

## 2.6 Extreme Learning Machine

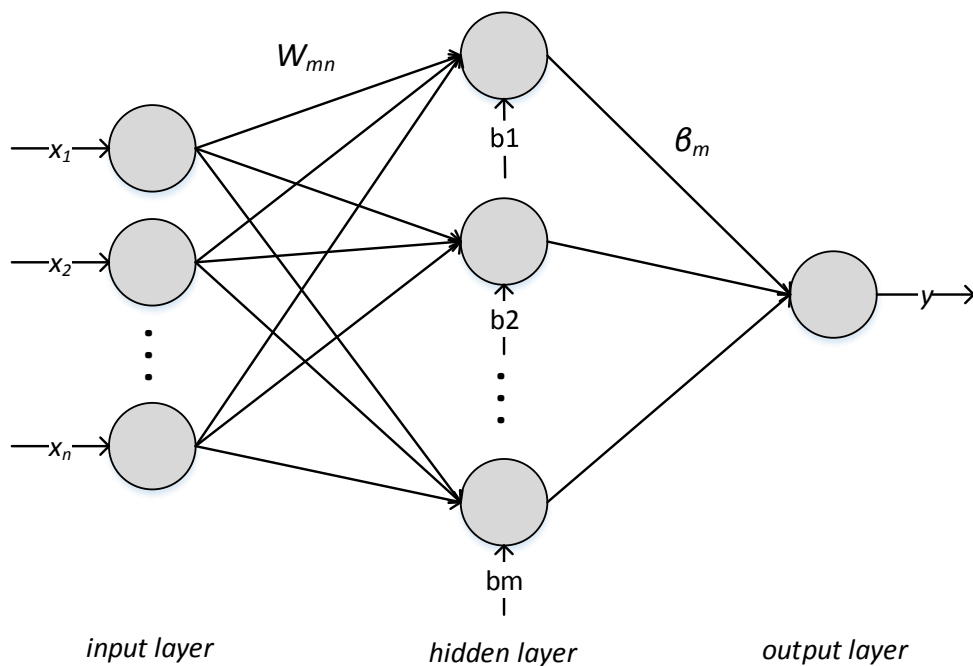
ELM singkatan dari *Extreme Learning Machine* adalah metode yang sering dibicarakan, metode ini merupakan metode pembelajaran baru dari Jaringan Syaraf Tiruan, metode ini ketika pertama kali diperkenalkan oleh Huang dkk pada tahun 2004, hanya memiliki satu *hidden layer* pada arsitektur jaringannya. Metode ELM di buat untuk memperbaiki kekurangan-kekurangan dari metode sebelumnya, misalnya metode jaringan syaraf tiruan *backpropagation*. Perbaikan yang diutamakan yaitu pada *learning-speed* dari proses pembelajaran (Siwi, 2016).

Pada tahap pembelajaran pada JST semua parameter ditentukan secara manual, seperti parameter *input weight* dan *bias*. Parameter tersebut saling berhubungan dalam jaringan, hal ini yang menyebabkan dibutuhkan *learning-speed* yang lama (Siwi, 2016), berbeda dengan parameter pada ELM yang ditentukan secara random, yang menyebabkan ELM memiliki *learning speed* yang

cepat dan dapat menghasilkan *good generalization performance*, inilah yang menyebabkan ELM memiliki kecepatan seribu kali dari pada algoritma *jaringan syaraf tiruan backpropagation* (Azizah, 2016). ELM juga telah menunjukkan kemampuan generalisasi yang sangat baik dan mengungguli metode *Neural Network* sebelumnya, tanpa masalah *overtraining* (Wan et al., 2014). Selain itu, ELM selalu dapat menjamin hasil yang terbaik sesuai dengan *input* bobot yang digunakan (Wan et al., 2014). ELM telah terbukti secara teoritis memiliki kemampuan generalisasi yang baik dan kecepatan yang sangat cepat, pekerjaan yang tersisa dengan menggunakan ELM adalah memilih fungsi aktivasi dan jumlah hidden neuron (Chen et al., 2012).

### 2.6.1 Arsitektur Jaringan ELM

Sebuah metode dari Jaringan Syaraf Tiruan tentu memiliki arsitektur yang berbeda-beda, arsitektur jaringan ELM hampir mirip dengan arsitektur jaringan *multi layer*, namun dalam ELM hanya mempunyai satu *hidden layer* saja. Arsitektur jaringan syaraf tiruan dari *Extreme Learning Machine* dapat di lihat pada Gambar 2.1.



**Gambar 2.1 Jaringan Extreme Learning Machine**

Sumber : Cholissodin et al. (2017) dan Ertugrul (2016)

### 2.6.2 Proses Training

Proses *training* merupakan proses yang bertujuan untuk melakukan pelatihan dengan menggunakan data latih, pelatihan ini bertujuan untuk mendapatkan nilai bobot yang optimal. Tahapan-tahapan yang dilakukan ketika proses training yaitu (Cholissodin et al., 2017):

1. Membuat nilai random untuk matriks  $W_{mn}$  sebagai bobot masukan dengan range  $[-1,1]$ , dalam bentuk *array* ukuran  $m$  (jumlah *hidden neuron*) x  $n$  (jumlah *input neuron*). Kemudian buat nilai random untuk matriks bias  $b$  dengan range  $[0,1]$  dalam ukuran  $1$  x (jumlah *hidden neuron*).
2. Menghitung nilai matriks keluaran pada *hidden layer* dengan menggunakan Persamaan 2.4. Perhitungan  $b(\text{ones}(i_{train}, 1), :)$  akan memperbanyak matriks bias sebanyak jumlah data latih.

$$H = \frac{1}{1 + \exp(-(x_{train} W^T + b(\text{ones}(i_{train}, 1), :)))} \quad (2.4)$$

Keterangan:

$H$  = matriks keluaran *hidden layer*

$x_{train}$  = matriks *input* pada data latih yang telah di normalisasi

$W^T$  = matriks *transpose* dari bobot

$i_{train}$  = jumlah data latih

$b$  = matriks bias

3. Menghitung  $\hat{\beta}$  sebagai bobot keluaran dengan menggunakan Persamaan 2.5, dimana  $H^+$  atau matriks *Moore-Penrose Pseudo Invers* dapat dihitung dengan Persamaan 2.6.

$$\hat{\beta} = H^+ t \quad (2.5)$$

$$H^+ = (H^T H)^{-1} H^T \quad (2.6)$$

Keterangan:

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

$H^+$  = matriks *Moore-Penrose Pseudo Invers*

$t$  = matriks target

$H$  = matriks keluaran *hidden layer*

4. Menghitung hasil keluaran dengan menggunakan Persamaan 2.7.

$$\hat{Y} = H \hat{\beta} \quad (2.7)$$

Keterangan:

$\hat{Y}$  = hasil prediksi

$H$  = matriks keluaran *hidden layer*

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

### 2.6.3 Proses Testing

Setelah melakukan proses *training*, maka dilakukan proses *testing* dengan menggunakan data uji. Proses *testing* bertujuan untuk melakukan uji coba hasil *training*, sehingga dapat diketahui akurasi dari program (Siwi, 2016). Tahapan-tahapan yang dilakukan ketika proses *testing* yaitu (Cholissodin et al., 2017):

1. Diketahui nilai  $W_{mn}$ ,  $b$  dan  $\hat{\beta}$  dari proses *testing*.
2. Menghitung nilai matriks keluaran pada *hidden layer* dengan menggunakan Persamaan 2.8. Perhitungan  $b(\text{ones}(i_{test}, 1), :)$  akan memperbanyak matriks bias sebanyak jumlah data uji.

$$H = \frac{1}{1 + \exp(-(x_{test} W^T + b(\text{ones}(i_{test}, 1), :)))} \quad (2.8)$$

Keterangan:

$H$  = matriks keluaran *hidden layer*

$x_{test}$  = matriks *input* pada data uji yang telah di normalisasi

$W^T$  = matriks *transpose* dari bobot

$i_{test}$  = jumlah data uji

$b$  = matriks bias

3. Menghitung hasil keluaran dengan menggunakan Persamaan 2.7.
4. Denormalisasi hasil prediksi dengan menggunakan Persamaan 2.2.
5. Menghitung nilai evaluasi dengan menggunakan Persamaan 2.15.

### 2.7 Operasi Baris Elementer

Operasi Baris Elementer (OBE) adalah sebuah operasi yang digunakan pada baris yang terdapat pada suatu matriks. Operasi-operasi yang dapat dilakukan pada matriks yaitu (Darmayasa, 2015):

1. Mengalikan suatu baris dengan bilangan yang tidak nol, operasi ini dapat disimbolkan dengan  $kR_i \rightarrow R_i$ , yang artinya baris ke- $i$  berubah setelah dikalikan  $k$ .
2. Menambahkan kelipatan suatu baris pada baris lain, operasi ini dapat disimbolkan dengan  $R_i + kR_j \rightarrow R_i$ , yang artinya baris ke- $i$  berubah menjadi hasil dari penjumlahan  $R_i + kR_j$ .
3. Menukar antara dua buah baris, operasi ini dapat disimbolkan dengan  $R_i \leftrightarrow R_j$ , yang artinya baris ke- $i$  dan baris ke- $j$  bertukar posisi.

OBE biasa digunakan untuk menentukan *invers* suatu matriks, untuk menentukan *invers* matriks  $A$ , dapat menggunakan sejumlah OBE pada matriks  $A$  dan melakukan OBE pada matriks  $I$  (matriks identitas). Perhitungan *invers* matriks ini menggunakan konsep  $[A|I]$  yang dilakukan OBE menjadi  $[I|A^{-1}]$ , yang artinya

OBE digunakan untuk mengubah  $A$  menjadi matriks  $I$  (Darmayasa, 2015). Berikut contoh penggunaan OBE untuk menentukan *invers* matriks (Siwi, 2016):

Misal matriks  $A = \begin{pmatrix} 1 & 2 \\ 2 & 5 \end{pmatrix}$

Bentuk awal :  $[A|I] = \begin{pmatrix} 1 & 2|1 & 0 \\ 2 & 5|0 & 1 \end{pmatrix}$

1.  $\frac{1}{2}R_2 + (-R_1) \rightarrow R_2 \begin{pmatrix} 1 & 2|1 & 0 \\ 0 & 0,5|-1 & 0,5 \end{pmatrix}$

2.  $\frac{0,5}{2}R_1 + (-R_2) \rightarrow R_1 \begin{pmatrix} 0,25 & 0|1,25 & -0,5 \\ 0 & 0,5|-1 & 0,5 \end{pmatrix}$

3.  $\frac{1}{0,25}R_1 \rightarrow R_1 \begin{pmatrix} 1 & 0|5 & -2 \\ 0 & 0,5|-1 & 0,5 \end{pmatrix}$

4.  $\frac{1}{0,5}R_2 \rightarrow R_2 \begin{pmatrix} 1 & 0|5 & -2 \\ 0 & 1|-2 & 1 \end{pmatrix}$

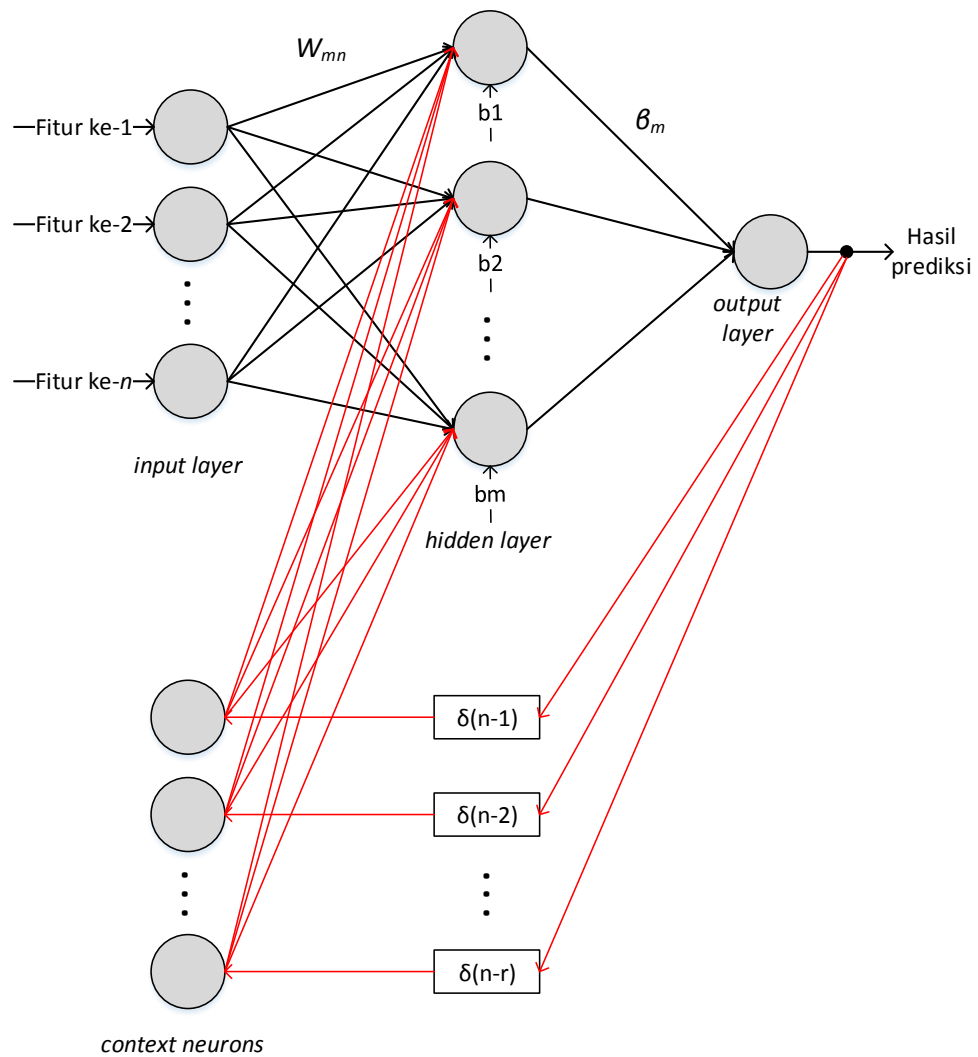
Bentuk akhir :  $[I|A^{-1}] = \begin{pmatrix} 1 & 0|5 & -2 \\ 0 & 1|-2 & 1 \end{pmatrix}$

## 2.8 Recurrent Extreme Learning Machine Neural Network (RELMNN)

*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).

### 2.8.1 Arsitektur Jaringan RELMNN

Arsitektur pada jaringan RELMNN sedikit berbeda dengan arsitektur jaringan ELM, perbedaannya terdapat pada mekanisme *recurrent* yang disimpan pada *context neurons*. *Context neurons* adalah *neuron* yang berfungsi menyimpan nilai *delayed output*, *neuron* ini dijalankan seperti *neuron input* tambahan (Ertugrul, 2016). Arsitektur RELMNN dapat di lihat pada Gambar 2.2, pada gambar tersebut merupakan arsitektur jaringan yang digunakan dalam satu data, sedangkan proses dalam metode RELMNN akan menggunakan keseluruhan data dalam satu iterasi.



**Gambar 2.2 Jaringan *Recurrent Extreme Learning Machine Neural Network***  
Sumber : Ertugrul (2016)

### 2.8.2 Proses *Training*

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

1. Inisialisasi matriks *delay*  $\delta$  dengan Persamaan 2.9.

$$\delta_{tr} = T (t - (n + r) + n) \quad (2.9)$$

Keterangan:

$\delta_{tr}$  = *delay* pada urutan data ke- $t$  kolom ke- $r$

$T$  = matriks target

$n$  = jumlah *input neuron*

$t$  = urutan data

$r$  = urutan *context neurons*



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*) x  $n+r$  (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 2.10. Perhitungan  $b(\text{ones}(i,1),:)$  akan memperbanyak matriks bias sebanyak jumlah data latih.

$$H' = \frac{1}{1 + \exp(-([x_{train}, \delta] W'^T + b(\text{ones}(i_{train},1),:)))} \quad (2.10)$$

Keterangan:

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

$x_{train}$  = matriks input yang telah di normalisasi

$\delta$  = matriks *delay*

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

$W'^T$  = 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 2.11, dimana  $H'^+$  atau matriks *Moore-Penrose Pseudo Invers* dapat dihitung dengan Persamaan 2.12.

$$\hat{\beta}' = H'^+ T \quad (2.11)$$

$$H'^+ = (H'^T H')^{-1} H'^T \quad (2.12)$$

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 2.13.

$$\hat{Y}' = H' \hat{\beta}' \quad (2.13)$$

Keterangan:

$\hat{Y}'$  = hasil prediksi setelah proses recurrent

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

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

### 2.8.3 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*  $\delta'$  dengan melakukan perhitungan pada Persamaan 2.9.
3. Menghitung nilai matriks keluaran pada *hidden layer* dengan menggunakan Persamaan 2.14.

$$H' = \frac{1}{1 + \exp(-([x_{test}, \delta'] W'^T + b(\text{ones}(i_{test}, 1, :))) )} \quad (2.14)$$

Keterangan:

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

$x_{test}$  = matriks input yang telah di normalisasi

$\delta'$  = matriks *delay*

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

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

$i_{test}$  = jumlah data uji

$b$  = matriks bias

4. Menghitung hasil prediksi dengan menggunakan Persamaan 2.13.
5. Denormalisasi hasil prediksi dengan menggunakan Persamaan 2.2.
6. Menghitung nilai evaluasi dengan menggunakan Persamaan 2.15.

### 2.9 Nilai Evaluasi

Ada berbagai jenis persamaan yang dapat digunakan untuk menghitung nilai evaluasi, tentunya untuk mengukur hasil dari prediksi, misalnya: MAE, MSE, MAPE, RMSE, dan lain-lain. Nilai *Mean Absolute Percentage Error* (MAPE) merupakan indikator yang sering digunakan dalam model prediksi. Nilai MAPE dapat dikalikan dengan 100 untuk menunjukkan nilai persentase, MAPE sering digunakan untuk melakukan validasi keakuratan pengujian (Worasuchep, 2015). MAPE digunakan ketika ukuran atau besar variabel prediksi dianggap penting dalam melakukan evaluasi hasil prediksi (Siwi, 2016). Selain itu, orang awam akan memahami persentase lebih baik dari pada nilai kesalahan kuadrat (Worasuchep, 2015). Kemampuan prediksi sangat baik jika memiliki nilai MAPE kurang dari 10% dan mempunyai kemampuan prediksi baik jika nilai MAPE kurang dari 20% (S dan W, 2015). Persamaan 2.15 digunakan untuk menghitung nilai MAPE (Jauhari et al., 2016a).

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|t_i - y_i|}{t_i} \times 100\% \quad (2.15)$$

Keterangan:

$n$  = jumlah data

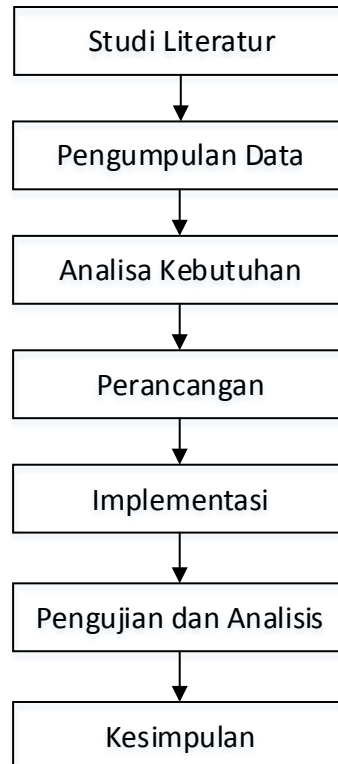
$t_i$  = nilai aktual pada data ke- $i$

$y_i$  = nilai hasil prediksi ke pada data ke- $i$

## BAB 3 METODOLOGI

### 3.1 Tahapan Penelitian

Terdapat beberapa tahapan yang ada pada penelitian ini yaitu studi literatur, pengumpulan data, pengolahan data, perancangan, implementasi, pengujian dan analisis, dan kesimpulan. Adapun tahapan metodologi penelitian dapat di lihat pada Gambar 3.1.



**Gambar 3.1 Tahapan-tahapan penelitian**

### 3.2 Studi Literatur

Studi literatur dapat dilakukan dengan membaca dan memahami jurnal dan skripsi yang terkait dengan penelitian. Jurnal dan skripsi yang terkait meliputi prediksi nilai tukar uang, *Extreme Learning Machine*, dan *Recurrent Neural Network*.

Beberapa jurnal yang dapat menjadi referensi penelitian ini, seperti jurnal yang di tulis oleh Rehman, Khan dan Mahmud (2014) yang melakukan prediksi nilai tukar uang, dengan menggunakan metode *Cartesian Genetic Programming* dan *Recurrent Neural Network*, kemudian skripsi yang di tulis oleh Siwi (2016) yang melakukan peramalan menggunakan metode *Extreme Learning Machine*.

### 3.3 Pengumpulan Data

Data yang digunakan pada penelitian ini adalah data nilai tukar Rupiah terhadap Dolar Amerika Serikat mulai dari tanggal 1 Januari 2009 sampai 11 Maret 2017. Sumber data yang digunakan dalam penelitian ini didapatkan dari website [www.ofx.com](http://www.ofx.com), yang merupakan salah satu penyedia data nilai tukar uang.

### 3.4 Analisa Kebutuhan

Analisa kebutuhan adalah tahap menganalisis kebutuhan, mengenai apa saja yang dibutuhkan dalam mengimplementasikan program pada penelitian ini. Spesifikasi dari kebutuhan yang digunakan dalam pembuatan program adalah:

1. Spesifikasi kebutuhan *Hardware*
  - Laptop dengan spesifikasi *processor* Intel(R) Core(TM) i3-3217U
  - RAM 8 GB
  - Harddisk 500GB
  - Monitor 14"
2. Spesifikasi kebutuhan *Software*
  - Sistem Operasi Windows 8
  - Microsoft Office 2013
  - NetBeans IDE 7.4
  - Notepad++
3. Spesifikasi kebutuhan Data
  - Data berupa nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat per hari mulai tanggal 1 Januari 2009 sampai 11 Maret 2017

### 3.5 Perancangan

Perancangan dilakukan untuk mempermudah proses implementasi, pengujian, dan analisis. Langkah-langkah yang dilakukan dalam perancangan adalah sebagai berikut:

1. Perancangan *User Interface*

Perancangan *user interface* atau bisa disebut antarmuka pengguna ini dilakukan agar dapat memudahkan pengguna dalam menggunakan program ini.
2. Perancangan Pengujian

Perancangan Pengujian yang dilakukan terdiri dari pengujian jangka waktu, pengujian jumlah fitur, jumlah *hidden neurons*, jumlah *context neurons* terhadap nilai MAPE.

### 3.6 Implementasi

Program ini dibuat dengan menggunakan bahasa Java GUI, dengan editor NetBeans. Implementasi di mulai dengan membuat *user interface* pada program, yang terdiri dari bagian pelatihan, pengujian, dan prediksi. Selanjutnya, mengimplementasikan metode *Recurrent Extreme Learning Machine Neural Network* pada bagian pelatihan, pengujian, dan prediksi.

### 3.7 Pengujian dan Analisis

Setelah proses pelatihan, maka akan didapatkan bobot yang optimal untuk memprediksi nilai tukar uang, nilai bobot digunakan untuk melakukan pengujian, untuk mengevaluasi hasil prediksi pada penelitian ini maka digunakan nilai evaluasi MAPE.

Terdapat empat pengujian yang akan dilakukan, 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.

Kemudian, tahap analisis dilakukan dengan melihat nilai MAPE dari masing-masing pengujian, untuk menilai kinerja dari *Recurrent Extreme Learning Machine Neural Network*.

### 3.8 Kesimpulan

Tahap penarikan kesimpulan akan didapatkan dari pengujian dan analisis. Penarikan kesimpulan bertujuan untuk menjawab rumusan permasalahan yang telah dirumuskan. Tahap yang terakhir yaitu memberikan saran, sehingga dapat memperbaiki kesalahan-kesalahan yang terjadi, maupun memberikan masukan untuk penelitian selanjutnya.

## BAB 4 PERANCANGAN

Bab ini menjelaskan formulasi permasalahan, perancangan jaringan, diagram alir program, perhitungan manual, perancangan *user interface* dan perancangan skenario pengujian.

### 4.1 Formulasi Permasalahan

Sistem prediksi pada permasalahan ini memiliki masukan berupa data nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat perhari dalam format .xls dan parameter lain yaitu: jumlah fitur, jumlah *hidden neuron*, jumlah *context neuron*, nilai *input* bobot yang didapatkan secara random dalam range  $[-1,1]$  dan nilai bias yang didapatkan secara random dalam range  $[0,1]$ . Dalam pengerjaannya, metode RELMNN terbagi menjadi dua bagian yaitu proses *training* dan proses *testing*. Namun, sebelum melakukan proses tersebut, perlu dilakukan proses normalisasi data pada range  $[0.1,0.9]$ . Setelah itu, proses *training* dilakukan dengan menggunakan data latih, sampai seluruh tahapan proses *training* selesai dilakukan dan didapatkan nilai bobot, bias, matriks *delay*, dan matriks bobot keluaran yang dibutuhkan pada proses selanjutnya. Selanjutnya, proses *testing* dilakukan dengan menggunakan data uji, sampai didapatkan hasil prediksi dan didapatkan nilai evaluasi dengan perhitungan MAPE. Tahapan terakhir yaitu melakukan denormalisasi hasil prediksi, untuk mendapatkan hasil prediksi berupa nilai tukar uang.

Adapun sampel data nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat yang digunakan pada penelitian ini adalah data dengan parameter teknikal yang berjumlah 15 *record* dengan 3 fitur. Sampel data ini ditunjukkan pada Tabel 4.1 (data lengkap terdapat pada Lampiran A).

**Tabel 4.1 Data nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat**

Data ke- <i>i</i>	Tanggal	Fitur 1	Fitur 2	Fitur 3	Nilai Tukar
1	24 Februari 2017	13360	13342	13311	13340
2	25 Februari 2017	13342	13311	13340	13330
3	26 Februari 2017	13311	13340	13330	13330
4	27 Februari 2017	13340	13330	13330	13355
5	28 Februari 2017	13330	13330	13355	13346
6	1 Maret 2017	13330	13355	13346	13340
7	2 Maret 2017	13355	13346	13340	13383
8	3 Maret 2017	13346	13340	13383	13373
9	4 Maret 2017	13340	13383	13373	13323
10	5 Maret 2017	13383	13373	13323	13323
11	6 Maret 2017	13373	13323	13323	13322.75

**Tabel 4.1 Data nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat**

Data ke- <i>i</i>	Tanggal	Fitur 1	Fitur 2	Fitur 3	Nilai Tukar
12	7 Maret 2017	13323	13323	13322.75	13337.75
13	8 Maret 2017	13323	13322.75	13337.75	13407
14	9 Maret 2017	13322.75	13337.75	13407	13394
15	10 Maret 2017	13337.75	13407	13394	13363

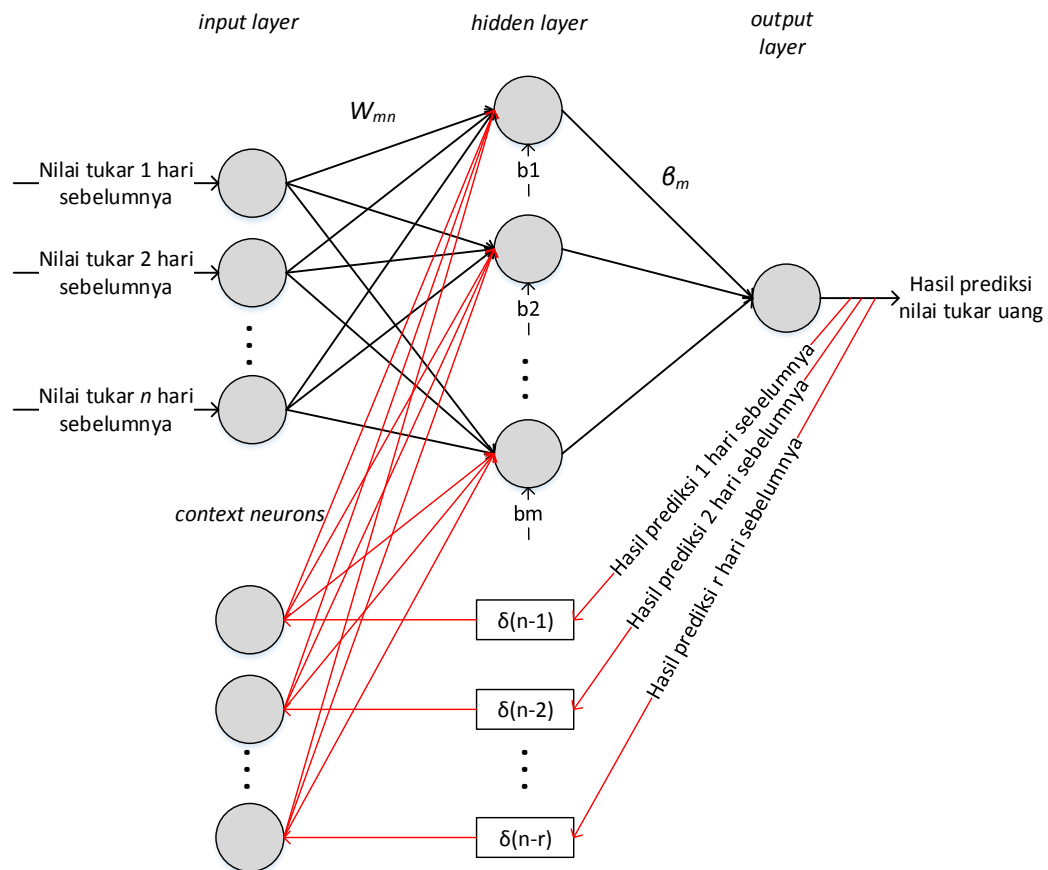
Nilai pada fitur 3 di dapatkan dari nilai tukar 1 hari sebelumnya, sedangkan nilai pada fitur 2 di dapatkan dari nilai tukar 2 hari sebelumnya, dan nilai pada fitur 3 di dapatkan dari nilai tukar 1 hari sebelumnya. Keterangan setiap parameter data pada Tabel 4.1 adalah sebagai berikut:

1. Tanggal : menerangkan tanggal saat nilai tukar itu terjadi (date)
2. Fitur 1 : menerangkan fitur pertama yang merupakan nilai tukar tiga hari sebelum tanggal pada parameter pertama. (double)
3. Fitur 2 : menerangkan fitur kedua yang merupakan nilai tukar dua hari sebelum tanggal pada parameter pertama. (double)
4. Fitur 3 : menerangkan fitur ketiga yang merupakan nilai tukar satu hari sebelum tanggal pada parameter pertama. (double)
5. Nilai Tukar : menerangkan nilai tukar pada saat tanggal di parameter pertama (double)

## 4.2 Perancangan Jaringan

Perancangan jaringan merupakan gambaran dari arsitektur jaringan yang digunakan pada penelitian ini. Jaringan yang digunakan pada penelitian ini dapat dilihat pada Gambar 4.1, 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*  $\delta$  yang merupakan nilai tukar uang sebelumnya. Setiap *context neuron* terhubung dengan *hidden layer* melalui *input weight* yang nilainya berbeda-beda.

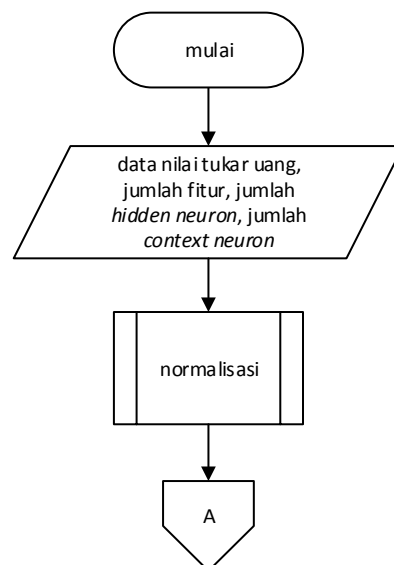


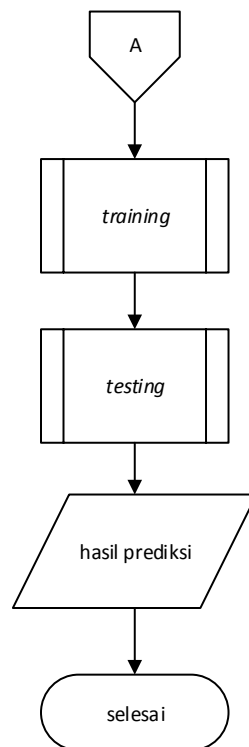


**Gambar 4.1** Arsitektur Jaringan

### 4.3 Diagram Alir Program

Diagram alir program merupakan gambaran dengan menggunakan flowchart mengenai proses pada program. Diagram alir program pada prediksi nilai tukar ini meliputi diagram proses *input* data, normalisasi, *training*, dan *testing*. Gambar 4.2 menunjukkan diagram alir pada program ini.





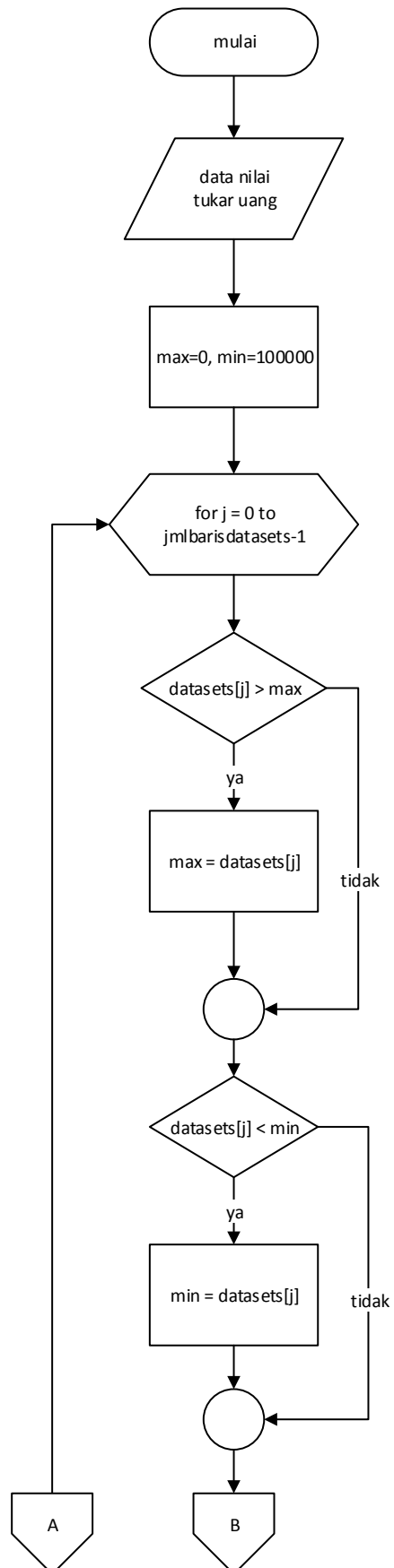
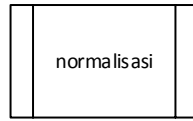
**Gambar 4.2 Diagram Alir Program**

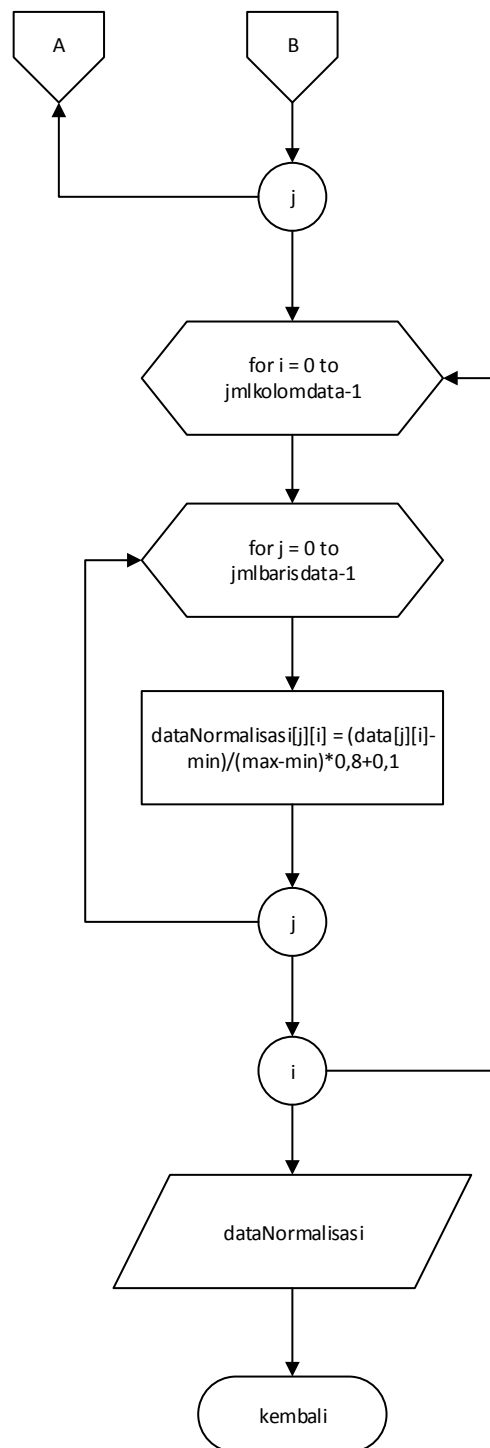
Berdasarkan Gambar 4.2 di atas, langkah-langkah proses pada program adalah sebagai berikut:

1. Masukkan data nilai tukar uang, jumlah fitur, jumlah *hidden neuron*, jumlah *context neuron* yang digunakan.
2. Melakukan proses normalisasi pada data nilai tukar uang atau *datasets* dengan menggunakan Persamaan 2.1. Diagram alir untuk proses normalisasi ditunjukkan pada Gambar 4.3.
3. Melakukan proses *training* dengan menggunakan data latih untuk mendapatkan nilai matriks bobot, bobot keluaran, bias, dan target. Diagram alir untuk proses *training* ditunjukkan pada Gambar 4.4.
4. Melakukan proses *testing* dengan menggunakan data uji untuk mendapatkan hasil prediksi. Diagram alir untuk proses *testing* ditunjukkan pada Gambar 4.18.
5. Menampilkan hasil prediksi.

#### **4.3.1 Normalisasi**

Proses normalisasi dilakukan untuk melakukan standarisasi pada semua data yang digunakan pada perhitungan, sehingga menghasilkan data pada range tertentu. Diagram alir dari proses normalisasi ditunjukkan pada Gambar 4.3.





**Gambar 4.3 Diagram Alir Proses Normalisasi**

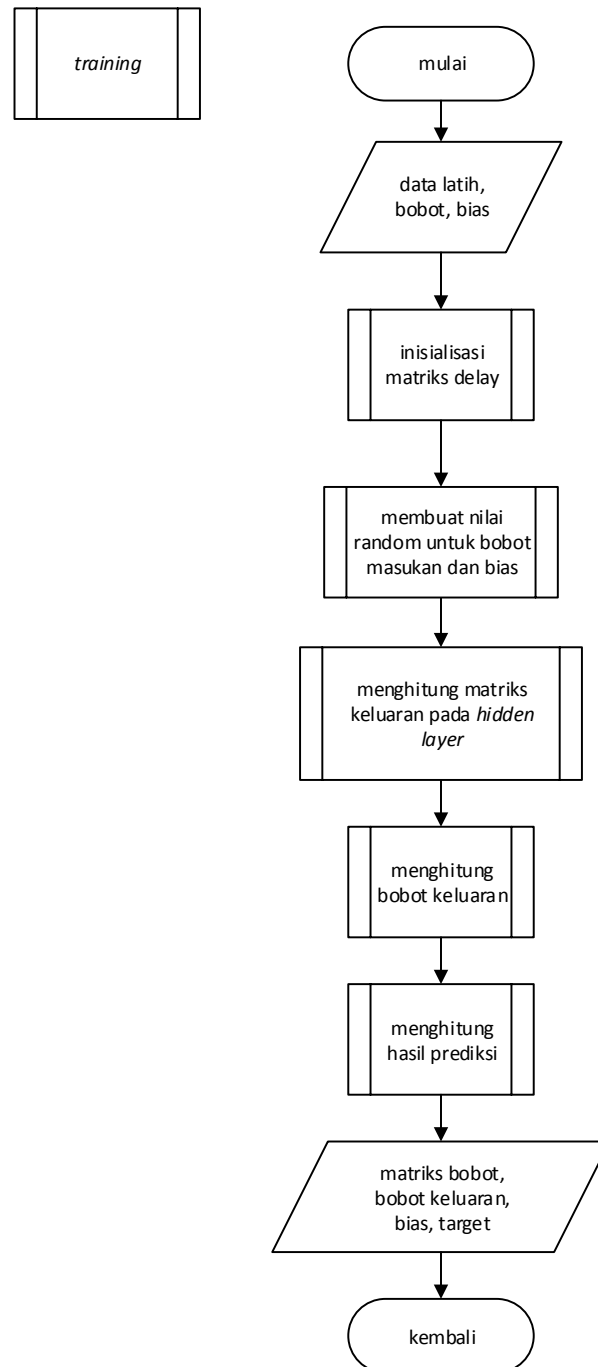
Berdasarkan Gambar 4.3 di atas, langkah-langkah proses normalisasi adalah sebagai berikut:

1. Masukkan berupa data latih.
2. Mencari nilai max dan min pada setiap kolom.

3. Melakukan perhitungan normalisasi dengan Persamaan 2.1.
4. Keluaran berupa data hasil normalisasi dengan range  $[0.1, 0.9]$ .

#### 4.3.2 Training

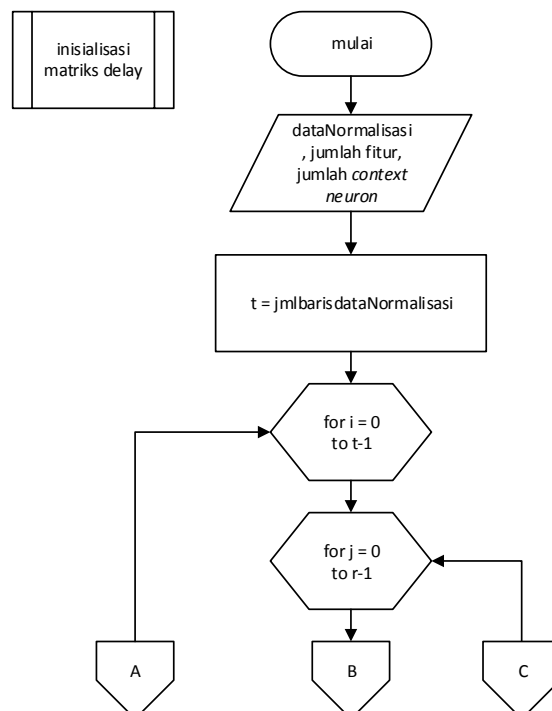
Proses *training* dilakukan untuk memperoleh bobot keluaran yang akan digunakan pada proses *testing*. Diagram alir dari proses *training* ditunjukkan pada Gambar 4.4.

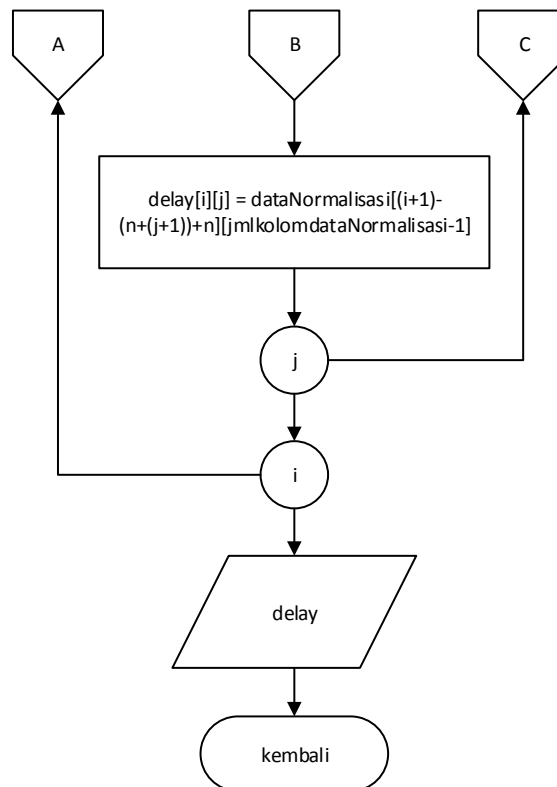


**Gambar 4.4 Diagram Alir Proses *Training***

Berdasarkan Gambar 4.4 di atas, langkah-langkah proses *training* adalah sebagai berikut:

1. Masukkan berupa data latih yang telah dinormalisasi, bobot yang didapatkan secara random dengan range  $[-1,1]$  dan bias yang didapatkan secara random dengan range  $[0,1]$ .
2. Melakukan inisialisasi matriks *delay*  $\delta$  dengan melakukan perhitungan pada Persamaan 2.9. Diagram alir untuk proses inisialisasi matriks *delay* ditunjukkan pada Gambar 4.5.
3. 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*). Diagram alir untuk proses membuat nilai random untuk bobot masukan dan bias ditunjukkan pada Gambar 4.6.
4. Menghitung nilai matriks keluaran pada *hidden layer* dengan menggunakan Persamaan 2.10. Diagram alir untuk proses menghitung nilai matriks keluaran pada *hidden layer* ditunjukkan pada Gambar 4.7.
5. Menghitung bobot keluaran dengan menggunakan Persamaan 2.11, dimana  $H'^{+}$  atau matriks *Moore-Penrose Pseudo Invers* dapat dihitung dengan Persamaan 2.12. Diagram alir untuk proses menghitung bobot keluaran ditunjukkan pada Gambar 4.14.
6. Menghitung hasil prediksi dengan menggunakan Persamaan 2.13. Diagram alir untuk proses menghitung hasil prediksi ditunjukkan pada Gambar 4.17.
7. Keluaran berupa matriks bobot, bobot keluaran, bias dan target.

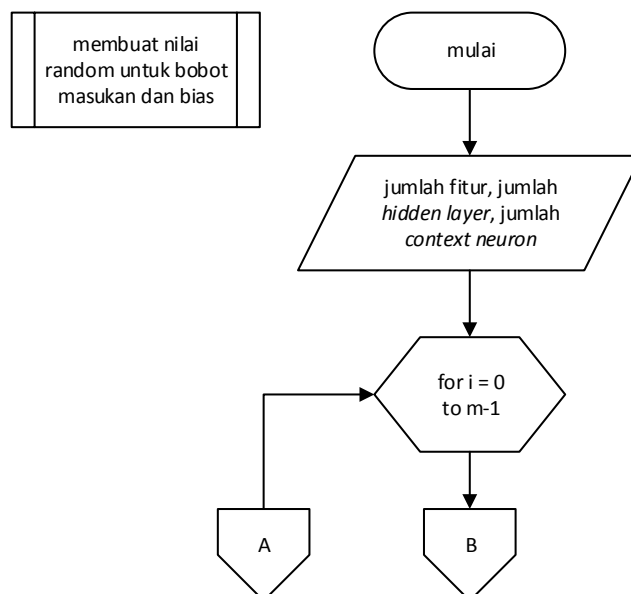


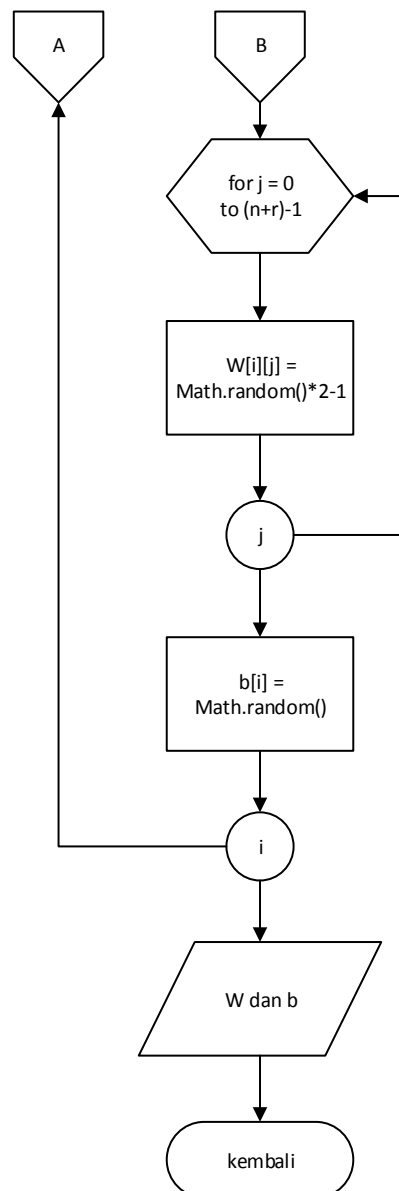


**Gambar 4.5 Diagram Alir Proses Inisialisasi Matriks *Delay***

Berdasarkan Gambar 4.5 di atas, langkah-langkah proses normalisasi adalah sebagai berikut:

1. Masukkan dataNormalisasi, jumlah fitur (n), jumlah *context neuron* (r).
2. Mendapatkan nilai *delay* dengan Persamaan 2.9.
3. Keluaran berupa matriks *delay*.



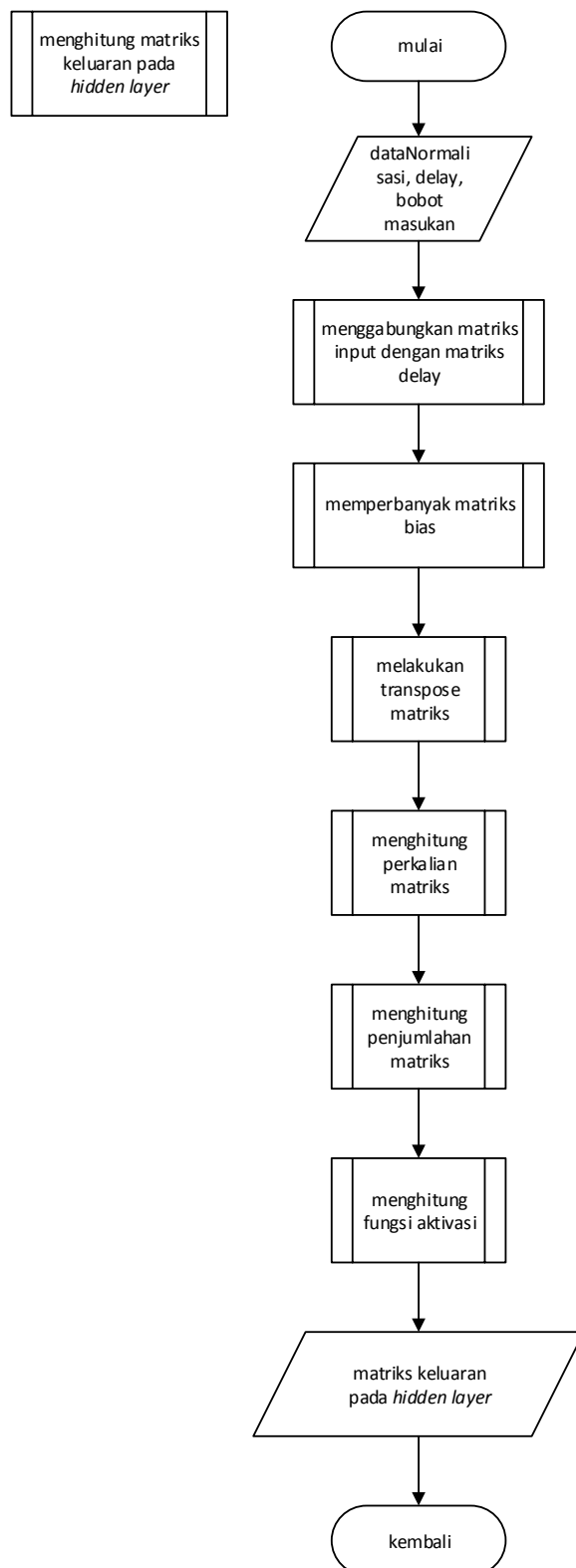


**Gambar 4.6 Diagram Alir Proses Membuat Nilai Random Untuk Bobot Masukan dan Bias**

Berdasarkan Gambar 4.6 di atas, langkah-langkah proses pembuatan nilai random adalah sebagai berikut:

1. Masukkan berupa fitur ( $n$ ), jumlah *hidden layer* ( $m$ ), jumlah *context neuron* ( $r$ ).
2. Dapatkan nilai random dengan range  $[-1,1]$ , untuk menginisialisasi nilai bobot masukan pada matriks  $W$ .
3. Dapatkan nilai random dengan range  $[0,1]$ , untuk menginisialisasi nilai bias pada matriks  $b$ .
4. Keluaran berupa matriks bobot masukan dan bias.

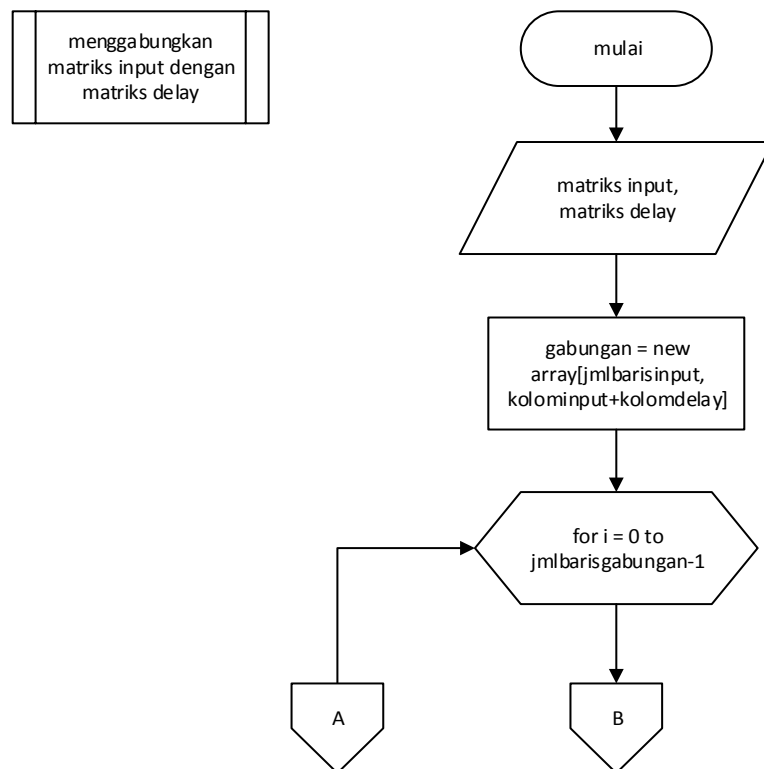


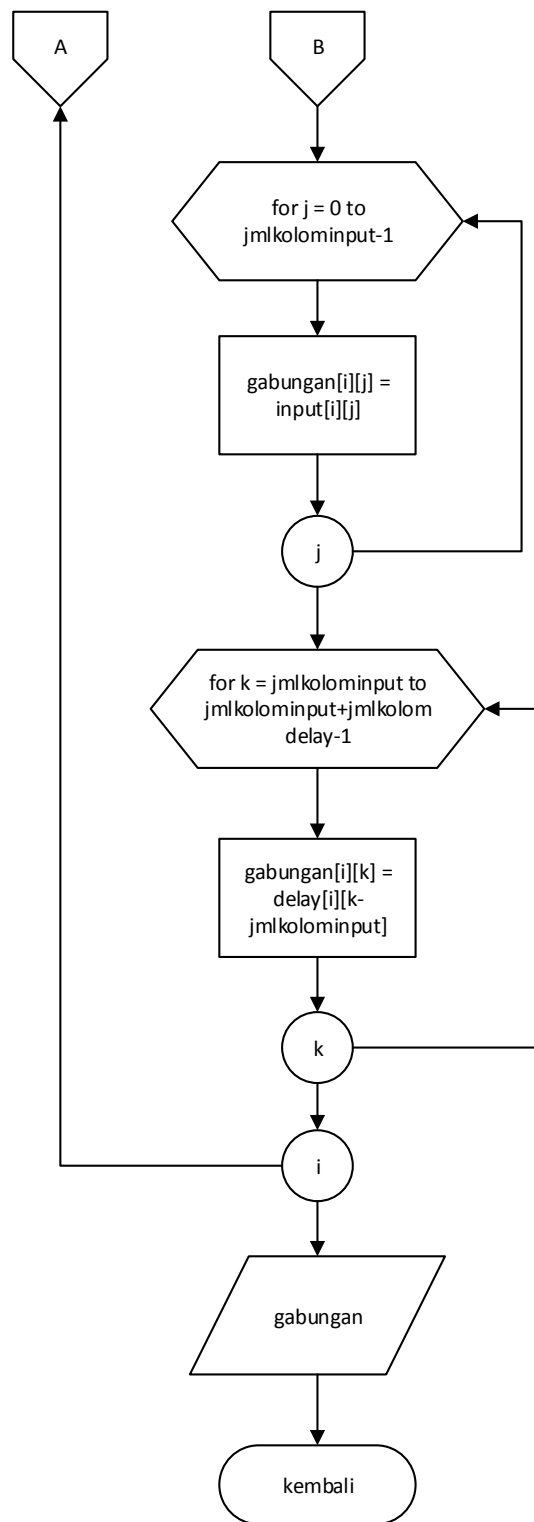


**Gambar 4.7 Diagram Alir Proses Menghitung Matriks Keluaran Pada Hidden Layer**

Berdasarkan Gambar 4.7 di atas, langkah-langkah proses menghitung nilai matriks keluaran pada hidden layer adalah sebagai berikut:

1. Masukkan berupa data normalisasi, *delay*, bobot masukan.
2. Menggabungkan matriks input yang merupakan fitur yang didapatkan dari dataNormalisasi dengan matriks *delay*. Diagram alir untuk proses penggabungan matriks ditunjukkan pada Gambar 4.8.
3. Memperbanyak matriks bias sejumlah baris dari dataNormalisasi. Diagram alir untuk proses memperbanyak matriks ditunjukkan pada Gambar 4.9.
4. Melakukan *transpose* matriks bobot masukan. Diagram alir untuk proses *transpose* matriks ditunjukkan pada Gambar 4.10.
5. Menghitung perkalian antara hasil gabungan matriks dengan bobot masukan yang telah ditranpose. Diagram alir untuk proses perkalian matriks ditunjukkan pada Gambar 4.11.
6. Menghitung penjumlahan antara matriks hasil perkalian dengan matriks bias. Diagram alir untuk proses penjumlahan matriks ditunjukkan pada Gambar 4.12.
7. Menghitung fungsi aktivasi pada matriks hasil penjumlahan. Diagram alir untuk proses menghitung fungsi aktivasi matriks ditunjukkan pada Gambar 4.13.
8. Keluaran berupa matriks keluaran pada hidden layer.

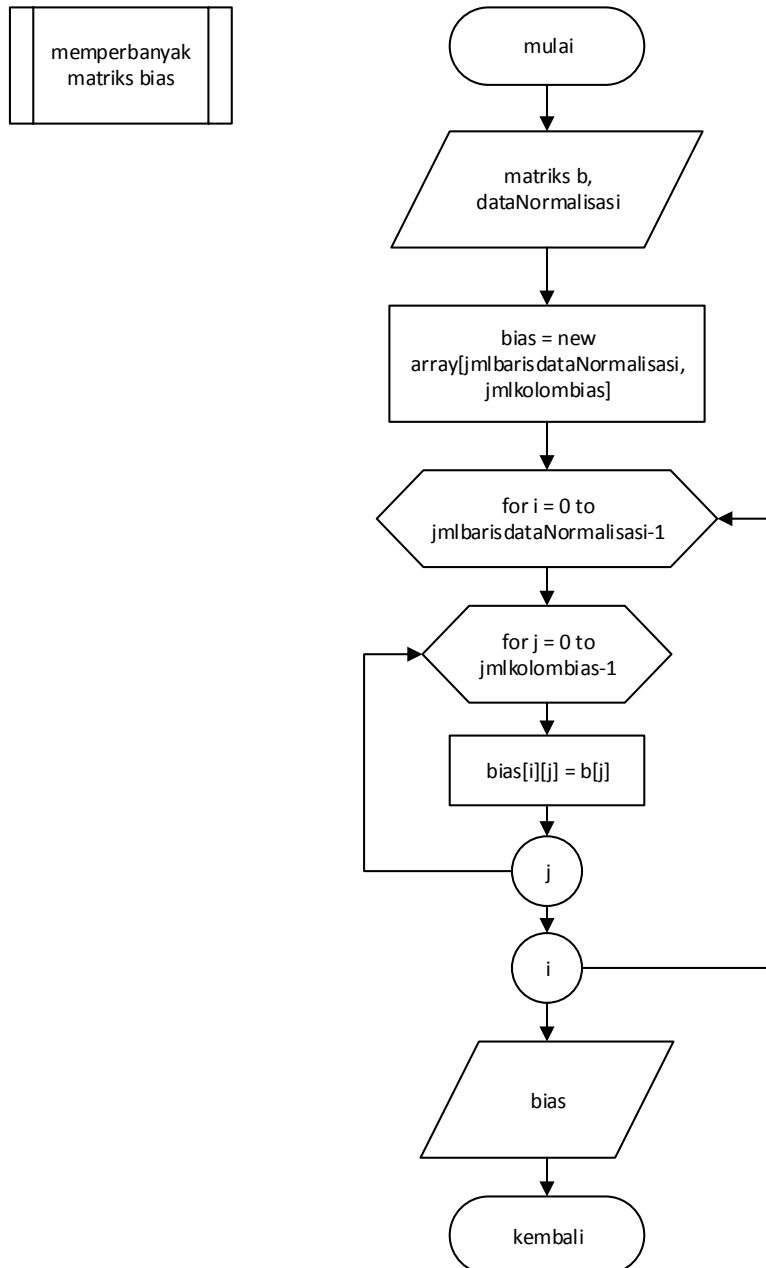




**Gambar 4.8 Diagram Alir Proses Menggabungkan Matriks Input Dengan Matriks *Delay***

Berdasarkan Gambar 4.8 di atas, langkah-langkah proses menggabungkan matriks input dengan matriks *delay* adalah sebagai berikut:

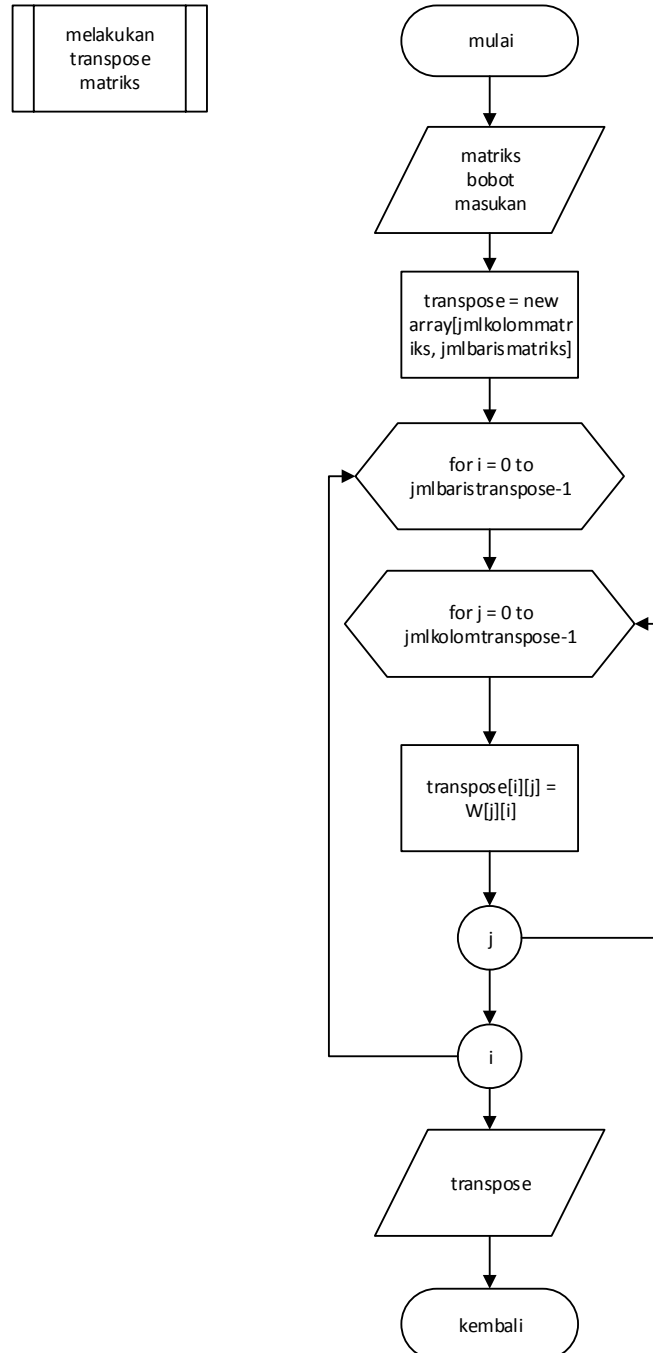
1. Masukkan berupa matriks input dan matriks *delay*.
2. Melakukan pemindahan nilai input ke matriks gabungan.
3. Melakukan pemindahan nilai *delay* ke matriks gabungan.
4. Keluaran berupa matriks gabungan.



**Gambar 4.9 Diagram Alir Proses Memperbanyak Matriks Bias**

Berdasarkan Gambar 4.9 di atas, langkah-langkah proses memperbanyak matriks bias adalah sebagai berikut:

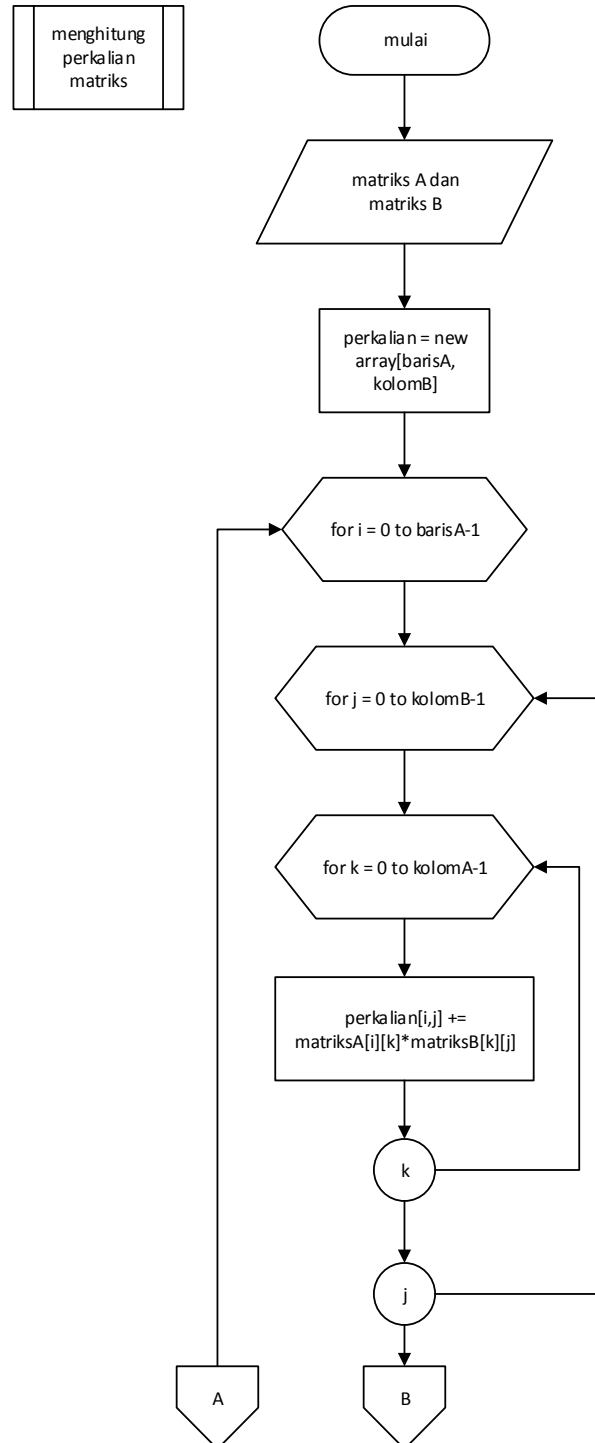
1. Masukkan berupa matriks b (bias) dan matriks dataNormalisasi.
2. Melakukan duplikasi nilai yang ada di matriks b ke matriks bias sejumlah baris dataNormalisasi.
3. Keluaran berupa matriks bias.

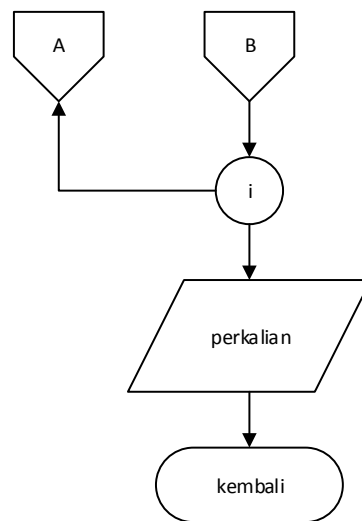


**Gambar 4.10 Diagram Alir Proses Melakukan *Transpose* Matriks**

Berdasarkan Gambar 4.10 di atas, langkah-langkah proses melakukan *transpose* matriks adalah sebagai berikut:

1. Masukkan berupa matriks bobot masukan.
2. Memindahkan matriks W yang semula baris menjadi kolom, yang semula kolom menjadi baris ke matriks *transpose*.
3. Keluaran berupa matriks *transpose*.

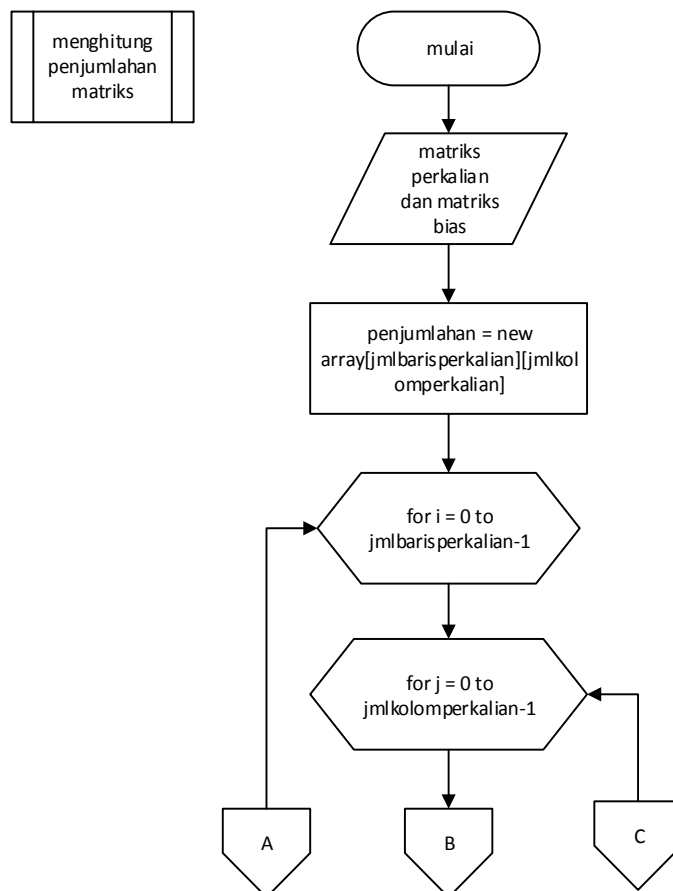


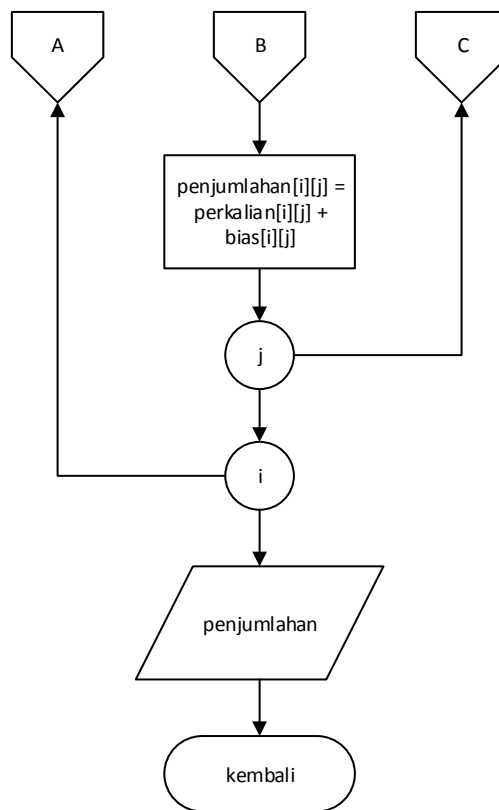


**Gambar 4.11 Diagram Alir Proses Menghitung Perkalian Matriks**

Berdasarkan Gambar 4.11 di atas, langkah-langkah proses menghitung perkalian matriks adalah sebagai berikut:

1. Masukkan berupa matriks A dan matriks B.
2. Melakukan perkalian baris matriks A terhadap kolom matriks B.
3. Keluaran berupa matriks perkalian.

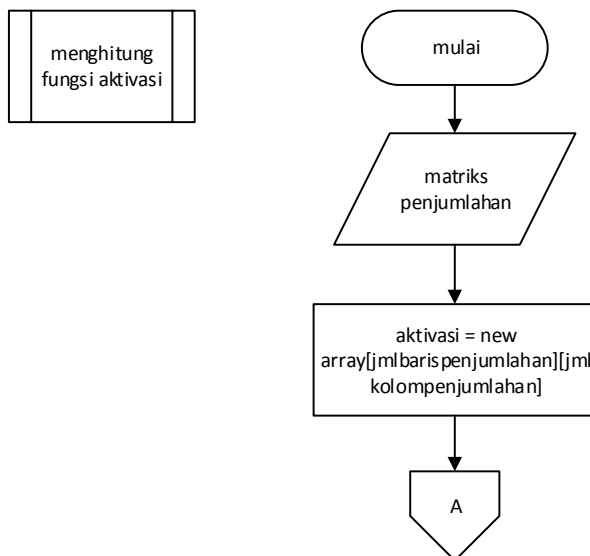




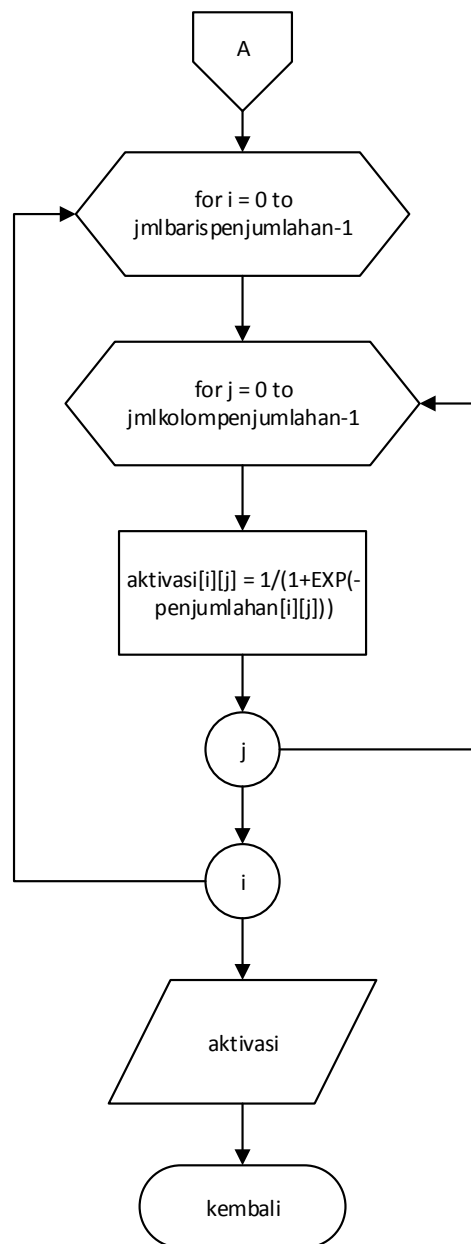
**Gambar 4.12 Diagram Alir Proses Menghitung Penjumlahan Matriks**

Berdasarkan Gambar 4.12 di atas, langkah-langkah proses menghitung penjumlahan matriks adalah sebagai berikut:

1. Masukkan berupa matriks perkalian dan matriks bias.
2. Melakukan penjumlahan antara matriks perkalian dengan matriks bias pada setiap baris dan kolom yang sama.
3. Keluaran berupa matriks penjumlahan.



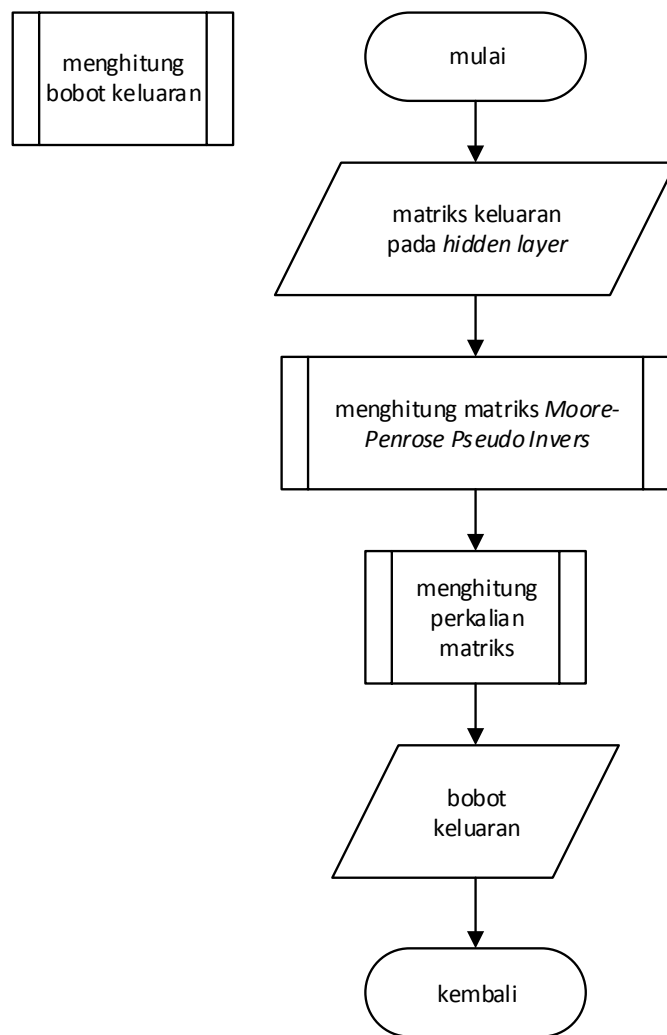




**Gambar 4.13 Diagram Alir Proses Menghitung Fungsi Aktivasi**

Berdasarkan Gambar 4.13 di atas, langkah-langkah proses menghitung fungsi aktivasi adalah sebagai berikut:

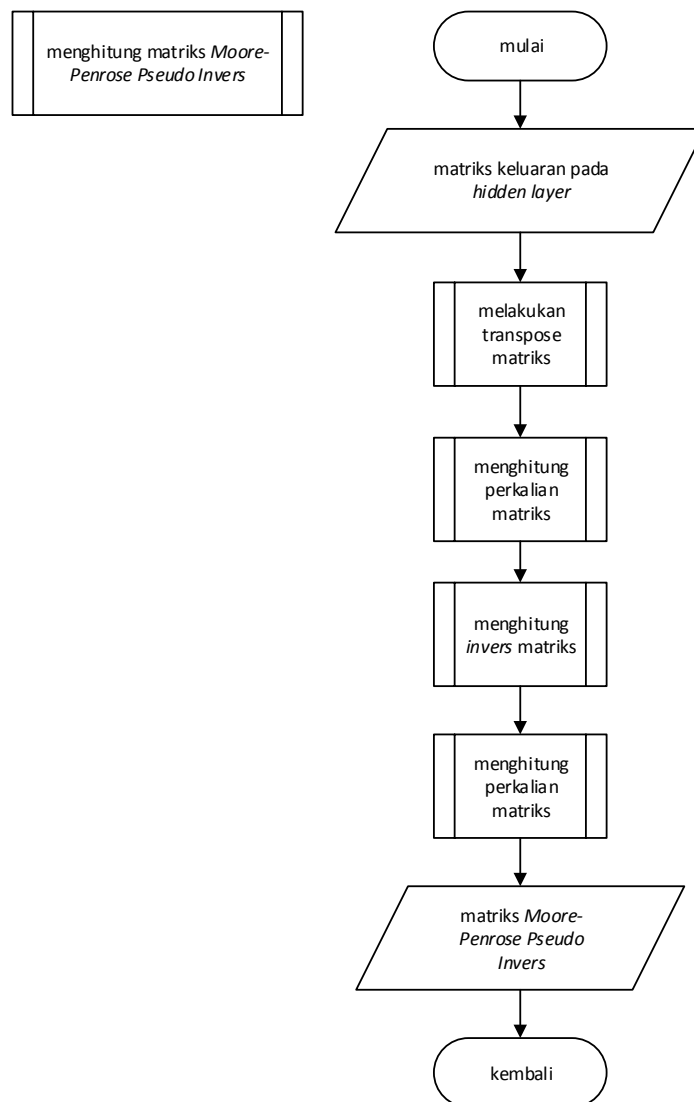
1. Masukkan berupa matriks penjumlahan.
2. Melakukan aktivasi pada setiap data yang ada pada matriks penjumlahan.
3. Keluaran berupa matriks aktivasi, yang nilai tersebut juga menjadi matriks keluaran pada *hidden layer*.



**Gambar 4.14 Diagram Alir Proses Menghitung Bobot Keluaran**

Berdasarkan Gambar 4.14 di atas, langkah-langkah proses menghitung bobot keluaran adalah sebagai berikut:

1. Masukkan berupa matriks keluaran pada *hidden layer*.
2. Menghitung matriks *Moore-Penrose Pseudo Invers*. Diagram alir untuk proses menghitung matriks *Moore-Penrose Pseudo Invers* ditunjukkan pada Gambar 4.15.
3. Menghitung perkalian matriks antara matriks *Moore-Penrose Pseudo Invers* dengan matriks target.
4. Keluaran berupa matriks bobot keluaran.

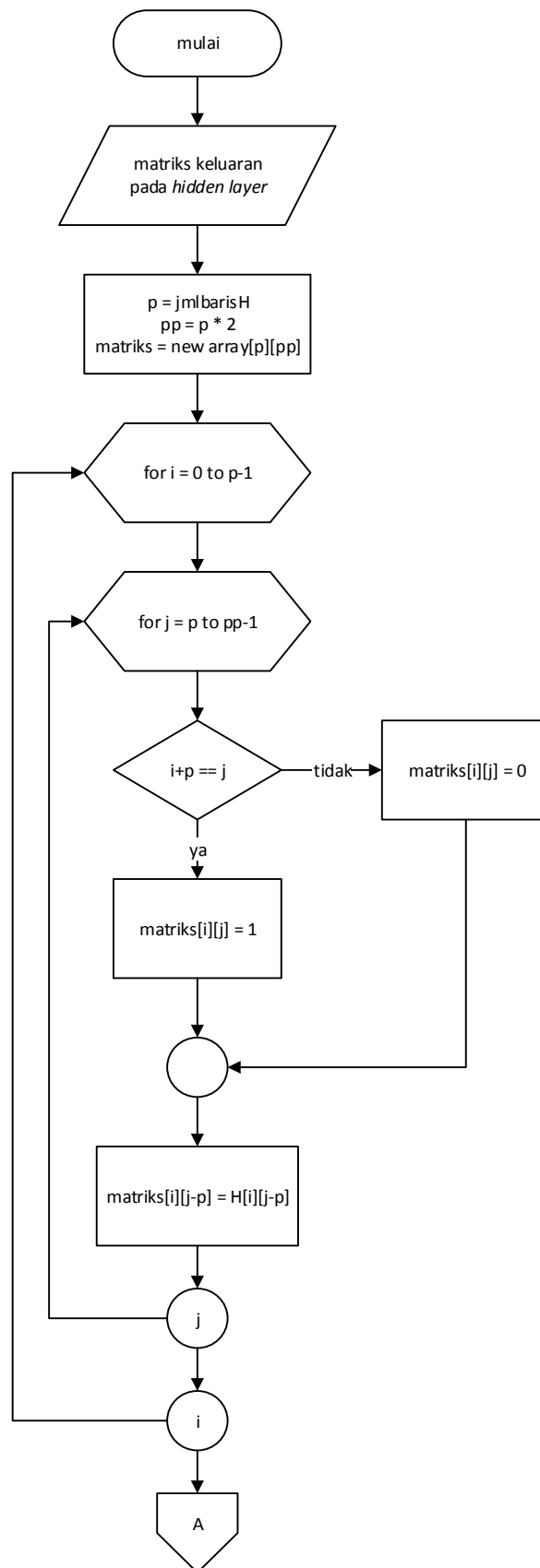


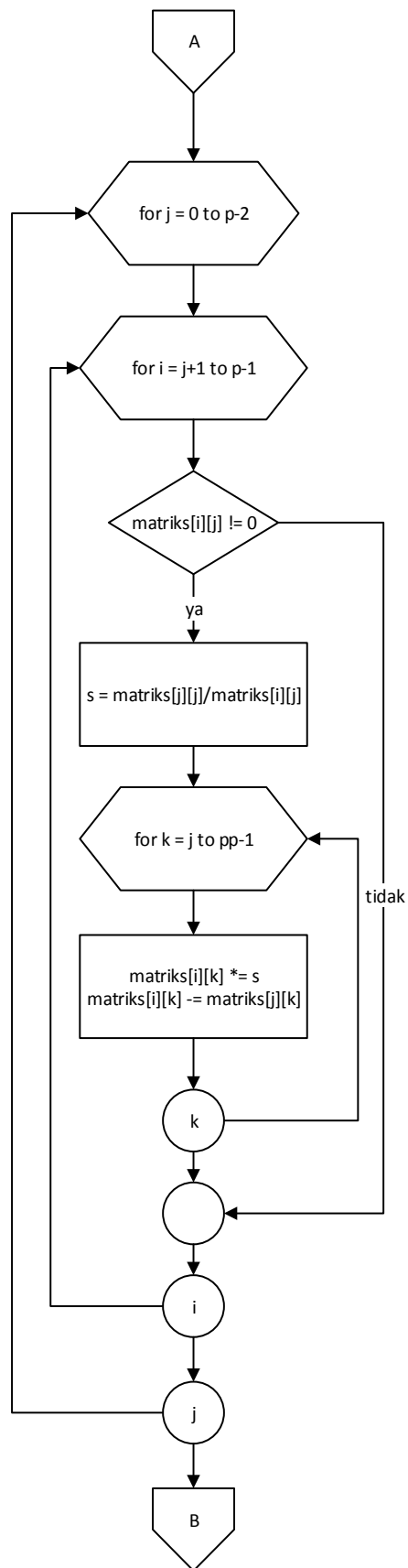
**Gambar 4.15 Diagram Alir Proses Menghitung Matriks *Moore-Penrose Pseudo Invers***

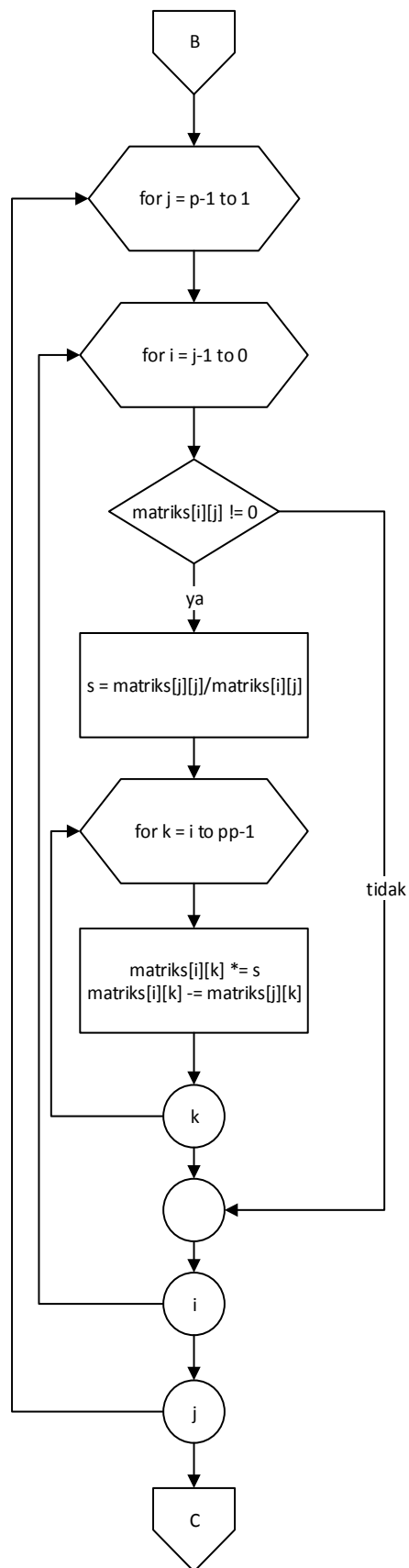
Berdasarkan Gambar 4.15 di atas, langkah-langkah proses menghitung matriks *Moore-Penrose Pseudo Invers* adalah sebagai berikut:

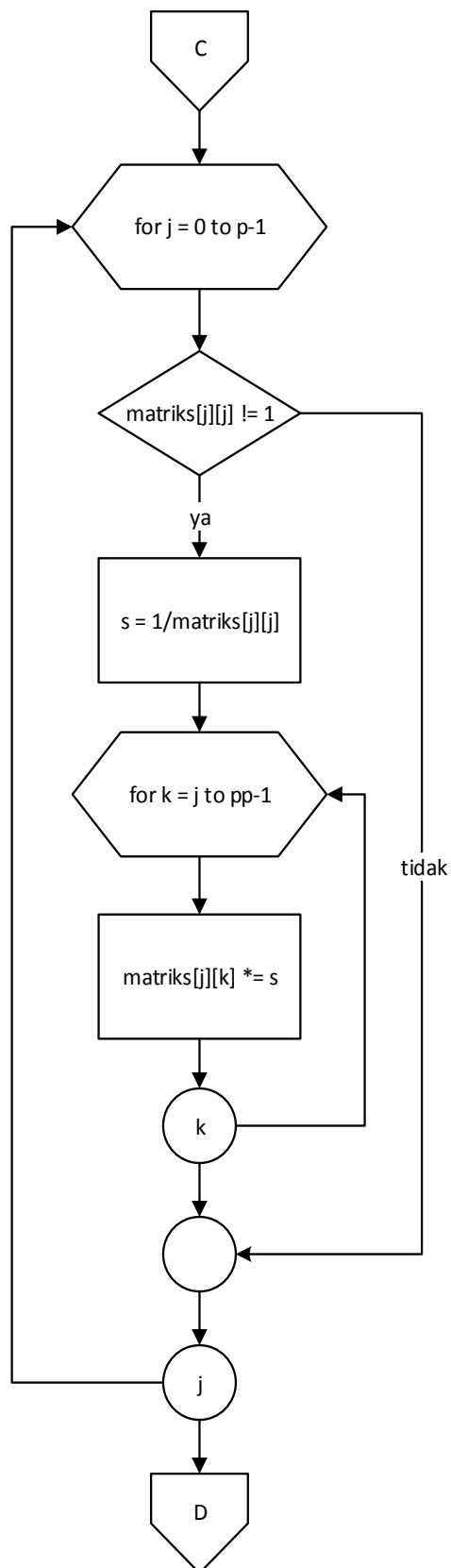
1. Masukkan berupa matriks keluaran *hidden layer*.
2. Melakukan *transpose* matriks keluaran *hidden layer*.
3. Menghitung perkalian matriks antara *tranpose* dari matriks keluaran *hidden layer* dengan matriks keluaran *hidden layer*.
4. Menghitung *invers* matriks hasil perkalian langkah ke-3. Diagram alir untuk proses menghitung *invers* matriks ditunjukkan pada Gambar 4.16.
5. Menghitung perkalian matriks *invers* dengan tranpose dari matriks keluaran *hidden layer*.
6. Keluaran berupa matriks *Moore-Penrose Pseudo Invers*.

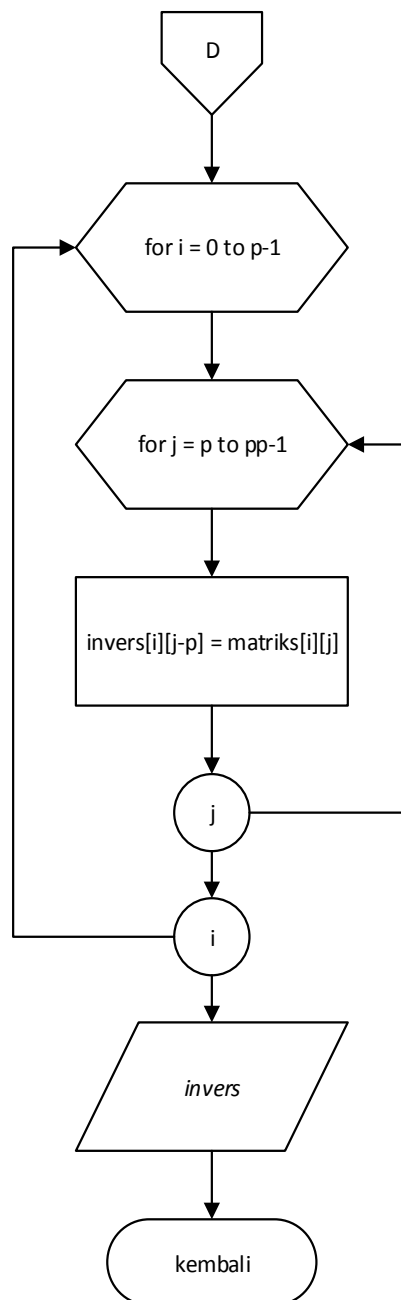
menghitung invers matriks
------------------------------











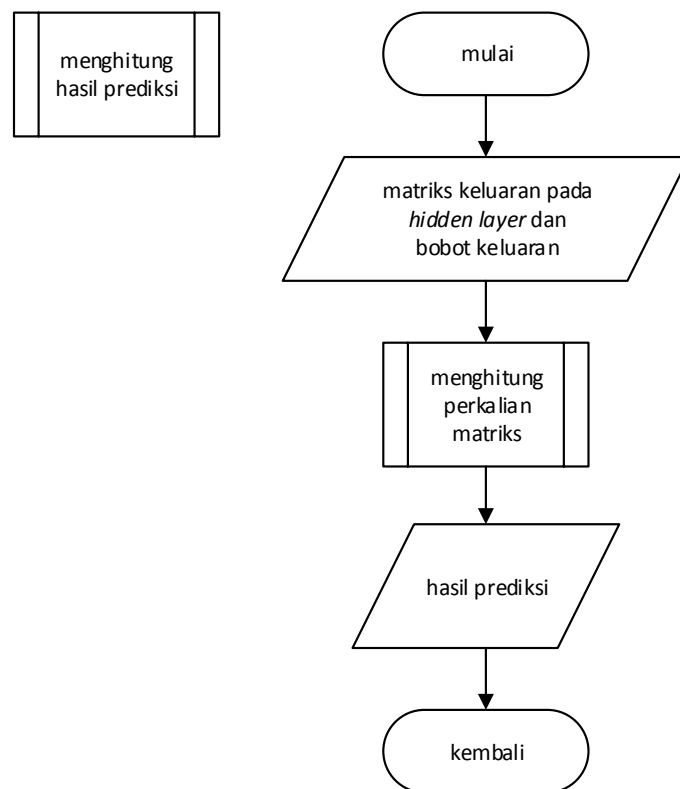
**Gambar 4.16 Diagram Alir Proses Menghitung *Invers* Matriks**

Berdasarkan Gambar 4.16 di atas, langkah-langkah proses menghitung *Invers* matriks adalah sebagai berikut:

1. Masukkan berupa matriks keluaran *hidden layer*.
2. Menggabungkan matriks keluaran *hidden layer* dengan identitas
3. Proses perhitungan matriks OBE langkah ke-1 yaitu mengosongkan nilai yang ada dikolom bagian kiri bawah nya diagonal.



4. Proses perhitungan matriks OBE langkah ke-2 yaitu mengosongkan nilai yang ada dikolom bagian kanan atas nya diagonal.
5. Proses perhitungan matriks OBE langkah ke-3 yaitu menjadikan nilai 1 pada bagian diagonal.
6. Mendapatkan hasil *invers* dan menyimpannya pada matriks *invers*.
7. Keluaran berupa matriks *invers*.



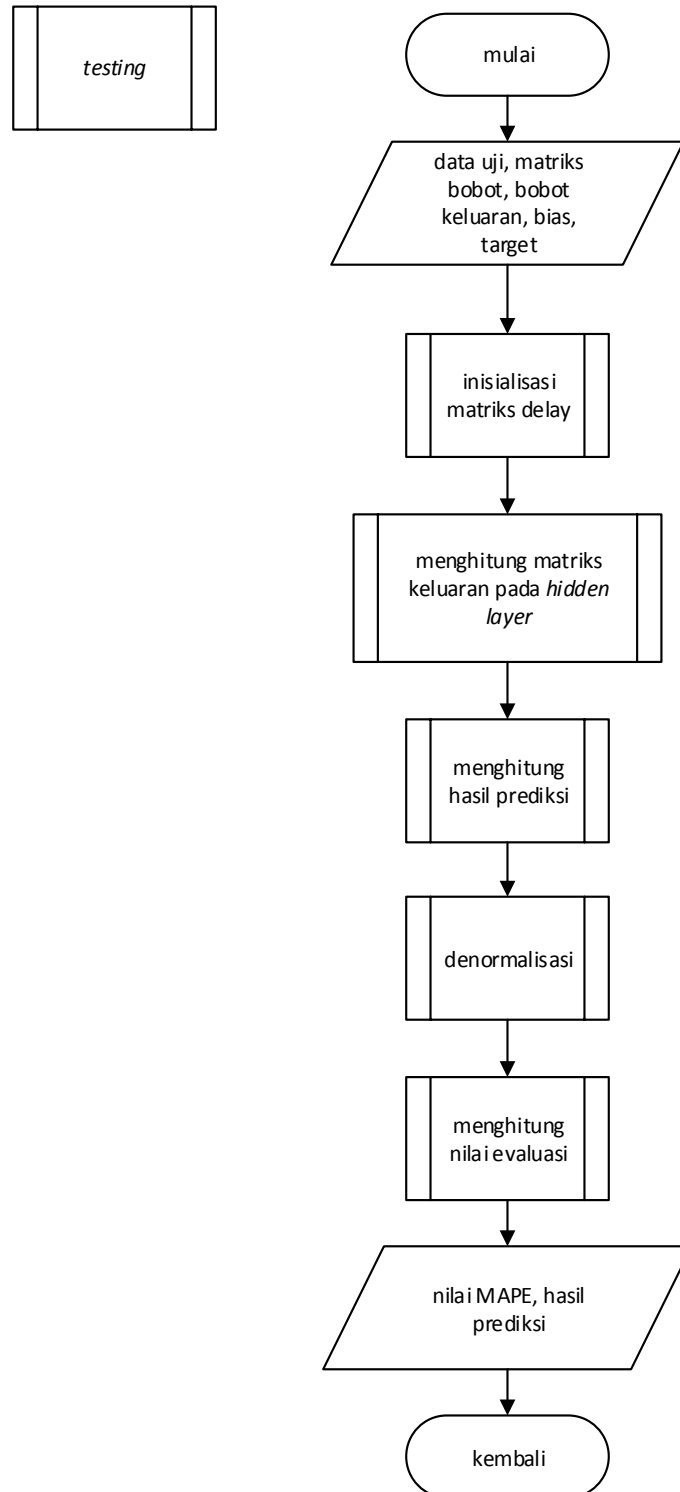
**Gambar 4.17 Diagram Alir Proses Menghitung Hasil Prediksi**

Berdasarkan Gambar 4.17 di atas, langkah-langkah proses menghitung hasil prediksi adalah sebagai berikut:

1. Masukkan berupa matriks keluaran hidden layer dan bobot keluaran.
2. Menghitung perkalian matriks antara matriks keluaran hidden layer dengan bobot keluaran.
3. Keluaran berupa hasil prediksi.
4. Mengembalikan hasil prediksi untuk ditampilkan pada program.

### 4.3.3 Testing

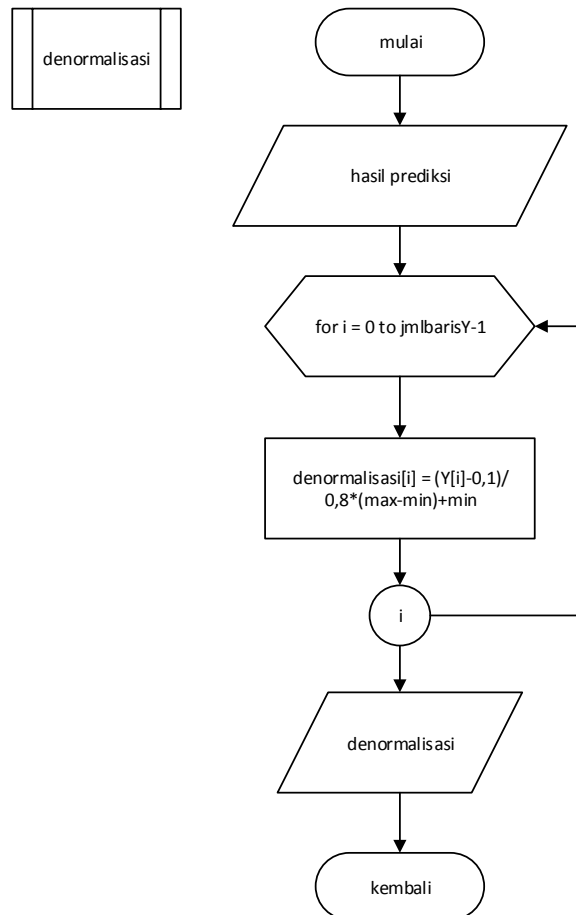
Proses *testing* dilakukan untuk menguji hasil dari proses *testing*. Diagram alir dari proses *testing* ditunjukkan pada Gambar 4.18.



**Gambar 4.18 Diagram Alir Proses *Testing***

Berdasarkan Gambar 4.18 di atas, langkah-langkah proses *testing* adalah sebagai berikut:

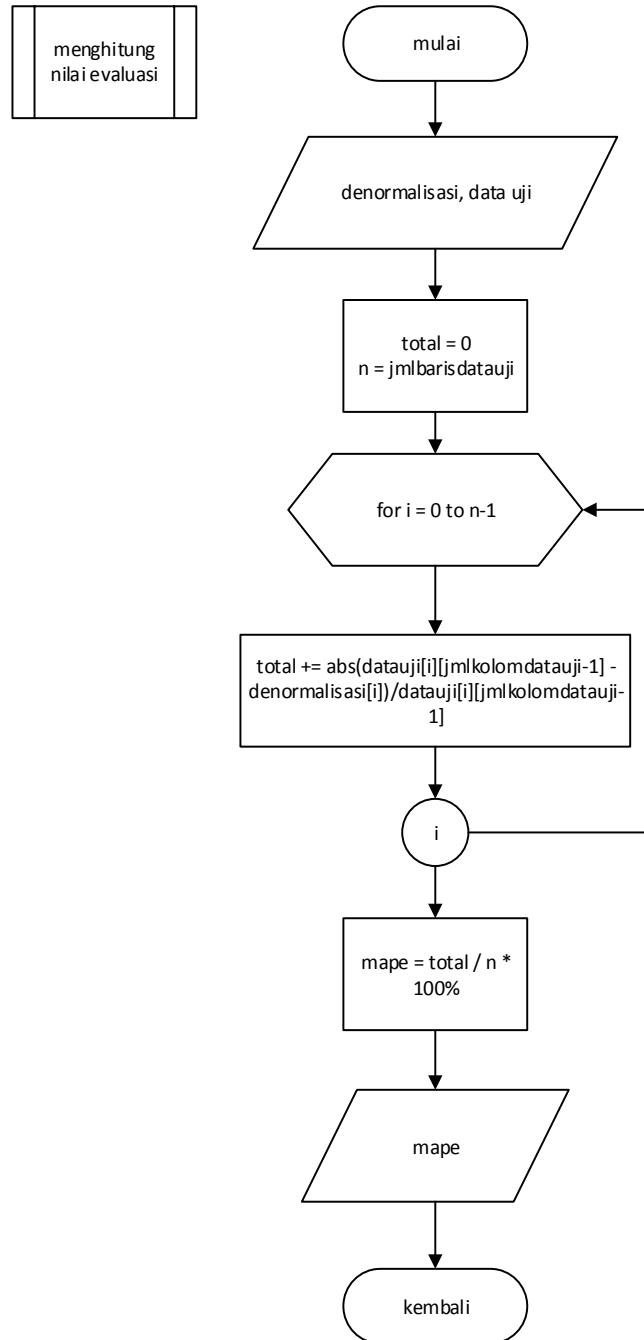
1. Masukkan berupa data uji, matriks bobot, bobot keluaran dan bias, yang didapatkan pada proses *testing*.
2. Melakukan inisialisasi matriks *delay*  $\delta$  dengan melakukan perhitungan pada Persamaan 2.9.
3. Menghitung nilai matriks keluaran pada *hidden layer* dengan menggunakan Persamaan 2.14.
4. Menghitung hasil prediksi dengan menggunakan Persamaan 2.13.
5. Melakukan denormalisasi pada hasil prediksi dengan menggunakan Persamaan 2.2. Diagram alir untuk proses denormalisasi ditunjukkan pada Gambar 4.19.
6. Menghitung nilai evaluasi dengan menggunakan Persamaan 2.15. Diagram alir untuk proses menghitung nilai evaluasi ditunjukkan pada Gambar 4.20.
7. Keluaran berupa nilai MAPE dan hasil prediksi.



**Gambar 4.19 Diagram Alir Proses Denormalisasi**

Berdasarkan Gambar 4.19 di atas, langkah-langkah proses denormalisasi adalah sebagai berikut:

1. Masukkan berupa matriks hasil prediksi.
2. Melakukan proses denormalisasi dengan menggunakan nilai max dan min data target pada proses normalisasi.
3. Keluaran berupa hasil denormalisasi.



**Gambar 4.20 Diagram Alir Proses Menghitung Nilai Evaluasi**

Berdasarkan Gambar 4.20 di atas, langkah-langkah proses menghitung nilai evaluasi adalah sebagai berikut:

1. Masukkan berupa matriks data uji dan denormalisasi.
2. Menghitung nilai mape dengan menggunakan Persamaan 2.15.
3. Keluaran berupa nilai mape.

#### 4.4 Perhitungan Manual

Perhitungan manual merupakan contoh perhitungan dari perancangan sistem yang dibuat dengan tujuan untuk mengetahui kebenaran dari perhitungan sistem. Langkah-langkah penyelesaian RELMNN dalam perhitungan manual adalah sebagai berikut:

1. Menentukan jumlah fitur, *hidden neuron*, dan *context neuron*

Dalam manualisasi ini, inisialisasi jumlah fitur ( $n$ ) yang digunakan sebanyak 3, sedangkan jumlah *hidden neuron* ( $m$ ) sebanyak 2, dan jumlah *context neuron* ( $r$ ) sebanyak 2. Sedangkan *datasets* dengan jumlah 15 data dibagi menjadi 2 bagian yaitu untuk *data training* sebanyak 10 data dan *data testing* sebanyak 5 data.

2. Proses Normalisasi

**Langkah 1:** Normalisasi matriks *datasets* dengan menggunakan Persamaan 2.1. Jumlah *datasets* yang digunakan sebanyak 15 data. Berikut contoh perhitungan normalisasi.

**Langkah 1.1:** Mencari nilai *minimum* dan *maximum* nilai tukar dari keseluruhan *datasets*. Tabel 4.2 menunjukkan nilai *minimum* dan *maximum*.

**Tabel 4.2 Nilai *Maximum* dan *Minimum***

	Nilai Tukar
Min	13311
Max	13407

**Langkah 1.2:** Menghitung nilai normalisasi pada setiap data. Tabel 4.3 menunjukkan hasil normalisasi keseluruhan data latih. Berikut contoh perhitungan nilai normalisasi.

$$y_{1,1} = \frac{x - \min}{\max - \min} (0,8) + 0,1$$

$$y_{1,1} = \frac{13360 - 13311}{13407 - 13311} (0,8) + 0,1$$

$$y_{1,1} = 0,508333$$

**Tabel 4.3 Normalisasi *Datasets***

Data ke- <i>i</i>	X1	X2	X3	T
1	0,508333	0,358333	0,100000	0,341667
2	0,358333	0,100000	0,341667	0,258333
3	0,100000	0,341667	0,258333	0,258333
4	0,341667	0,258333	0,258333	0,466667
5	0,258333	0,258333	0,466667	0,391667
6	0,258333	0,466667	0,391667	0,341667
7	0,466667	0,391667	0,341667	0,700000
8	0,391667	0,341667	0,700000	0,616667
9	0,341667	0,700000	0,616667	0,200000
10	0,700000	0,616667	0,200000	0,200000
11	0,616667	0,200000	0,200000	0,197917
12	0,200000	0,200000	0,197917	0,322917
13	0,200000	0,197917	0,322917	0,900000
14	0,197917	0,322917	0,900000	0,791667
15	0,322917	0,900000	0,791667	0,533333

### 3. Proses *training*

**Langkah 1:** Inisialisasi matriks *delay*  $\delta$  dengan Persamaan 2.9. Tabel 4.4 menunjukkan hasil inisialisasi matriks *delay*  $\delta$  dimana *t* adalah urutan data, *r* adalah urutan *context neuron*. Berikut contoh menghitung nilai *delay*  $\delta$ .

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

$$\delta_{1,1} = T(1 - (3 + 1) + 3)$$

$$\delta_{1,1} = T(0) = 0$$

$$\delta_{2,1} = T(2 - (3 + 1) + 3)$$

$$\delta_{2,1} = T(1) = 0,341667$$

**Tabel 4.4 Matriks *Delay* Proses *Training***

Data ke- <i>t</i>	<i>r1</i>	<i>r2</i>
1	0,000000	0,000000
2	0,341667	0,000000
3	0,258333	0,341667
4	0,258333	0,258333
5	0,466667	0,258333

**Tabel 4.4 Matriks *Delay* Proses *Training***

Data ke-t	r1	r2
6	0,391667	0,466667
7	0,341667	0,391667
8	0,700000	0,341667
9	0,616667	0,700000
10	0,200000	0,616667

**Langkah 2:** Membuat nilai random untuk matriks  $W'_{m(n+r)}$  sebagai bobot masukan (*input weight*) dengan range  $[-1;1]$ , dalam bentuk *array* ukuran  $m$  (jumlah *hidden neuron*) x  $n+r$  (jumlah fitur + *context neurons*). Kemudian buat nilai random untuk matriks bias  $b$  dengan range  $[0;1]$  dalam ukuran  $1 \times$  (jumlah *hidden neuron*).

**Langkah 2.1:** Membuat nilai random untuk matriks bobot. Tabel 4.5 menunjukkan hasil random untuk nilai bobot.

**Tabel 4.5 Matriks Bobot**

$M \backslash N+R$	1	2	3	4	5
1	-0,956910	0,763482	0,476625	0,291437	0,922641
2	-0,863870	-0,278740	-0,415650	0,517960	-0,431670

**Langkah 2.2:** Membuat nilai random untuk bias. Tabel 4.6 menunjukkan hasil random untuk nilai bobot.

**Tabel 4.6 Matriks Bias**

	1	2
1	0,488982	0,286403

**Langkah 3:** Menghitung nilai matriks keluaran pada *hidden layer* dengan menggunakan Persamaan 2.10.

**Langkah 3.1:** Menggabungkan matriks data latih yang hanya berisi fitur ( $x$ ) dengan matriks *delay*, dengan cara menambahkan kolom pada matriks data latih dengan matriks *delay*. Tabel 4.7 menunjukkan hasil penggabungan antara dua matriks tersebut.

**Tabel 4.7 Gabungan Matriks  $X$  Dengan Matriks *Delay* Proses *Training***

No	x1	x2	x3	d1	d2
1	0,508333	0,358333	0,100000	0,000000	0,000000
2	0,358333	0,100000	0,341667	0,341667	0,000000

**Tabel 4.7 Gabungan Matriks  $X$  Dengan Matriks  $\text{Delay}$  Proses  $\text{Training}$**

No	x1	x2	x3	d1	d2
3	0,100000	0,341667	0,258333	0,258333	0,341667
4	0,341667	0,258333	0,258333	0,258333	0,258333
5	0,258333	0,258333	0,466667	0,466667	0,258333
6	0,258333	0,466667	0,391667	0,391667	0,466667
7	0,466667	0,391667	0,341667	0,341667	0,391667
8	0,391667	0,341667	0,700000	0,700000	0,341667
9	0,341667	0,700000	0,616667	0,616667	0,700000
10	0,700000	0,616667	0,200000	0,200000	0,616667

**Langkah 3.2:** Memperbanyak matriks bias sebanyak jumlah data latih, dengan cara memperbanyak baris matriks bias sejumlah data latih dengan nilai yang sama. Tabel 4.8 menunjukkan hasil dari matriks bias yang sudah diperbanyak.

**Tabel 4.8 Matriks Bias Proses  $\text{Training}$**

	1	2
1	0,488982	0,286403
2	0,488982	0,286403
3	0,488982	0,286403
4	0,488982	0,286403
5	0,488982	0,286403
6	0,488982	0,286403
7	0,488982	0,286403
8	0,488982	0,286403
9	0,488982	0,286403
10	0,488982	0,286403

**Langkah 3.3:** Menghitung perkalian matriks gabungan dengan *transpose* matriks bobot. Tabel 4.9 menunjukkan hasil dari perkalian matriks tersebut. Berikut contoh dari perhitungannya.

$$\begin{aligned}
 ([x, \delta] * W'^T)_{1,1} &= [x, \delta] * W'^T \\
 &= (0,508333 * -0,95691) + (0,358333 * 0,763482) \\
 &\quad + (0,100000 * 0,476625) \\
 &\quad + (0,000000 * 0,291437) \\
 &\quad + (0,000000 * 0,922641) \\
 &= -0,165186
 \end{aligned}$$



**Tabel 4.9 Matriks Hasil Perkalian**

	1	2
1	-0,165186	-0,580581
2	-0,004123	-0,302472
3	0,678817	-0,302680
4	0,307054	-0,452248
5	0,546809	-0,358944
6	0,840480	-0,514620
7	0,476261	-0,646427
8	0,738946	-0,509456
9	1,326980	-0,729351
10	0,523551	-1,022333

**Langkah 3.4:** Menghitung penjumlahan matriks hasil perkalian dengan matriks bias. Tabel 4.10 menunjukkan hasil dari penjumlahan matriks tersebut. Berikut contoh dari perhitungannya.

$$\begin{aligned}
 ([x, \delta] * W'^T + b(\text{ones}(i, 1), :))_{1,1} &= ([x, \delta] * W'^T + b(\text{ones}(i, 1), :)) \\
 &= -0,165186 + 0,488982 \\
 &= 0,323796
 \end{aligned}$$

**Tabel 4.10 Matriks Hasil Penjumlahan**

	1	2
1	0,323796	-0,294178
2	0,484859	-0,016069
3	1,167799	-0,016277
4	0,796036	-0,165845
5	1,035791	-0,072541
6	1,329462	-0,228217
7	0,965243	-0,360024
8	1,227928	-0,223053
9	1,815962	-0,442948
10	1,012533	-0,735930

**Langkah 3.5:** Menghitung fungsi aktivasi dari hasil penjumlahan untuk mendapatkan matriks keluaran pada *hidden layer*. Tabel 4.11 menunjukkan hasil matriks keluaran pada *hidden layer*. Berikut contoh dari perhitungannya.

$$H' = \frac{1}{1 + \exp(-([x, \delta]W''^T + b(\text{ones}(i, 1), :)))}$$

$$H'_{1,1} = \frac{1}{1 + \exp(-0,323796)} = 0,580249$$

**Tabel 4.11 Matriks keluaran *hidden layer***

$H'$	1	2
1	0,580249	0,426981
2	0,618895	0,495983
3	0,762747	0,495931
4	0,689126	0,458633
5	0,738037	0,481873
6	0,790752	0,443192
7	0,724170	0,410954
8	0,773456	0,444467
9	0,860081	0,391039
10	0,733516	0,323895

**Langkah 4:** Menghitung  $\hat{\beta}'$  sebagai bobot keluaran setelah proses *recurrent* dengan menggunakan Persamaan 2.11, dimana  $H'^+$  atau matriks *Moore-Penrose Pseudo Invers* dapat dihitung dengan Persamaan 2.12.

**Langkah 4.1:** Menghitung perkalian antara *transpose* dari matriks keluaran hidden layer dengan matriks keluaran hidden layer. Tabel 4.12 menunjukkan hasil dari perkalian tersebut. Berikut contoh dari perhitungannya.

$$\begin{aligned} (H'^T H')_{1,1} &= H'^T H' \\ &= (0,580249 * 0,580249) + (0,618895 * 0,618895) + \dots \\ &\quad + (0,733516 * 0,733516) \\ &= 5,346824 \end{aligned}$$

**Tabel 4.12 Matriks Hasil Perkalian**

$H'^T H'$	1	2
1	5,346824	3,170420
2	3,170420	1,937478

**Langkah 4.2:** Menghitung matriks *invers* dari hasil perhitungan  $H'^T H'$ . Tabel 4.13 menunjukkan hasil dari *invers* matriks tersebut. Berikut langkah-langkah dari perhitungan *invers* dengan menggunakan perhitungan OBE.

1. Bentuk matriks identitas ( $I$ )

$$[H'^T H' | I] = \begin{bmatrix} 5,346824 & 3,170420 \\ 3,170420 & 1,937478 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

2.  $\frac{5,346824}{3,170420} R2 + (-R1) \rightarrow R2$

$$[H'^T H' | I] = \begin{bmatrix} 5,346824 & 3,170420 \\ 0 & 0,097081 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -1 & 1,686472 \end{bmatrix}$$

3.  $\frac{0,097081}{3,170420} R1 + (-R2) \rightarrow R1$

$$[H'^T H' | I] = \begin{bmatrix} 0,163725 & 0 \\ 0 & 0,097081 \end{bmatrix} \begin{bmatrix} 1,030621 & -1,686472 \\ -1 & 1,686472 \end{bmatrix}$$

4.  $\frac{1}{0,163725} R1 \rightarrow R1$

$$[H'^T H' | I] = \begin{bmatrix} 1 & 0 \\ 0 & 0,097081 \end{bmatrix} \begin{bmatrix} 6,294826 & -10,300631 \\ -1 & 1,686472 \end{bmatrix}$$

5.  $\frac{1}{0,097081} R2 \rightarrow R2$

$$[H'^T H' | I] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 6,294826 & -10,300631 \\ -10,300631 & 17,371726 \end{bmatrix}$$

**Tabel 4.13 Matriks Invers**

$(H^T H)^{-1}$	1	2
1	6,294826	-10,300631
2	-10,300631	17,371726

**Langkah 4.3:** Menghitung matriks *Moore-Penrose Pseudo Invers* dengan mengalikan matriks *invers* dengan matriks *transpose* dari  $H'$ . Tabel 4.14 menunjukkan hasil dari perkalian tersebut. Berikut contoh dari perhitungannya.

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

$$H'^+_{1,1} = (6,294826 * 0,580249) + (-10,300631 * 0,426981)$$

$$H'^+_{1,1} = -0,745611$$

**Tabel 4.14 Matriks Moore-Penrose Pseudo Invers**

No	1	2	3	...	10
1	-0,745611	-1,213105	-0,307041	...	1,281033
2	1,440471	2,241076	0,758398	...	-1,929063

**Langkah 4.4:** Menghitung  $\hat{\beta}'$  sebagai bobot keluaran. Tabel 4.15 menunjukkan matriks dari bobot keluaran. Berikut contoh dari perhitungannya.

$$\hat{\beta}' = H'^+ T$$

$$\hat{\beta}'_{1,1} = (-0,745611 * 0,341667) + (-1,213105 * 0,258333) + \dots \\ + (1,281033 * 0,200000)$$

$$\hat{\beta}'_{1,1} = 0,129116$$

**Tabel 4.15 Matriks Bobot Keluaran**

$\hat{\beta}'$	1
1	0,129116
2	0,646049

**Langkah 5:** Menghitung hasil prediksi dengan menggunakan Persamaan 2.13. Tabel 4.16 menunjukkan matriks dari hasil prediksi. Berikut contoh dari perhitungannya.

$$\hat{Y}' = H' \hat{\beta}'$$

$$\hat{Y}'_{1,1} = (0,580249 * 0,129116) + (0,426981 * 0,646049)$$

$$\hat{Y}'_{1,1} = 0,350771$$

**Tabel 4.16 Matriks Bobot Keluaran**

$\hat{Y}'$	1
1	0,350771
2	0,400339
3	0,418879
4	0,385277
5	0,406606
6	0,388423
7	0,358998
8	0,387013
9	0,363681
10	0,303961

#### 4. Proses *testing*

**Langkah 1:** Mengetahui nilai  $W'_{mn}$ ,  $b$ , dan  $\hat{\beta}'$  dari proses *training*.

**Langkah 2:** Inisialisasi nilai matriks *delay*  $\delta'$  dengan melakukan perhitungan pada Persamaan 2.9. Tabel 4.17 menunjukkan hasil inisialisasi matriks *delay*  $\delta'$  dimana  $t$  adalah urutan data. Berikut contoh menghitung nilai *delay*  $\delta'$ .

$$\delta'_{tr} = T(t - (n + r) + n)$$

$$\delta'_{11,1} = T(11 - (3 + 1) + 3)$$

$$\delta'_{11,1} = T(10) = 0,200000$$

**Tabel 4.17 Matriks *Delay* Proses *Testing***

Data ke-t	r1	r2
11	0,200000	0,200000
12	0,197917	0,200000
13	0,322917	0,197917
14	0,900000	0,322917
15	0,791667	0,900000

**Langkah 3:** Menghitung nilai matriks keluaran pada *hidden layer* dengan menggunakan Persamaan 2.14.

**Langkah 3.1:** Menggabungkan matriks data uji yang hanya berisi fitur ( $x$ ) dengan matriks *delay*, dengan cara menambahkan kolom matriks data uji dengan matriks *delay*. Tabel 4.18 menunjukkan hasil penggabungan antara dua matriks tersebut.

**Tabel 4.18 Gabungan Matriks  $X$  Dengan Matriks *Delay* Proses *Testing***

No	x1	x2	x3	d1	d2
1	0,616667	0,200000	0,200000	0,200000	0,200000
2	0,200000	0,200000	0,197917	0,197917	0,200000
3	0,200000	0,197917	0,322917	0,322917	0,197917
4	0,197917	0,322917	0,900000	0,900000	0,322917
5	0,322917	0,900000	0,791667	0,791667	0,900000

**Langkah 3.2:** Memperbanyak matriks bias sebanyak jumlah data uji, dengan cara memperbanyak matriks bias sejumlah data uji dengan nilai yang sama. Tabel 4.19 menunjukkan hasil dari matriks bias yang sudah diperbanyak.

**Tabel 4.19 Matriks Bias Proses *Testing***

	1	2
1	0,488982	0,286403
2	0,488982	0,286403
3	0,488982	0,286403
4	0,488982	0,286403
5	0,488982	0,286403

**Langkah 3.3:** Menghitung perkalian matriks gabungan dengan *transpose* matriks bobot. Tabel 4.20 menunjukkan hasil dari perkalian matriks tersebut. Berikut contoh dari perhitungannya.

$$\begin{aligned}
 ([x, \delta] * W'^T)_{1,1} &= [x, \delta] * W'^T \\
 &= (0,616667 * -0,95691) + (0,200000 * 0,763482) \\
 &\quad + (0,200000 * 0,476625) \\
 &\quad + (0,200000 * 0,291437) \\
 &\quad + (0,200000 * 0,922641) \\
 &= -0,099258
 \end{aligned}$$

**Tabel 4.20 Matriks Hasil Perkalian**

	1	2
1	-0,099258	-0,654340
2	0,297855	-0,294607
3	0,390350	-0,280338
4	1,046345	-0,308299
5	1,816558	-0,837332

**Langkah 3.4:** Menghitung penjumlahan matriks hasil perkalian dengan matriks bias. Tabel 4.21 menunjukkan hasil dari penjumlahan matriks tersebut. Berikut contoh dari perhitungannya.

$$\begin{aligned}
 ([x, \delta] * W'^T + b(ones(i, 1), :))_{1,1} &= ([x, \delta] * W'^T + b(ones(i, 1), :)) \\
 &= -0,099258 + 0,488982 \\
 &= 0,389725
 \end{aligned}$$

**Tabel 4.21 Matriks Hasil Penjumlahan**

	1	2
1	0,389725	-0,367937
2	0,786837	-0,008204
3	0,879332	0,006065

**Tabel 4.21 Matriks Hasil Penjumlahan**

	1	2
4	1,535327	-0,021896
5	2,305540	-0,550929

**Langkah 3.5:** Menghitung fungsi aktivasi dari hasil penjumlahan untuk mendapatkan matriks keluaran pada *hidden layer*. Tabel 4.22 menunjukkan hasil matriks keluaran pada *hidden layer*. Berikut contoh dari perhitungannya.

$$H' = \frac{1}{1 + \exp(-([x, \delta]W''^T + b(\text{ones}(i, 1), :)))}$$

$$H'_{1,1} = \frac{1}{1 + \exp(-0,389725)}$$

$$H'_{1,1} = 0,596216$$

**Tabel 4.22 Matriks keluaran *hidden layer***

$H'$	1	2
1	0,596216	0,409040
2	0,687152	0,497949
3	0,706684	0,501516
4	0,822784	0,494526
5	0,909335	0,365649

**Langkah 4:** Menghitung hasil prediksi dengan menggunakan Persamaan 2.13. Tabel 4.23 menunjukkan matriks dari hasil prediksi. Berikut contoh dari perhitungannya.

$$\hat{Y}' = H' \hat{\beta}'$$

$$\hat{Y}'_{1,1} = (0,596216 * 0,129116) + (0,409040 * 0,646049)$$

$$\hat{Y}'_{1,1} = 0,341241$$

**Tabel 4.23 Matriks Bobot Keluaran**

$\hat{Y}'$	1
1	0,341241
2	0,410422
3	0,415248
4	0,425723
5	0,353637

**Langkah 5:** Denormalisasi hasil prediksi dengan menggunakan Persamaan 2.2. Nilai *max* dan *min* merupakan nilai yang didapatkan pada proses normalisasi. Tabel 4.24 menunjukkan matriks hasil denormalisasi. Berikut contoh dari perhitungannya.

$$\hat{Y}' = \frac{x' - (0,1)}{0,8} (max - min) + min$$

$$\hat{Y}'_{1,1} = \frac{0,341241 - (0,1)}{0,8} (13407 - 13311) + 13311$$

$$\hat{Y}'_{1,1} = 13339,948909$$

**Tabel 4.24 Matriks Denormalisasi**

$\hat{Y}'$	1
1	13339,948909
2	13348,250631
3	13348,829808
4	13350,086772
5	13341,436448

**Langkah 6:** Menghitung nilai evaluasi dengan menggunakan Persamaan 2.15. Berikut contoh dari perhitungannya.

$$MAPE = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} \frac{|t_i - y_i|}{t_i} \times 100\%$$

$$MAPE = \frac{1}{5} * \left( \left( \frac{|13322,75 - 13339,948909|}{13322,75} \right) + \left( \frac{|13337,75 - 13348,250631|}{13337,75} \right) + \dots + \left( \frac{|13363 - 13341,436448|}{13363} \right) \right) * 100\%$$

$$MAPE = 0,002262 * 100\%$$

$$MAPE = 0,2262 \%$$

## 4.5 Perancangan *User Interface*

Pada perancangan *user interface* akan rancang tampilan dari beberapa *form* yang ada pada program. *Form* tersebut adalah *form datasets*, *training*, dan *testing*.



#### 4.5.1 Perancangan *Form Datasets*

*Form datasets* adalah *form* yang menampilkan ruang kerja untuk melakukan proses inisialisasi parameter, pembentukan fitur, normalisasi data, dan pembagian antara data latih dan data uji. Gambar 4.21 menampilkan perancangan dari *form training*.

**Gambar 4.21 Perancangan *Form Datasets***

Penjelasan mengenai rancangan menu training sebagai berikut:

1. *Button* untuk melakukan load *datasets*.
2. Tabel untuk menampilkan *datasets*.
3. *TextBox* untuk memasukkan jumlah fitur.
4. *Button* untuk melakukan proses pembentukan fitur.
5. *Button* untuk melakukan proses normalisasi.
6. Tabel untuk menampilkan hasil pembentukan fitur.
7. Tabel untuk menampilkan hasil normalisasi.
8. *TextBox* untuk menampilkan jumlah data latih.
9. *TextBox* untuk memasukkan jumlah data uji.
10. *Button* untuk melakukan proses pembagian data.
11. Tabel untuk menampilkan data latih.
12. Tabel untuk menampilkan data uji.
13. *Button* untuk melanjutkan ke proses *training*.

#### 4.5.2 Perancangan *Form Training*

*Form training* adalah *form* yang menampilkan ruang kerja untuk melakukan proses *training*. Gambar 4.22 menampilkan perancangan dari *form training*.

The image shows a software interface titled "TRAINING". It contains the following elements:

- 1**: A text box for "Jumlah Hidden Neuron".
- 2**: A text box for "Jumlah Context Neuron".
- 3**: A button labeled "Random Bobot dan Bias".
- 4**: A button labeled "Proses Training".
- 5**: A large text area for "Log:".
- 6**: A table for "Bobot" (Weights).
- 7**: A table for "Bias".
- 8**: A table for "Delay".
- 9**: A button labeled "LANJUT KE PROSES TESTING".

**Gambar 4.22 Perancangan *Form Training***

Penjelasan mengenai rancangan menu training sebagai berikut:

1. *TextBox* untuk memasukkan jumlah *hidden neuron*.
2. *TextBox* untuk memasukkan jumlah *context neuron*.
3. *Button* untuk melakukan random bobot dan bias.
4. *Button* untuk melakukan proses *training*.
5. *TextArea* untuk menampilkan log program.
6. Tabel untuk menampilkan bobot.
7. Tabel untuk menampilkan bias.
8. Tabel untuk menampilkan *delay*.
9. *Button* untuk melanjutkan ke proses *testing*.

#### **4.5.3 Perancangan *Form Testing***

*Form testing* adalah *form* yang menampilkan ruang kerja untuk melakukan proses *testing*. Gambar 4.23 menampilkan perancangan dari *form testing*.

The diagram shows a rectangular form with a title bar at the top labeled 'TESTING'. Below the title bar, there are four numbered components: 1. A button labeled 'Proses Testing'. 2. A text box for displaying MAPE values. 3. A table for displaying prediction results. 4. A text area for displaying program logs.

**Gambar 4.23 Perancangan Form Testing**

Penjelasan mengenai rancangan menu training sebagai berikut:

1. *Button* untuk melakukan proses *testing*.
2. *TextBox* untuk menampilkan nilai MAPE.
3. Tabel untuk menampilkan hasil prediksi.
4. *TextArea* untuk menampilkan log program.

#### 4.6 Perancangan Uji Coba dan Evaluasi

Pengujian dan evaluasi pada penelitian ini digunakan untuk mengetahui kinerja dari metode RELMNN dalam memprediksi permasalahan nilai tukar Rupiah Indonesia terhadap dolar Amerika Serikat. Skenario pengujian yang dilakukan pada penelitian ini adalah:

1. Pengujian jumlah *hidden neuron*
2. Pengujian jumlah fitur
3. Pengujian jangka waktu prediksi
4. Pengujian jumlah *context neuron*
5. Pengujian Perbandingan metode RELMNN dan ELM

#### 4.6.1 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, jumlah tersebut didapatkan berdasarkan penelitian Ertugrul (2016) yang mendapatkan jumlah *hidden neuron* yang optimal sebesar 10 pada pengujian dengan range 1 sampai 30. 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. Tabel 4.25 merupakan tabel rancangan pengujian pengaruh *hidden neuron* terhadap nilai MAPE.

**Tabel 4.25 Rancangan Pengujian *Hidden Neuron***

Jumlah <i>Hidden Neuron</i>	Nilai MAPE (%) Percobaan Ke- <i>i</i>										Rata- rata MAPE (%)
	1	2	3	4	5	6	7	8	9	10	
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											
17											
18											
19											
20											

#### 4.6.2 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, jumlah tersebut didasarkan pada penelitian yang dilakukan oleh Sumarto et al. (2016), yang menggunakan 4 fitur dengan data teknikal, dan penelitian yg dilakukan oleh Pramesti dan Mahmudy (2016), yang menggunakan 4 fitur dengan data teknikal, dalam memprediksi nilai tukar uang. 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. Tabel 4.26 merupakan tabel rancangan pengujian pengaruh jumlah fitur terhadap nilai MAPE.

**Tabel 4.26 Rancangan Pengujian Jumlah Fitur**

Jumlah Fitur	Nilai MAPE (%) Percobaan Ke- <i>i</i>										Rata-rata MAPE (%)
	1	2	3	4	5	6	7	8	9	10	
3											
4											
5											
6											
7											

#### 4.6.3 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, jangka waktu tersebut didasarkan pada jangka waktu panjang, menengah dan pendek dalam memprediksi nilai tukar uang, sedangkan pembagian data latih dimulai dari data pertama kemudian dilanjutkan data uji. 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. Tabel 4.27 merupakan tabel rancangan pengujian pengaruh jangka waktu prediksi terhadap nilai MAPE.

**Tabel 4.27 Rancangan Pengujian Jangka Waktu Prediksi**

Jangka Waktu	Data Latih (hari)	Data Uji (hari)	Nilai MAPE (%) Percobaan Ke- <i>i</i>						Rata-rata MAPE (%)
			1	2	3	...	9	10	
1 hari	1890	1							
1 minggu	1890	7							
1 bulan	1890	30							
3 bulan	1890	91							
4 bulan	1890	121							
6 bulan	1890	182							
1 tahun	1890	365							
2 tahun	1890	730							
3 tahun	1890	1095							

#### 4.6.4 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, jumlah tersebut didapatkan berdasarkan penelitian Ertugrul (2016) yang mendapatkan jumlah *context neuron* yang optimal sebesar 3 pada range 1 sampai 365. 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. Tabel 4.28 merupakan tabel rancangan pengujian pengaruh *context neuron* terhadap nilai MAPE.

**Tabel 4.28 Rancangan Pengujian *Context Neuron***

Jumlah <i>Context Neuron</i>	Nilai MAPE (%) Percobaan Ke- <i>i</i>										Rata-rata MAPE (%)
	1	2	3	4	5	6	7	8	9	10	
1											
2											
3											
4											
5											
6											
7											

**Tabel 4.28 Rancangan Pengujian *Context Neuron***

Jumlah <i>Context Neuron</i>	Nilai MAPE (%) Percobaan Ke- <i>i</i>										Rata-rata MAPE (%)
	1	2	3	4	5	6	7	8	9	10	
8											
9											
10											
11											
12											
13											
14											
15											
16											
17											
18											
19											
20											

#### 4.6.5 Pengujian Perbandingan Metode RELMNN dengan Metode ELM

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. Tabel 4.29 merupakan tabel rancangan pengujian perbandingan metode RELMNN dan metode ELM dengan jenis *datasets* yang berbeda terhadap nilai MAPE. Tabel 4.30 merupakan tabel rancangan pengujian perbandingan metode RELMNN dengan metode ELM terhadap selisih nilai tukar Rupiah Indonesia terhadap Dolar Amerika dengan menggunakan jenis *datasets* yang optimal. Sedangkan, Tabel 4.31 merupakan tabel rancangan pengujian perbandingan metode RELMNN dengan metode ELM terhadap waktu yang dibutuhkan pada proses *training* dan *testing* menggunakan jenis *datasets* yang optimal.

**Tabel 4.29 Rancangan Pengujian Perbandingan Metode Terhadap Nilai MAPE**

Metode (Jenis <i>Datasets</i> )	Nilai MAPE (%) Percobaan Ke- <i>i</i>						Rata-rata MAPE (%)
	1	2	3	...	9	10	
RELMNN ( <i>datasets</i> terdapat <i>outliers</i> )							
ELM ( <i>datasets</i> terdapat <i>outliers</i> )							
RELMNN ( <i>datasets</i> tanpa <i>outliers</i> )							
ELM ( <i>datasets</i> tanpa <i>outliers</i> )							

**Tabel 4.30 Rancangan Pengujian Perbandingan Metode Terhadap Selisih Nilai Tukar Rupiah Indonesia terhadap Dolar Amerika**

Metode	Rata-rata Selisih Nilai Tukar antara Target dengan Prediksi (IDR) Percobaan Ke- <i>i</i>										Rata-rata Selisih (IDR)
	1	2	3	4	5	6	7	8	9	10	
RELMNN											
ELM											

**Tabel 4.31 Rancangan Pengujian Perbandingan Metode Terhadap Waktu yang Digunakan**

Metode (Proses)	Waktu yang digunakan (detik) Percobaan Ke- <i>i</i>										Rata-rata waktu (detik)
	1	2	3	4	5	6	7	8	9	10	
RELMNN (Training)											
ELM (Training)											
RELMNN (Testing)											
ELM (Testing)											



## BAB 5 IMPLEMENTASI

Bab ini membahas tentang implementasi dan antarmuka dari perancangan yang telah dibuat untuk prediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat menggunakan metode *Recurrent Extreme Learning Machine Neural Network*.

### 5.1 Implementasi Program

Berdasarkan perancangan yang telah dijelaskan pada Bab 4, maka pada bab ini akan membahas implementasi program, sesuai dengan perancangan yang telah dibuat. Program di implementasikan dengan menggunakan bahasa Java Graphic User Interface (Java GUI).

#### 5.1.1 Implementasi Proses Normalisasi Data

Proses ini diawali dengan mendapatkan nilai max dan min pada keseluruhan *datasets*, kemudian nilai tersebut digunakan untuk melakukan normalisasi seluruh *datasets*. Kode Program pada proses ini dapat dilihat pada Kode Program 5.1.

```
1 public void normalisasi() {
2     max = 0;
3     min = 1000000;
4
5     for (int j = 0; j < datasets.length; j++) {
6         if (datasets[j] > max) {
7             max = datasets[j];
8         }
9         if (datasets[j] < min) {
10             min = datasets[j];
11         }
12     }
13
14     for (int i = 0; i < data[0].length; i++) {
15         for (int j = 0; j < data.length; j++) {
16             data[j][i] = (data[j][i] - min) / (max - min) *
17                 0.8 + 0.1;
18         }
19     }
20
21     String header[] = new String[data[0].length];
22     for (int i = 0; i < data[0].length - 1; i++) {
23         header[i] = "x" + (i + 1);
24     }
25     header[data[0].length - 1] = "T";
26     DefaultTableModel model = new DefaultTableModel(null,
27         header);
28     model.setColumnCount(data[0].length);
29     model.setRowCount(data.length);
30
31     for (int row = 0; row < data.length; row++) {
32         for (int column = 0; column < data[0].length;
33             column++) {
34             model.setValueAt(data[row][column], row,
```

35	column);
36	}
37	}
38	t_normalisasi.setModel(model);
39	}

### Kode Program 5.1 Proses Normalisasi Data

Penjelasan dari Kode Program 5.1 adalah sebagai berikut:

1. Baris 2-3 merupakan proses inisialisasi nilai max dan min
2. Baris 5-12 merupakan proses menentukan nilai max dan min pada keseluruhan *datasets*
3. Baris 14-19 merupakan proses normalisasi pada setiap data yang ada pada baris ke-*j* kolom ke-*i*.
4. Baris 21-25 merupakan proses membuat model header pada tabel hasil normalisasi.
5. Baris 26-29 merupakan proses membuat model tabel hasil normalisasi.
6. Baris 31-37 merupakan proses menambahkan nilai hasil normalisasi pada tabel normalisasi.
7. Baris 38 merupakan proses pengaturan model pada tabel hasil normalisasi.

### 5.1.2 Implementasi Proses Inisialisasi Matriks *Delay*

Proses ini dilakukan dengan menambahkan nilai target pada indeks yang didapatkan ke dalam matriks *delay*. Kode Program pada proses ini dapat dilihat pada Kode Program 5.2.

1	public double[][] inisialisasiMatriksDelay(double
2	datalatih[][]) {
3	int t = datalatih.length;
4	int n = datalatih[0].length - 1;
5	double delay[][] = new double[t][context];
6	
7	for (int i = 0; i < t; i++) {
8	for (int j = 0; j < context; j++) {
9	if (i > j) {
10	delay[i][j] = datalatih[((i-j) -
11	1)][datalatih[0].length - 1];
12	} else {
13	delay[i][j] = 0;
14	}
15	}
16	}
17	
18	return delay;
19	}

### Kode Program 5.2 Proses Inisialisasi Matriks *Delay*

Penjelasan dari Kode Program 5.2 adalah sebagai berikut:

1. Baris 3-5 merupakan proses inisialisasi matriks *delay*.
2. Baris 7-16 merupakan proses memasukkan nilai target pada index yang didapatkan pada matriks *delay*. Jika nilai target pada index tidak ada maka diisi dengan nilai 0.

### 5.1.3 Implementasi Proses Membuat Nilai Random Bobot dan Bias

Proses ini diawali dengan pembentukan matriks untuk menampung bobot masukan dan bias, kemudian membuat nilai random dengan range [-1,1] untuk bobot masukan dan range [0,1] untuk bias. Kode Program pada proses ini dapat dilihat pada Kode Program 5.3.

```
1 public void membuatNilaiRandomBobotBias() {
2     this.hidden = Integer.parseInt(jhidden.getText());
3     this.context = Integer.parseInt(jcontext.getText());
4     W = new double[hidden][fitur + context];
5     b = new double[1][hidden];
6     for (int i = 0; i < hidden; i++) {
7         for (int j = 0; j < (fitur + context); j++) {
8             W[i][j] = Math.random() * 2 - 1;
9         }
10        b[0][i] = Math.random();
11    }
12    //tampilkan hasil ke tabel
13    String header[] = new String[W[0].length + 1];
14    for (int i = 1; i <= W[0].length; i++) {
15        header[i] = i + "";
16    }
17    header[0] = "M\\N+R";
18    DefaultTableModel model = new DefaultTableModel(null,
19    header);
20    model.setColumnCount(W[0].length + 1);
21    model.setRowCount(W.length);
22    for (int row = 0; row < W.length; row++) {
23        for (int column = 1; column < W[0].length + 1;
24        column++) {
25            model.setValueAt(W[row][column - 1], row,
26            column);
27        }
28        model.setValueAt(row + 1, row, 0);
29    }
30    t_bobot.setModel(model);
31
32    String header2[] = new String[b[0].length];
33    for (int i = 0; i < b[0].length; i++) {
34        header2[i] = "b" + (i + 1);
35    }
36    DefaultTableModel model2 = new DefaultTableModel(null,
37    header2);
38    model2.setColumnCount(b[0].length);
39    model2.setRowCount(1);
40 }
```

41	for (int column = 0; column < b[0].length; column++) {
42	model2.setValueAt(b[0][column], 0, column);
43	}
44	t_bias.setModel(model2);
45	}

#### Kode Program 5.3 Proses Membuat Nilai Random Bobot dan Bias

Penjelasan dari Kode Program 5.3 adalah sebagai berikut:

1. Baris 2-5 merupakan proses inisialisasi matriks  $W$  sebagai bobot dan  $b$  sebagai bias.
2. Baris 6-11 merupakan proses membuat nilai random untuk bobot dan bias.
3. Baris 13-17 merupakan proses membuat model header tabel bobot.
4. Baris 18-21 merupakan proses membuat model tabel bobot.
5. Baris 22-30 merupakan proses memasukkan nilai bobot ke tabel bobot.
6. Baris 32-35 merupakan proses membuat model header tabel bias.
7. Baris 36-39 merupakan proses membuat model tabel bias.
8. Baris 41-43 merupakan proses memasukkan nilai bias ke tabel bias.

#### 5.1.4 Implementasi Proses Menghitung Keluaran pada *Hidden Layer*

Proses ini diawali dengan pembentukan matriks  $X$  untuk menampung nilai fitur saja, kemudian menjalankan proses gabung matriks, proses perbanyak matriks bias, proses *transpose*, proses perkalian matriks, penjumlahan matriks dan aktivasi. Kode Program pada proses ini dapat dilihat pada Kode Program 5.4.

1	public double[][]
2	menghitungMatriksKeluaranPadaHiddenLayer(double[][] data,
3	double[][] delay, double[][] b, double[][] W) {
4	double[][] X = new double[data.length][data[0].length-
5	1];
6	for (int i = 0; i < X.length; i++) {
7	for (int j = 0; j < X[0].length; j++) {
8	X[i][j] = data[i][j];
9	}
10	}
11	double[][] gabungan = gabungMatriks(X, delay);
12	double[][] bias = perbanyakMatriks(b, X.length);
13	double[][] Wt = transpose(W);
14	double[][] perkalian = perkalianMatriks(gabungan, Wt);
15	double[][] penjumlahan = penjumlahanMatriks(perkalian,
16	bias);
17	double[][] aktivasi = aktivasi(penjumlahan);
18	
19	return aktivasi;
20	}

#### Kode Program 5.4 Proses Menghitung Keluaran pada *Hidden Layer*

Penjelasan dari Kode Program 5.4 adalah sebagai berikut:

1. Baris 4-10 merupakan proses pembentukan matriks *X* untuk menampung fitur yang terdapat pada data.
2. Baris 11 menjalankan proses gabung matriks *X* dengan matriks *delay*.
3. Baris 12 menjalankan proses memperbanyak matriks bias sebanyak panjang baris matriks *X*.
4. Baris 13 menjalankan proses *transpose* matriks *W*.
5. Baris 14 menjalankan proses perkalian matriks gabungan dengan *Wt*.
6. Baris 15-16 menjalankan proses penjumlahan matriks perkalian dengan bias.
7. Baris 17 menjalankan proses aktivasi pada matriks penjumlahan.

### 5.1.5 Implementasi Proses Menggabungkan Matriks Input dengan Matriks *Delay*

Proses ini diawali dengan pembentukan matriks gabungan, kemudian memasukkan nilai pada matriks *A* dan *B* ke dalam matriks gabungan. Kode Program pada proses ini dapat dilihat pada Kode Program 5.5.

1	<code>public double[][] gabungMatriks(double A[][], double B[][])</code>
2	<code>{</code>
3	<code>    double[][] gabungan = new double[A.length][A[0].length</code>
4	<code>    + B[0].length];</code>
5	<code>    for (int i = 0; i &lt; gabungan.length; i++) {</code>
6	<code>        for (int j = 0; j &lt; A[0].length; j++) {</code>
7	<code>            gabungan[i][j] = A[i][j];</code>
8	<code>        }</code>
9	<code>        for (int k = A[0].length; k &lt; gabungan[0].length;</code>
10	<code>        k++) {</code>
11	<code>            gabungan[i][k] = B[i][k - A[0].length];</code>
12	<code>        }</code>
13	<code>    }</code>
14	<code>    return gabungan;</code>
15	<code>}</code>

**Kode Program 5.5 Proses Menggabungkan Matriks Input dengan Matriks *Delay***

Penjelasan dari Kode Program 5.5 adalah sebagai berikut:

1. Baris 3-4 merupakan proses pembentukan matriks gabungan.
2. Baris 6-8 merupakan proses memasukkan nilai yang ada pada matriks *A* ke dalam matriks gabungan.
3. Baris 9-12 merupakan proses memasukkan nilai yang ada pada matriks *B* ke dalam matriks gabungan.

### 5.1.6 Implementasi Proses Memperbanyak Matriks Bias

Proses ini diawali dengan pembentukan matriks bias, kemudian memperbanyak matriks *b* sebanyak nilai jumlah. Kode Program pada proses ini dapat dilihat pada Kode Program 5.6.

1	public double[][] perbanyakMatriks(double b[][], int
2	jumlah) {
3	double[][] bias = new double[jumlah][b[0].length];
4	for (int i = 0; i < jumlah; i++) {
5	for (int j = 0; j < b[0].length; j++) {
6	bias[i][j] = b[0][j];
7	}
8	}
9	return bias;
10	}

**Kode Program 5.6 Proses Memperbanyak Matriks Bias**

Penjelasan dari Kode Program 5.6 adalah sebagai berikut:

1. Baris 3 merupakan proses pembentukan matriks bias.
2. Baris 4-8 merupakan proses memperbanyak matriks bias sebanyak nilai jumlah.

### 5.1.7 Implementasi Proses *Transpose* Matriks

Proses ini diawali dengan pembentukan matriks *transpose*, kemudian melakukan proses *transpose* dengan menempatkan baris matriks A sebagai kolom matriks *transpose*. Kode Program pada proses ini dapat dilihat pada Kode Program 5.7.

1	public double[][] transpose(double A[][]) {
2	double[][] transpose = new
3	double[A[0].length][A.length];
4	for (int i = 0; i < transpose.length; i++) {
5	for (int j = 0; j < transpose[0].length; j++) {
6	transpose[i][j] = A[j][i];
7	}
8	}
9	return transpose;
10	}

**Kode Program 5.7 Proses Tranpose Matriks**

Penjelasan dari Kode Program 5.7 adalah sebagai berikut:

1. Baris 2-3 merupakan proses pembentukan matriks *transpose*.
2. Baris 4-8 merupakan proses *tranpose* matriks dengan memindahkan nilai yang sebelumnya berada di baris menjadi di kolom dan sebaliknya.

### 5.1.8 Implementasi Proses Perkalian Matriks

Proses ini diawali dengan pembentukan matriks perkalian, kemudian melakukan proses perkalian matriks. Kode Program pada proses ini dapat dilihat pada Kode Program 5.8.

1	public double[][] perkalianMatriks(double A[][], double
2	B[][]) {
3	double[][] perkalian = new
4	double[A.length][B[0].length];
5	for (int i = 0; i < A.length; i++) {
6	for (int j = 0; j < B[0].length; j++) {

7	for (int k = 0; k < A[0].length; k++) {
8	perkalian[i][j] += A[i][k] * B[k][j];
9	}
10	}
11	}
12	return perkalian;
13	}

**Kode Program 5.8 Proses Perkalian Matriks**

Penjelasan dari Kode Program 5.8 adalah sebagai berikut:

1. Baris 3-4 merupakan proses pembentukan matriks perkalian.
2. Baris 5-11 merupakan proses perkalian matriks.

### 5.1.9 Implementasi Proses Penjumlahan Matriks

Proses ini diawali dengan pembentukan matriks penjumlahan, kemudian melakukan proses penjumlahan matriks. Kode Program pada proses ini dapat dilihat pada Kode Program 5.9.

1	public double[][] penjumlahanMatriks(double A[][], double
2	B[][]) {
3	double[][] penjumlahan = new
4	double[A.length][A[0].length];
5	for (int i = 0; i < A.length; i++) {
6	for (int j = 0; j < A[0].length; j++) {
7	penjumlahan[i][j] = A[i][j] + B[i][j];
8	}
9	}
10	return penjumlahan;
11	}

**Kode Program 5.9 Proses Penjumlahan Matriks**

Penjelasan dari Kode Program 5.9 adalah sebagai berikut:

1. Baris 3-4 merupakan proses pembentukan matriks penjumlahan.
2. Baris 5-9 merupakan proses penjumlahan matriks.

### 5.1.10 Implementasi Proses Menghitung Fungsi Aktivasi

Proses ini diawali dengan pembentukan matriks aktivasi, kemudian melakukan proses menghitung fungsi aktivasi. Kode Program pada proses ini dapat dilihat pada Kode Program 5.10.

1	public double[][] aktivasi(double A[][]) {
2	double[][] aktivasi = new
3	double[A.length][A[0].length];
4	for (int i = 0; i < aktivasi.length; i++) {
5	for (int j = 0; j < aktivasi[0].length; j++) {
6	aktivasi[i][j] = 1 / (1 + Math.exp(-A[i][j]));
7	}
8	} return aktivasi;
9	}

**Kode Program 5.10 Proses Menghitung Fungsi Aktivasi**

Penjelasan dari Kode Program 5.10 adalah sebagai berikut:

1. Baris 2-3 merupakan proses pembentukan matriks aktivasi.
2. Baris 4-8 merupakan proses menghitung fungsi aktivasi pada setiap nilai matriks *A*.

#### 5.1.11 Implementasi Proses Menghitung Bobot Keluaran

Proses ini diawali dengan melakukan perhitungan matriks *Moore-Penrose Pseudo Invers*, kemudian melakukan perkalian matriks tersebut dengan target. Kode Program pada proses ini dapat dilihat pada Kode Program 5.11.

1	public double[][] menghitungBobotKeluaran(double[][] H,
2	double[][] datalatih) {
3	double[][] Hplus =
4	menghitungMoorePenrosePseudoInvers(H);
5	double[][] T = new double[datalatih.length][1];
6	for (int i = 0; i < datalatih.length; i++) {
7	T[i][0] = datalatih[i][datalatih[0].length - 1];
8	}
9	double[][] B = perkalianMatriks(Hplus, T);
10	return B;
11	}

**Kode Program 5.11 Proses Menghitung Bobot Keluaran**

Penjelasan dari Kode Program 5.11 adalah sebagai berikut:

1. Baris 3-4 proses menghitung matriks *Moore-Penrose Pseudo Invers*.
2. Baris 5-8 proses pembentukan matriks target.
3. Baris 9 proses perkalian matriks keluaran pada *hidden layer* dengan target.

#### 5.1.12 Implementasi Proses Menghitung Matriks *Moore-Penrose Pseudo Invers*

Proses ini terdiri dari menjalankan proses *transpose*, perkalian, dan *invers*. Kode Program pada proses ini dapat dilihat pada Kode Program 5.12.

1	public double[][]
2	menghitungMoorePenrosePseudoInvers(double[][] H) {
3	double[][] Ht = transpose(H);
4	double[][] HtH = perkalianMatriks(Ht, H);
5	double[][] HtHinv = invers(HtH);
6	double[][] Hplus = perkalianMatriks(HtHinv, Ht);
7	return Hplus;
8	}

**Kode Program 5.12 Proses Menghitung Matriks *Moore-Penrose Pseudo Invers***

Penjelasan dari Kode Program 5.12 adalah sebagai berikut:

1. Baris 3 menjalankan proses *transpose* pada matriks *H*.
2. Baris 4 menjalankan proses perkalianMatriks antara matriks *Ht* (*H transpose*) dengan *H*.



3. Baris 5 menjalankan proses *invers* pada matriks  $HtH$  ( $H$  transpose x  $H$ ).
4. Baris 6 menjalankan proses perkalian Matriks antara matriks  $HtHinv$  (*invers* dari  $H$  transpose x  $H$ ) dengan  $Ht$ .

### 5.1.13 Implementasi Proses Menghitung *Invers* Matriks

Proses ini diawali dengan membuat matriks identitas dan menggabungkan dengan matriks  $A$ , kemudian menghitung *invers* pada matriks  $A$  dengan OBE. Kode Program pada proses ini dapat dilihat pada Kode Program 5.13.

```

1 public double[][] invers(double[][] A) {
2     int p = A.length;
3     int pp = p * 2;
4     double[][] invers = new double[A.length][A[0].length];
5     double[][] matriks = new double[p][pp];
6     for (int i = 0; i < p; i++) {
7         for (int j = p; j < pp; j++) {
8             if (i + p == j) {
9                 matriks[i][j] = 1;
10            } else {
11                matriks[i][j] = 0;
12            }
13            matriks[i][j - p] = A[i][j - p];
14        }
15    }
16    for (int j = 0; j < p - 1; j++) {
17        for (int i = j + 1; i < p; i++) {
18            if (matriks[i][j] != 0) {
19                double s = matriks[j][j] / matriks[i][j];
20                for (int k = j; k < pp; k++) {
21                    matriks[i][k] *= s;
22                    matriks[i][k] -= matriks[j][k];
23                }
24            }
25        }
26    }
27    for (int j = p - 1; j > 0; j--) {
28        for (int i = j - 1; i >= 0; i--) {
29            if (matriks[i][j] != 0) {
30                double s = matriks[j][j] / matriks[i][j];
31                for (int k = i; k < pp; k++) {
32                    matriks[i][k] *= s;
33                    matriks[i][k] -= matriks[j][k];
34                }
35            }
36        }
37    }
38    for (int j = 0; j < p; j++) {
39        if (matriks[j][j] != 1) {
40            double s = 1 / matriks[j][j];
41            for (int k = j; k < pp; k++) {
42                matriks[j][k] *= s;
43            }
44        }
45    }
46    for (int i = 0; i < p; i++) {

```

47	for (int j = p; j < pp; j++) {
48	invers[i][j - p] = matriks[i][j];
49	}
50	}
51	return invers;
52	}

#### Kode Program 5.13 Proses Menghitung *Invers* Matriks

Penjelasan dari Kode Program 5.13 adalah sebagai berikut:

1. Baris 2 merupakan proses inisialisasi  $p$  dengan panjang baris matriks  $A$ .
2. Baris 3 merupakan proses inisialisasi nilai  $pp$  dengan 2 kali nilai  $p$ .
3. Baris 4 merupakan proses inisialisasi matriks *invers* untuk menampung hasil *invers* matriks.
4. Baris 5 merupakan proses inisialisasi matriks untuk menampung matriks  $A$  dan matriks identitas.
5. Baris 6-15 merupakan proses pembuatan identitas dan penggabungan dengan matriks  $A$ .
6. Baris 16-26 merupakan proses perhitungan OBE langkah pertama untuk menghitung *invers* matriks.
7. Baris 27-37 merupakan proses perhitungan OBE langkah kedua untuk menghitung *invers* matriks.
8. Baris 38-45 merupakan proses perhitungan OBE langkah ketiga untuk menghitung *invers* matriks.
9. Baris 46-50 merupakan proses memasukkan hasil *invers* matriks ke matriks *invers*.

#### 5.1.14 Implementasi Proses Menghitung Hasil Prediksi

Proses ini terdiri dari menjalankan proses perkalian antara matriks keluaran pada *hidden layer* dengan matriks bobot keluaran. Kode Program pada proses ini dapat dilihat pada Kode Program 5.14.

1	public double[][] menghitungHasilPrediksi(double[][] H,
2	double[][] B){
3	return perkalianMatriks(H, B);
4	}

#### Kode Program 5.14 Proses Menghitung Hasil Prediksi

Penjelasan dari Kode Program 5.14 adalah sebagai berikut:

1. Baris 3 menjalankan proses perkalian antara matriks  $H$  dengan matriks  $B$ .

#### 5.1.15 Implementasi Proses Denormalisasi

Proses ini diawali dengan membentuk matriks denormalisasi, kemudian menghitung nilai denormalisasi dengan nilai *Max* dan *Min* pada data kolom target

yang didapatkan pada saat proses normalisasi. Kode Program pada proses ini dapat dilihat pada Kode Program 5.15.

1	public double[][] denormalisasi(double[][] Y) {
2	double[][] denormalisasi = new
3	double[Y.length][Y[0].length];
4	for (int i = 0; i < denormalisasi.length; i++) {
5	denormalisasi[i][0] = (Y[i][0] - 0.1) / 0.8 * (max
6	- min) + min;
7	}
8	return denormalisasi;
9	}

**Kode Program 5.15 Proses Menghitung Hasil Prediksi**

Penjelasan dari Kode Program 5.15 adalah sebagai berikut:

1. Baris 2-3 melakukan inisialisasi matriks denormalisasi untuk mendapatkan hasil prediksi berupa nilai tukar.
2. Baris 4-7 merupakan proses perhitungan nilai denormalisasi pada setiap nilai matriks Y.

### 5.1.16 Implementasi Proses Menghitung Nilai Evaluasi

Proses ini merupakan proses menghitung nilai evaluasi dengan menggunakan MAPE. Kode Program pada proses ini dapat dilihat pada Kode Program 5.16.

1	public double hitungMAPE(JTable hasil){
2	double total = 0;
3	int n = hasil.getRowCount();
4	for (int i = 0; i < n; i++) {
5	double target = Double.parseDouble(
6	String.valueOf(hasil.getValueAt(i,1)).replace(",","
7	".");
8	double prediksi = Double.parseDouble(
9	String.valueOf(hasil.getValueAt(i,2)).replace(",","
10	".");
11	total += Math.abs(target-prediksi)/target;
12	}
13	double mape = total/n * 100;
14	return mape;
15	}

**Kode Program 5.16 Proses Menghitung Nilai Evaluasi**

Penjelasan dari Kode Program 5.16 adalah sebagai berikut:

1. Baris 2-3 melakukan inisialisasi nilai *total* dan *n*.
2. Baris 4-13 merupakan proses perhitungan nilai MAPE.

## 5.2 Implementasi *User Interface*

Sub bab ini merupakan pembahasan mengenai hasil implementasi *user interface* yang telah dirancang pada Bab 4.

### 5.2.1 Implementasi Form Datasets

*Form Datasets* merupakan *form* yang digunakan untuk melakukan *input* file datasets dengan ekstensi .xls, jumlah fitur, dan jumlah data uji. *Form* ini juga digunakan untuk membentuk fitur, normalisasi data, dan pembagian data latih dan data uji. Implementasi *form datasets* dapat dilihat pada Gambar 5.1.

135150200111002 - Daneswara Jauhari

**DATASETS** **PEMBENTUKAN FITUR & NORMALISASI** **PEMBAGIAN DATA LATIH & DATA UJI**

Datasets Load Datasets Jumlah Fitur 4 Proses Normalisasi Jumlah Data Latih 1890 Jumlah Data Uji 30 hari Proses

No Tanggal Nilai Tukar

1 04 Januari 11196

2 05 Januari 11040

3 06 Januari 11021,5

4 07 Januari 11030,5

5 08 Januari 10946

6 09 Januari 11075,5

7 10 Januari 11075,5

8 11 Januari 11075,5

9 12 Januari 11035

10 13 Januari 11116,5

11 14 Januari 11306

12 15 Januari 11165

13 16 Januari 11177

14 17 Januari 11177

15 18 Januari 11177

16 19 Januari 11001,5

17 20 Januari 11114,5

18 21 Januari 11335,5

19 22 Januari 11337,5

20 23 Januari 11203,5

21 24 Januari 11203,5

22 25 Januari 11203,5

23 26 Januari 11126,78

24 27 Januari 11078

25 28 Januari 11121

26 29 Januari 11347

27 30 Januari 11430

28 31 Januari 11430

29 01 Februari 11430

30 02 Februari 11423,5

31 03 Februari 11393

Hasil Pembentukan Fitur

x1 x2 x3 x4 T

11196 0 11040 0 11021,5 11030,5 10946 0

11040 0 11021,5 11030,5 10946 0 11075,5

11021,5 11030,5 10946 0 11075,5 11075,5

11030,5 10946 0 11075,5 11075,5 11075,5

10946 0 11075,5 11075,5 11075,5 11035 0

11075,5 11075,5 11035 0 11116,5 11165 0

11075,5 11035 0 11116,5 11306 11165 0

11035 0 11116,5 11306 11165 0 11177 0

Hasil Normalisasi

x1 x2 x3 x4 T

0,6851410... 0,6520858... 0,6481658... 0,6500728... 0,6321679...

0,6520858... 0,6481658... 0,6500728... 0,6321679... 0,6596079...

0,6481658... 0,6500728... 0,6321679... 0,6596079... 0,6596079...

0,6500728... 0,6321679... 0,6596079... 0,6596079... 0,6596079...

0,6321679... 0,6596079... 0,6596079... 0,6596079... 0,6510263...

0,6596079... 0,6596079... 0,6596079... 0,6510263... 0,6682955...

0,6596079... 0,6510263... 0,6682955... 0,7084492... 0,7084492...

0,6510263... 0,6682955... 0,7084492... 0,7857223... 0,6811150...

0,6682955... 0,7084492... 0,7857223... 0,6811150... 0,6811150...

0,7084492... 0,7857223... 0,6811150... 0,6811150... 0,6811150...

0,7857223... 0,6811150... 0,6811150... 0,6811150... 0,6439279...

0,6811150... 0,6811150... 0,6811150... 0,6439279... 0,6678718...

0,6811150... 0,6811150... 0,6439279... 0,6678718... 0,7147000...

0,6811150... 0,6439279... 0,6678718... 0,7147000... 0,7151238...

Data Latih

x1 x2 x3 x4 T

0,6851410409... 0,6520858164... 0,6481658058... 0,6500728380... 0,6321679247...

0,6520858164... 0,6481658058... 0,6500728380... 0,6321679247... 0,6596079989...

0,6481658058... 0,6500728380... 0,6321679247... 0,6596079989... 0,6596079989...

0,6500728380... 0,6321679247... 0,6596079989... 0,6596079989... 0,6596079989...

0,6321679247... 0,6596079989... 0,6596079989... 0,6596079989... 0,6510263541...

0,6596079989... 0,6596079989... 0,6596079989... 0,6510263541... 0,6682955899...

0,6596079989... 0,6596079989... 0,6510263541... 0,6682955899... 0,7084492120...

0,6596079989... 0,6510263541... 0,6682955899... 0,7084492120... 0,6785723745...

0,6510263541... 0,6682955899... 0,7084492120... 0,6785723745... 0,6811150840...

Data Uji

x1 x2 x3 x4 T

0,7274135876... 0,7274135876... 0,7417163289... 0,7448947159... 0,7406586866...

0,7274135876... 0,7417163289... 0,7448947159... 0,7406586866... 0,7358892861...

0,7417163289... 0,7448947159... 0,7406586866... 0,7358892861... 0,7083432657...

0,7448947159... 0,7406586866... 0,7358892861... 0,7083432657... 0,7083432657...

0,7406586866... 0,7358892861... 0,7083432657... 0,7083432657... 0,7083432657...

0,7358892861... 0,7083432657... 0,7083432657... 0,7083432657... 0,6981724274...

0,7083432657... 0,7083432657... 0,7083432657... 0,6981724274... 0,7072838034...

0,7083432657... 0,6981724274... 0,6981724274... 0,7072838034... 0,7235995232...

0,7083432657... 0,6981724274... 0,7072838034... 0,7235995232... 0,7421401138...

LANJUT KE PROSES TRAINING

Dosen Pembimbing 1 : Imam Cholissodin, S.Si, M.Kom

Dosen Pembimbing 2 : Candra Dewi, S.Kom, M.Sc

135150200111002 - Daneswara Jauhari

**DATASETS** **PEMBENTUKAN FITUR & NORMALISASI** **PEMBAGIAN DATA LATIH & DATA UJI**

Datasets Load Datasets Jumlah Fitur 3 Proses Normalisasi Jumlah Data Latih 1890 Jumlah Data Uji 365 hari Proses

No Tanggal Nilai Tukar

1 05 Januari 11040

2 06 Januari 11021,5

3 07 Januari 11030,5

4 08 Januari 10946

5 09 Januari 11075,5

6 10 Januari 11075,5

7 11 Januari 11075,5

8 12 Januari 11035

9 13 Januari 11116,5

10 14 Januari 11306

11 15 Januari 11165

12 16 Januari 11177

13 17 Januari 11177

14 18 Januari 11177

15 19 Januari 11001,5

16 20 Januari 11114,5

17 21 Januari 11335,5

18 22 Januari 11337,5

19 23 Januari 11203,5

20 24 Januari 11203,5

21 25 Januari 11203,5

22 26 Januari 11126,78

23 27 Januari 11078

24 28 Januari 11121

25 29 Januari 11347

26 30 Januari 11430

27 31 Januari 11430

28 01 Februari 11430

29 02 Februari 11423,5

30 03 Februari 11393

31 04 Februari 11682

Hasil Pembentukan Fitur

x1 x2 x3 T

11040 0 11021,5 11030,5 10946 0

11021,5 11030,5 10946 0 11075,5

11030,5 10946 0 11075,5 11075,5

10946 0 11075,5 11075,5 11075,5

11075,5 11075,5 11035 0

11075,5 11075,5 11035 0 11116,5

11075,5 11035 0 11116,5 11306 0

11035 0 11116,5 11306 11165 0

11116,5 11306 0 11165 0 11177 0

Hasil Normalisasi

x1 x2 x3 T

0,5362494767... 0,5331519464... 0,5346588530... 0,5205106739...

0,5331519464... 0,5346588530... 0,5205106739... 0,5421933863...

0,5346588530... 0,5205106739... 0,5421933863... 0,5421933863...

0,5205106739... 0,5421933863... 0,5421933863... 0,5421933863...

0,5421933863... 0,5421933863... 0,5421933863... 0,5354123064...

0,5421933863... 0,5421933863... 0,5354123064... 0,5490581833...

0,5421933863... 0,5354123064... 0,5490581833... 0,5807869401...

0,5354123064... 0,5490581833... 0,5807869401... 0,5571787358...

0,5490581833... 0,5807869401... 0,5571787358... 0,5591879447...

0,5807869401... 0,5571787358... 0,5591879447... 0,5591879447...

0,5571787358... 0,5591879447... 0,5591879447... 0,5591879447...

0,5591879447... 0,5591879447... 0,5591879447... 0,5298032649...

0,5591879447... 0,5298032649... 0,5487233151... 0,5857262452...

0,5298032649... 0,5487233151... 0,5857262452... 0,5806111334...

Data Latih

x1 x2 x3 T

0,53624947676852... 0,53315194642109... 0,53465885307660... 0,52051067392214...

0,53315194642109... 0,53465885307660... 0,52051067392214... 0,54219338635412...

0,53465885307660... 0,52051067392214... 0,54219338635412... 0,54219338635412...

0,52051067392214... 0,54219338635412... 0,54219338635412... 0,54219338635412...

0,54219338635412... 0,54219338635412... 0,54219338635412... 0,53541230640435...

0,54219338635412... 0,54219338635412... 0,53541230640435... 0,54905818334030...

0,54219338635412... 0,53541230640435... 0,54905818334030... 0,580786940142319...

0,53541230640435... 0,54905818334030... 0,580786940142319... 0,55717873587275...

0,54905818334030... 0,580786940142319... 0,55717873587275... 0,55918794474675...

Data Uji

x1 x2 x3 T

0,595772289660948... 0,607074089577229... 0,60958560066973... 0,60623691921305...

0,607074089577229... 0,60958560066973... 0,60623691921305... 0,60246965257429...

0,60958560066973... 0,60623691921305... 0,60246965257429... 0,580730223105902...

0,60623691921305... 0,60246965257429... 0,580730223105902... 0,580730223105902...

0,60246965257429... 0,580730223105902... 0,580730223105902... 0,57266538760987...

0,580730223105902... 0,580730223105902... 0,57266538760987... 0,579866052741733...

0,580730223105902... 0,57266538760987... 0,579866052741733... 0,59275847634993...

0,57266538760987... 0,579866052741733... 0,59275847634993... 0,60740895772289...

LANJUT KE PROSES TRAINING

Dosen Pembimbing 1 : Imam Cholissodin, S.Si, M.Kom

Dosen Pembimbing 2 : Candra Dewi, S.Kom, M.Sc

Gambar 5.1 Implementasi Form Datasets

### 5.2.2 Implementasi Form Training

*Form Training* merupakan *form* yang digunakan untuk melakukan *training* pada data latih. Untuk menjalankan *training* pada *form* ini harus menginputkan jumlah *hidden neuron*, *context neuron* dan menekan button Random Bobot dan Bias. Implementasi *form training* dapat dilihat pada Gambar 5.2.

135150200111002 - Daneswara Jauhari

### TRAINING

Jumlah Hidden Neuron

Jumlah Context Neuron

Random Bobot dan Bias

Proses Training

Log:

```

0.7701801135000910
0.7703046890739582
0.7413638965269023
0.7397634345888093
0.7307407577343517
0.7294698176899983
0.7378469175848643

```

**Bobot**

MN+R	1	2	3	4	5
1	-0.46501661...	-0.32358341...	0.781467665...	0.001837216...	-0.47065641...
2	0.066098961...	-0.42728751...	-0.03303385...	-0.00516044...	-0.39622193...
3	-0.05346153...	-0.11804810...	0.178470527...	0.916055127...	0.329946549...
4	-0.80451393...	-0.99766774...	-0.56398218...	-0.55355382...	-0.02783881...

**Bias**

b1	b2	b3	b4	b5	b6	b7	b8	b9	b10	b11	b12
0.97...	0.36...	0.21...	0.16...	0.18...	0.29...	0.48...	0.19...	0.76...	0.80...	0.24...	0.23...

**Delay**

r1
0.0
0.6321679247781751
0.6596079989405377
0.6596079989405377
0.6596079989405377
0.6510263541252814
0.668295589988081
0.7084492120248974
0.678572374519311

LANJUT KE PROSES TESTING

Dosen Pembimbing 1 : Imam Cholissodin, S.Si, M.Kom  
Dosen Pembimbing 2 : Candra Dewi, S.Kom, M.Sc

**Gambar 5.2 Implementasi Form Training**

### 5.2.3 Implementasi Form Testing

*Form testing* merupakan *form* yang digunakan untuk melakukan *testing* pada data uji. Form ini dapat dijalankan dengan menekan button *Proses Testing*, hasil dari proses ini dapat juga dilihat dalam bentuk *chart* dengan menekan button *Chart*. Implementasi *form testing* dapat dilihat pada Gambar 5.3, sementara implementasi *chart* hasil prediksi dapat dilihat pada Gambar 5.4.

135150200111002 - Daneswara Jauhari

### TESTING

Proses Testing

0.4115623677367388

Hasil Prediksi Chart

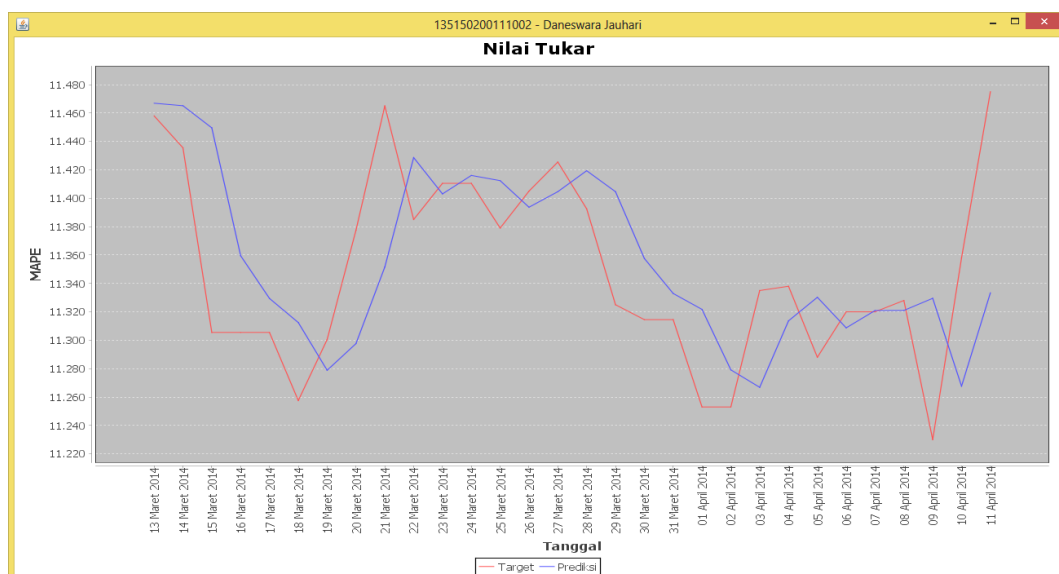
Tanggal	Target	Prediksi
13 Maret 2014	11458	11466,92
14 Maret 2014	11435,5	11465,16
15 Maret 2014	11305,5	11449,57
16 Maret 2014	11305,5	11359,52
17 Maret 2014	11305,5	11329,45
18 Maret 2014	11257,5	11312,45
19 Maret 2014	11300,5	11278,83
20 Maret 2014	11377,5	11297,62
21 Maret 2014	11465	11351,57
22 Maret 2014	11385	11428,64
23 Maret 2014	11410,5	11403,05
24 Maret 2014	11410,5	11416,05
25 Maret 2014	11379	11412,39
26 Maret 2014	11405	11393,67
27 Maret 2014	11425,5	11404,63

Log:

0.7116084458988077  
 0.7116230193508146  
 0.7134318894091809  
 0.7003216671423313  
 0.7142472830664879

Dosen Pembimbing 1 : Imam Cholissodin, S.Si, M.Kom  
 Dosen Pembimbing 2 : Candra Dewi, S.Kom, M.Sc

**Gambar 5.3 Implementasi Form Testing**



**Gambar 5.4 Implementasi Chart hasil prediksi**

## BAB 6 PENGUJIAN DAN ANALISIS

Bab ini menjelaskan hasil pengujian dan analisis dari uji coba yang telah dilakukan pada penelitian ini, hal ini berguna untuk mengetahui kinerja dari metode RELMNN dalam memprediksi permasalahan nilai tukar Rupiah Indonesia terhadap dolar Amerika Serikat. Pada bab ini terdapat 5 macam pengujian yang dilakukan, yaitu:

1. Pengujian jumlah *hidden neuron*
2. Pengujian jumlah fitur
3. Pengujian jangka waktu prediksi
4. Pengujian jumlah *context neuron*
5. Pengujian Perbandingan metode RELMNN dengan metode ELM

### 6.1 Hasil dan Analisis Uji Coba Jumlah *Hidden Neuron*

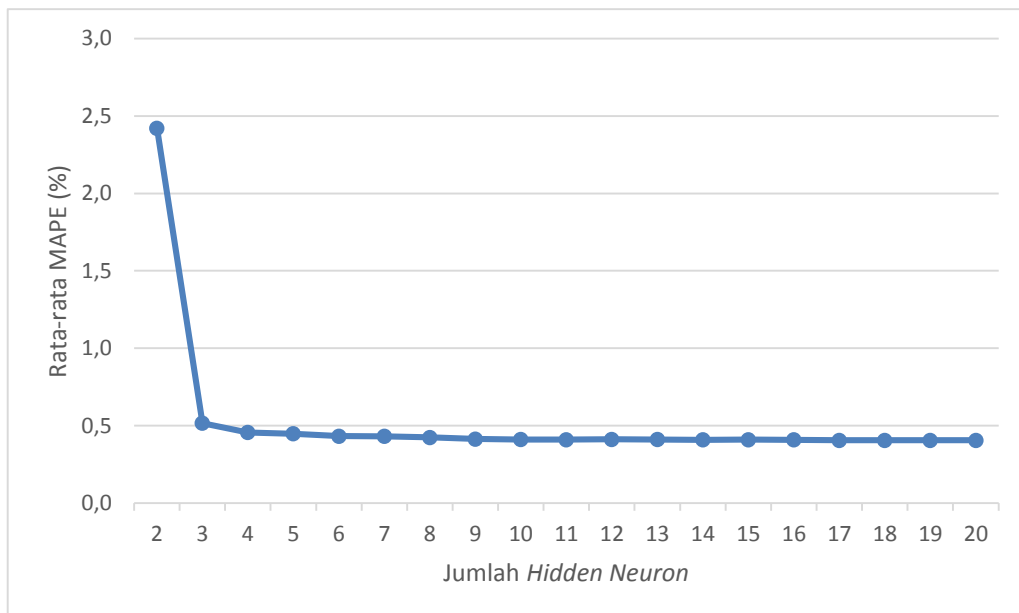
Pengujian berdasarkan *hidden neuron* digunakan untuk mengetahui jumlah *hidden neuron* yang memiliki nilai MAPE terbaik. Banyaknya *hidden neuron* yang di uji coba adalah dimulai dari 2 sampai 20. Inisialisasi parameter yang digunakan pada pengujian ini adalah bobot masukan dengan range  $[-1,1]$ , 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. Pengujian dilakukan sebanyak 10 kali untuk mendapatkan hasil yang dapat mempresentasikan kemampuan algoritme secara utuh. Hasil uji coba jumlah *hidden neuron* dapat dilihat pada Tabel 6.1 (hasil lengkap disertakan pada Lampiran B.1).

**Tabel 6.1 Hasil Uji Coba Jumlah *Hidden Neuron***

Jumlah <i>Hidden Neuron</i>	Nilai MAPE (%) Percobaan Ke- <i>i</i>						Rata- rata MAPE (%)
	1	2	3	...	9	10	
2	4,194604	0,817861	0,824841	...	1,653970	0,388327	<b>2,421157</b>
3	0,494568	0,527909	0,513291	...	0,441589	0,491982	<b>0,515774</b>
4	0,461076	0,484359	0,532880	...	0,415652	0,446810	<b>0,455642</b>
5	0,483653	0,402289	0,430690	...	0,438730	0,479428	<b>0,447870</b>
6	0,423355	0,399917	0,437309	...	0,443312	0,440056	<b>0,432083</b>
7	0,463171	0,418358	0,418353	...	0,413895	0,413200	<b>0,431255</b>
8	0,395567	0,409887	0,426667	...	0,476103	0,413006	<b>0,423598</b>
9	0,415378	0,416715	0,416166	...	0,417544	0,415131	<b>0,414057</b>
10	0,421155	0,409799	0,416234	...	0,410357	0,408321	<b>0,410153</b>
11	0,409599	0,414757	0,405008	...	0,398803	0,409052	<b>0,409208</b>
12	<b>0,410396</b>	<b>0,412163</b>	<b>0,409206</b>	...	<b>0,416620</b>	<b>0,413590</b>	<b>0,411797</b>
13	0,412895	0,409128	0,409317	...	0,405843	0,414406	<b>0,410476</b>

**Tabel 6.1 Hasil Uji Coba Jumlah *Hidden Neuron***

Jumlah <i>Hidden Neuron</i>	Nilai MAPE (%) Percobaan Ke- <i>i</i>						Rata-rata MAPE (%)
	1	2	3	...	9	10	
14	0,410539	0,411644	0,407630	...	0,414388	0,405912	<b>0,408615</b>
15	0,408395	0,410156	0,409469	...	0,410374	0,414316	<b>0,409794</b>
16	0,410152	0,403252	0,411284	...	0,409882	0,411031	<b>0,408251</b>
17	0,412265	0,404297	0,408259	...	0,410311	0,402056	<b>0,405628</b>
18	0,402874	0,407599	0,410834	...	0,403766	0,402318	<b>0,405555</b>
19	0,402683	0,414387	0,405218	...	0,403073	0,403268	<b>0,405639</b>
20	0,403148	0,402040	0,404558	...	0,407429	0,409225	<b>0,405268</b>



**Gambar 6.1 Grafik Rata-rata MAPE Uji Coba Jumlah *Hidden Neuron***

Grafik pada Gambar 6.1 dibuat untuk mempermudah dalam menganalisis jumlah *hidden neuron* terbaik berdasarkan Tabel 6.1. Berdasarkan grafik uji coba pada Gambar 6.1, 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 Hasil dan Analisis Uji Coba Jumlah Fitur

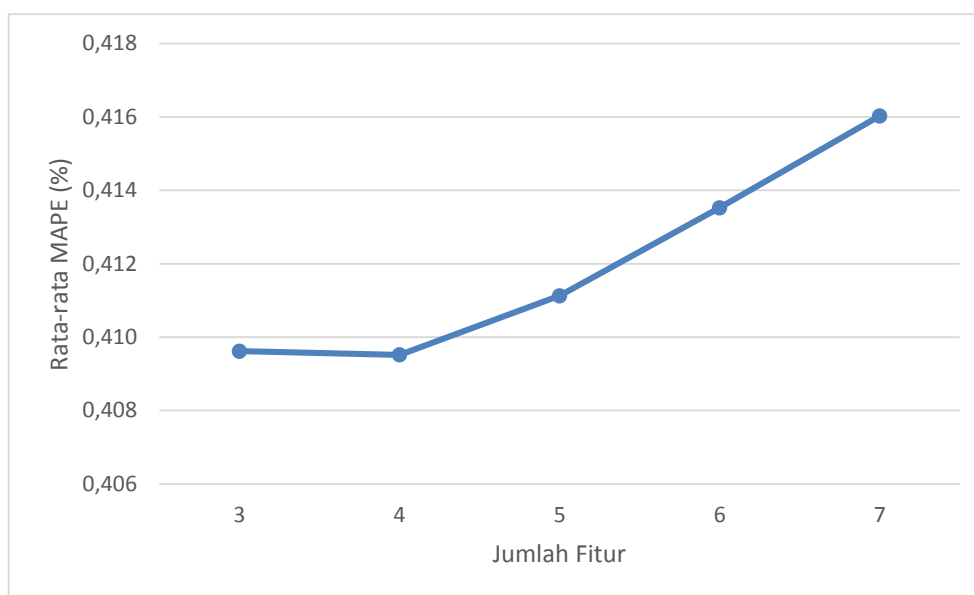
Pengujian berdasarkan jumlah fitur digunakan untuk mengetahui jumlah fitur yang memiliki nilai MAPE terbaik. Banyaknya jumlah fitur yang di uji coba adalah



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. Pengujian dilakukan sebanyak 10 kali untuk mendapatkan hasil yang dapat mempresentasikan kemampuan algoritme secara utuh. Hasil uji coba jumlah fitur dapat dilihat pada Tabel 6.2 (hasil lengkap disertakan pada Lampiran B.2).

**Tabel 6.2 Hasil Uji Coba Jumlah Fitur**

Jumlah Fitur	Nilai MAPE (%) Percobaan Ke- <i>i</i>						Rata-rata MAPE (%)
	1	2	3	...	9	10	
3	0,409799	0,409330	0,407776	...	0,411206	0,411012	<b>0,409616</b>
4	<b>0,408314</b>	<b>0,407615</b>	<b>0,410759</b>	...	<b>0,408858</b>	<b>0,412793</b>	<b>0,409519</b>
5	0,405412	0,410826	0,408745	...	0,413157	0,414362	<b>0,411125</b>
6	0,411955	0,413854	0,406034	...	0,409066	0,415022	<b>0,413524</b>
7	0,424462	0,408854	0,417926	...	0,406462	0,408994	<b>0,416031</b>



**Gambar 6.2 Grafik Rata-rata MAPE Uji Coba Jumlah Fitur**

Grafik pada Gambar 6.2 dibuat untuk mempermudah dalam menganalisis jumlah fitur terbaik berdasarkan Tabel 6.2. Berdasarkan grafik uji coba pada Gambar 6.2, 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 Hasil dan Analisis Uji Coba Jangka Waktu Prediksi

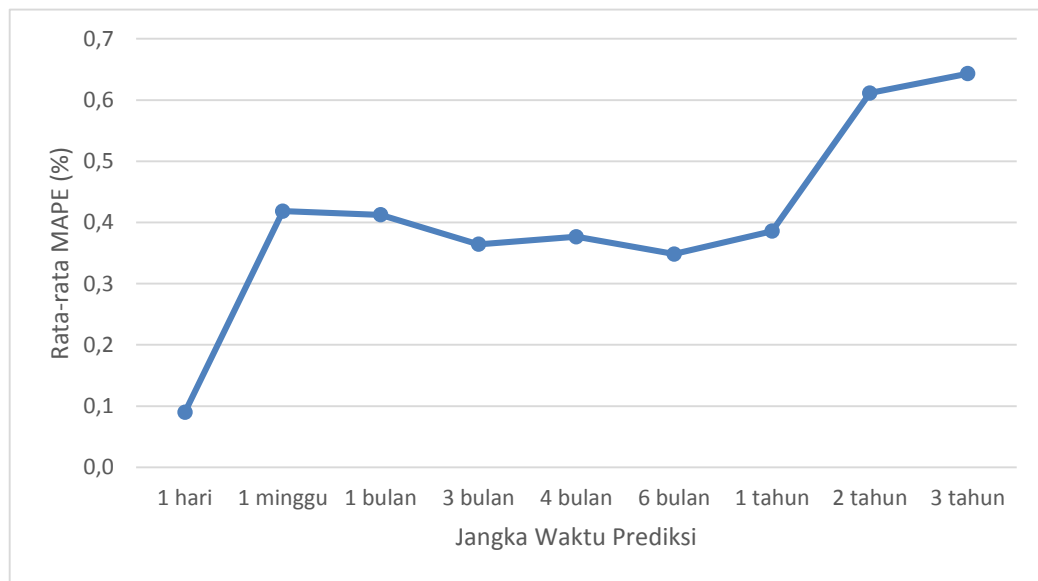
Pengujian berdasarkan jangka waktu prediksi digunakan untuk mengetahui akurasi prediksi pada jangka waktu tertentu. Terdapat 9 jangka waktu yang diuji, yaitu 1 hari, 1 minggu, 1 bulan, 3 bulan, 4 bulan, 6 bulan, 1 tahun, 2 tahun, dan 3 tahun. Inisialisasi parameter yang digunakan pada pengujian ini adalah 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. Pengujian dilakukan sebanyak 10 kali untuk mendapatkan hasil yang dapat mempresentasikan kemampuan algoritme secara utuh. Hasil uji coba jumlah jangka waktu prediksi dapat dilihat pada Tabel 6.3 (hasil lengkap disertakan pada Lampiran B.3).

**Tabel 6.3 Hasil Uji Coba Jangka Waktu Prediksi**

Jangka Waktu	Data Latih (hari)	Data Uji (hari)	Nilai MAPE (%) Percobaan Ke- <i>i</i>					Rata-rata MAPE (%)
			1	2	3	...	10	
1 hari	1890	1	0,131507	0,097036	0,061702	...	0,097046	0,089858
1 minggu	1890	7	0,410546	0,414849	0,418404	...	0,425062	0,418498
1 bulan	1890	30	0,408250	0,418982	0,408516	...	0,410150	0,412437
3 bulan	1890	91	0,359210	0,364372	0,367847	...	0,361371	0,364220
4 bulan	1890	121	0,375936	0,376387	0,376653	...	0,376369	0,376537
6 bulan	1890	182	0,348519	0,346427	0,346981	...	0,348457	0,348205
1 tahun	1890	365	0,386505	0,358192	0,370999	...	0,382200	0,385703
2 tahun	1890	730	0,668624	0,471223	1,191993	...	0,490973	0,611372
3 tahun	1890	1095	0,576049	0,605966	0,797463	...	0,629944	0,643144

Grafik pada Gambar 6.3 dibuat untuk mempermudah dalam menganalisis jangka waktu terbaik berdasarkan Tabel 6.3. Berdasarkan grafik uji coba pada Gambar 6.3, terlihat bahwa rata-rata nilai MAPE terkecil didapatkan ketika menggunakan jangka waktu 1 hari, dan hasil ini memiliki selisih yang cukup jauh dibandingkan jangka waktu yang lainnya dikarenakan ketika memprediksi dengan jangka waktu 1 hari, jaringan dengan mudah dapat mengenali pola data, hal ini dikarenakan jaringan telah banyak mengenali pola data dari data latih yang cukup banyak. 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 dinegara 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 6.3 Grafik Rata-rata MAPE Uji Coba Jangka Waktu Prediksi**

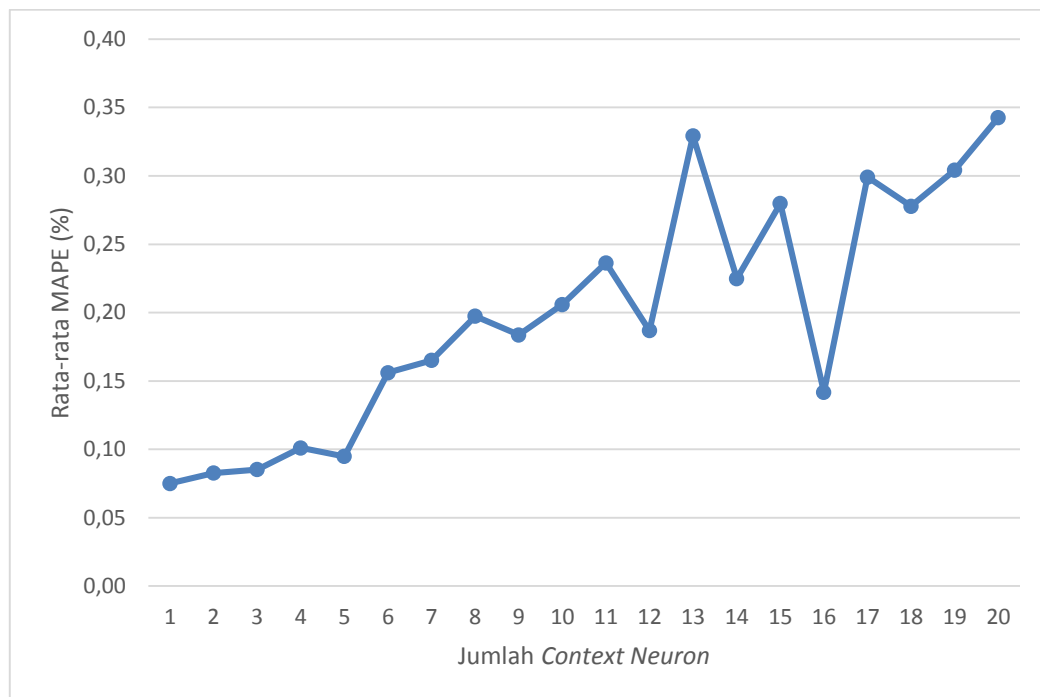
#### **6.4 Hasil dan Analisis Uji Coba Jumlah *Context Neuron***

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. Pengujian dilakukan sebanyak 10 kali untuk mendapatkan hasil yang dapat mempresentasikan kemampuan algoritme secara utuh. Hasil uji coba jumlah *context neuron* dapat dilihat pada Tabel 6.4 (hasil lengkap disertakan pada Lampiran B.4).

**Tabel 6.4 Hasil Uji Coba Jumlah *Context Neuron***

Jumlah <i>Context Neuron</i>	Nilai MAPE (%) Percobaan Ke- <i>i</i>						Rata-rata MAPE (%)
	1	2	3	...	9	10	
1	<b>0,071246</b>	<b>0,076168</b>	<b>0,071492</b>	...	<b>0,075213</b>	<b>0,077212</b>	<b>0,075023</b>
2	0,080920	0,082270	0,091132	...	0,094374	0,099021	<b>0,082729</b>
3	0,102717	0,092137	0,085597	...	0,074996	0,102798	<b>0,085228</b>
4	0,101466	0,068160	0,036046	...	0,145309	0,050698	<b>0,101039</b>
5	0,009337	0,131366	0,086204	...	0,108340	0,119445	<b>0,094897</b>
6	0,180178	0,108780	0,102611	...	0,271442	0,139981	<b>0,156042</b>
7	0,301898	0,278966	0,177707	...	0,335577	0,224930	<b>0,165120</b>
8	0,469265	0,124220	0,180003	...	0,307845	0,082155	<b>0,197412</b>
9	0,129554	0,071091	0,165559	...	0,205283	0,640905	<b>0,183673</b>
10	0,143206	0,139186	0,147279	...	0,538470	0,236599	<b>0,205946</b>
11	0,391520	0,125995	0,106897	...	0,004046	0,098521	<b>0,236409</b>
12	0,135382	0,132533	0,067419	...	0,136701	0,017767	<b>0,186831</b>
13	0,305822	0,504034	0,026216	...	0,088048	0,115886	<b>0,329265</b>
14	0,288916	0,463146	0,294471	...	0,138576	0,018387	<b>0,224857</b>
15	0,296255	0,159392	0,191865	...	0,042799	0,450507	<b>0,279846</b>
16	0,475175	0,002208	0,087375	...	0,029648	0,206279	<b>0,141801</b>
17	0,371534	0,039979	0,561387	...	0,360864	0,279395	<b>0,299113</b>
18	0,541836	0,054831	0,219748	...	0,114088	0,349023	<b>0,277803</b>
19	0,383842	0,193903	0,261909	...	0,108688	0,296979	<b>0,304110</b>
20	0,422098	0,175980	0,290490	...	0,185848	0,271589	<b>0,342697</b>

Grafik pada Gambar 6.4 dibuat untuk mempermudah dalam menganalisis jumlah *context neuron* terbaik berdasarkan Tabel 6.4. Berdasarkan grafik uji coba pada Gambar 6.4, 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.



**Gambar 6.4 Grafik Rata-rata MAPE Uji Coba Jumlah *Context Neuron***

## 6.5 Hasil dan Analisa Uji Coba Perbandingan Metode RELMNN dengan Metode ELM

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. Pengujian dilakukan sebanyak 10 kali untuk mendapatkan hasil yang dapat mempresentasikan kemampuan masing-masing algoritme secara utuh. Hasil uji coba perbandingan metode RELMNN dan metode ELM dengan jenis *datasets* yang berbeda terhadap nilai MAPE dapat dilihat pada Tabel 6.5 (hasil lengkap disertakan pada Lampiran B.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 6.6 (hasil lengkap disertakan pada Lampiran B.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 6.7 (hasil lengkap disertakan pada Lampiran B.7). *Datasets outliers* pada penelitian ini didapatkan dengan cara manual, yaitu dengan melihat grafik keseluruhan *datasets*.

**Tabel 6.5 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Nilai MAPE**

Metode (Jenis <i>Datasets</i> )	Nilai MAPE (%) Percobaan Ke- <i>i</i>					Rata-rata MAPE (%)
	1	2	3	...	10	
RELMNN ( <i>datasets</i> terdapat <i>outliers</i> )	0,062141	0,027497	0,230403	...	0,016901	<b>0,128711</b>
ELM ( <i>datasets</i> terdapat <i>outliers</i> )	0,066954	0,016593	0,045698	...	0,114762	<b>0,122319</b>
RELMNN ( <i>datasets</i> tanpa <i>outliers</i> )	<b>0,056300</b>	<b>0,078954</b>	<b>0,062583</b>	...	<b>0,077971</b>	<b>0,069502</b>
ELM ( <i>datasets</i> tanpa <i>outliers</i> )	0,065234	0,111507	0,079631	...	0,084324	<b>0,090423</b>

**Tabel 6.6 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Selisih Nilai Tukar Rupiah Indonesia terhadap Dolar Amerika**

Metode	Rata-rata Selisih Nilai Tukar antara Target dengan Prediksi (IDR) Percobaan Ke- <i>i</i>					Rata-rata Selisih (IDR)	Pembulatan (IDR)
	1	2	3	...	10		
RELMNN	6,450820	9,046567	7,170730	...	8,933864	7,963540	8
ELM	7,474515	12,776487	9,124114	...	9,661816	10,360598	10

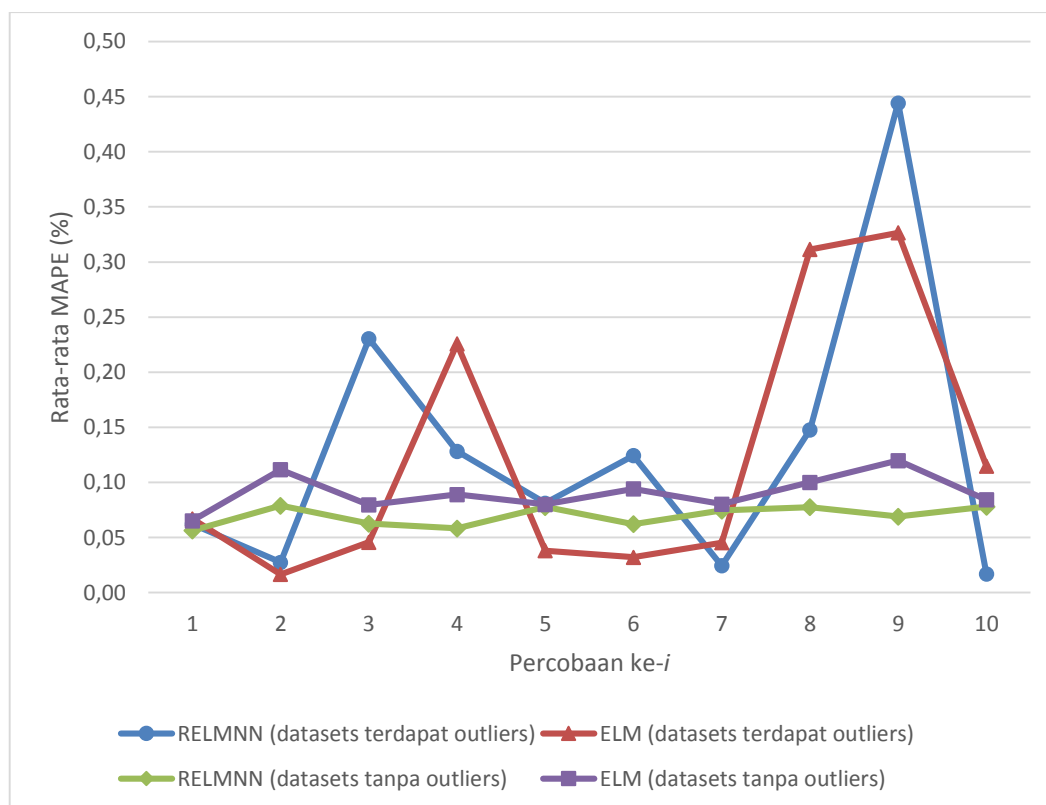
**Tabel 6.7 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Waktu yang Dibutuhkan Pada Proses *Training* dan *Testing***

Metode	Waktu yang Digunakan (Detik) Percobaan Ke- <i>i</i>						Rata-rata Waktu (detik)
	1	2	3	...	9	10	
RELMNN ( <i>training</i> )	0,024035	0,021425	0,012197	...	0,013894	0,011156	<b>0,015907</b>
ELM ( <i>training</i> )	0,015163	0,016483	0,011473	...	0,012624	0,012559	<b>0,014400</b>
RELMNN ( <i>testing</i> )	0,001871	0,002013	0,001096	...	0,001792	0,000753	<b>0,001359</b>
ELM ( <i>testing</i> )	0,000878	0,000866	0,004333	...	0,001015	0,001143	<b>0,001849</b>

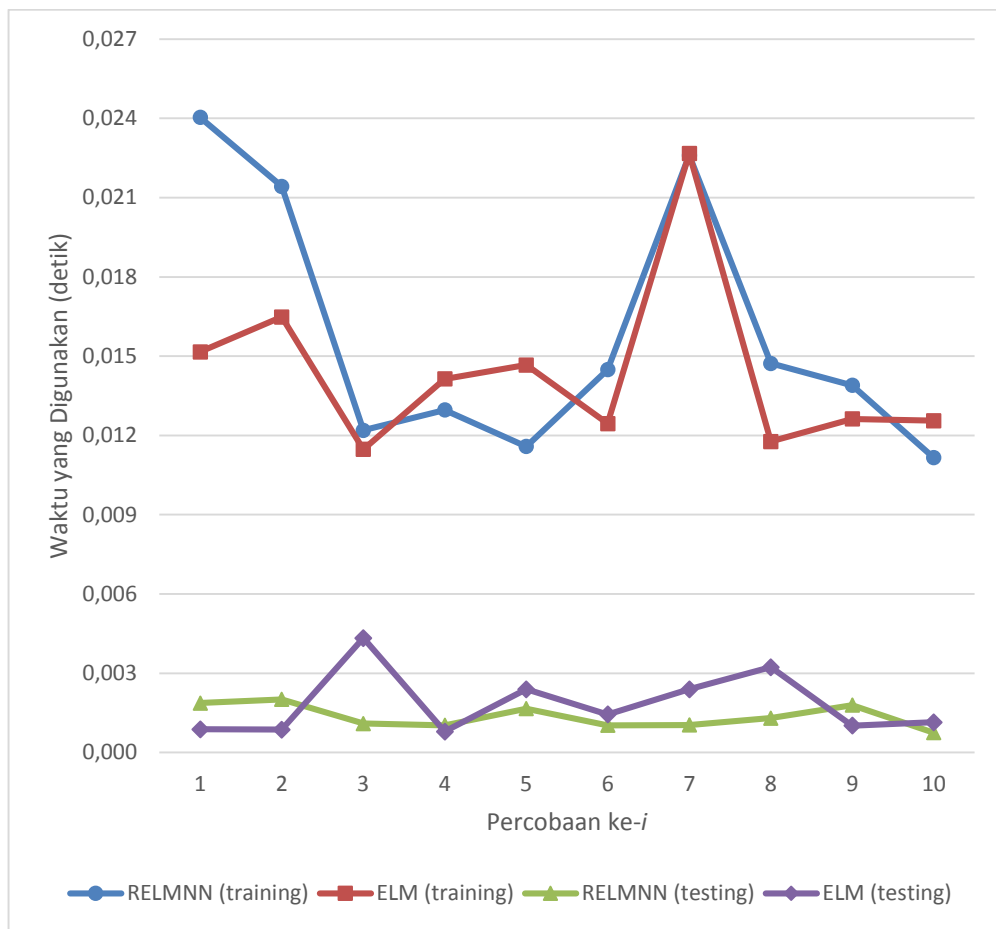
Grafik pada Gambar 6.5 dibuat untuk mempermudah dalam menganalisis metode terbaik dengan nilai MAPE terkecil berdasarkan Tabel 6.5. Berdasarkan grafik uji coba pada Gambar 6.5, terlihat bahwa pada setiap percobaan perbandingan metode dengan jenis *datasets* tanpa *outliers* nilai MAPE terkecil didapatkan 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, walaupun ELM lebih unggul sedikit dari pada RELMNN. Hasil tersebut membuktikan bahwa kedua metode yang digunakan masih belum bisa menangani data yang *outliers*. Berdasarkan Tabel 6.6 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 6.6 dibuat untuk mempermudah dalam menganalisis metode terbaik dengan waktu yang digunakan paling kecil berdasarkan Tabel 6.7. Berdasarkan grafik uji coba pada Gambar 6.6, 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 6.5 Grafik Nilai MAPE Uji Coba Perbandingan Metode RELMNN dengan Metode ELM**



**Gambar 6.6 Grafik Waktu Uji Coba Perbandingan Metode RELMNN dengan Metode ELM**



## BAB 7 PENUTUP

Bab ini menjelaskan kesimpulan dari penelitian ini dan memberikan saran untuk penelitian selanjutnya.

### 7.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis dari prediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika menggunakan metode *Recurrent Extreme Learning Machine Neural Network* maka diperoleh kesimpulan sebagai berikut:

1. Metode RELMNN dapat diimplementasikan untuk memprediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat dengan nilai evaluasi yang lebih baik, jika dibandingkan dengan metode ELM. Adapun alurnya adalah sebagai berikut:
  - a. 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*.
  - b. Melakukan proses *training* dengan data latih yang didapatkan dari sebagian *datasets* nilai tukar yang telah dinormalisasi.
  - c. Melakukan proses *testing* dengan data uji dan juga bobot masukan, bias dan bobot luaran dari proses *training*. Dari hasil *testing* maka didapatkan hasil prediksi.
  - d. Melakukan denormalisasi hasil prediksi sehingga mendapatkan nilai dalam bentuk nilai tukar.
  - e. Menghitung nilai evaluasi dari hasil prediksi nilai tukar uang Rupiah Indonesia terhadap Dolar Amerika Serikat.
2. Nilai evaluasi berupa rata-rata nilai MAPE terbaik dengan menggunakan metode RELMNN didapatkan sebesar 0,069502%. Nilai tersebut didapatkan dengan menggunakan jenis *datasets* tanpa *outliers*, 12 *hidden neuron*, 4 fitur, jangka waktu 1 hari dan 1 *context neuron*.
3. Perbandingan nilai evaluasi dalam bentuk nilai MAPE yang dihasilkan antara metode RELMNN dan ELM dalam memprediksi nilai tukar Rupiah Indonesia terhadap Dolar Amerika Serikat adalah 0,069502:0,090423 (RELMNN:ELM). Pada hasil perbandingan tersebut, didapatkan rata-rata selisih nilai tukar antara target dengan prediksi mencapai 8 rupiah untuk metode RELMNN, sementara ELM mencapai 10 rupiah.

## 7.2 Saran

Berdasarkan penelitian yang telah dilakukan, didapatkan beberapa saran yang dapat dikembangkan untuk penelitian selanjutnya, berikut saran-saran yang didapatkan:

1. Penelitian lebih lanjut dapat melakukan proses khusus untuk menangani *datasets outliers* secara otomatis, sehingga hasil dari prediksi bisa tetap stabil walaupun terdapat *datasets outliers*.
2. Melakukan pengujian perbandingan nilai evaluasi ketika memasukkan nilai *context neuron* dengan nilai tukar aktual dibandingkan dengan menggunakan nilai tukar hasil prediksi, sehingga dapat mengetahui bagaimana kinerja RELMNN ketika melakukan prediksi dengan nilai *context neuron* yang didapatkan dari hasil prediksi sebelumnya.

## DAFTAR PUSTAKA

- Azizah, A.N., 2016. *Penentuan Kualitas Air Sungai Menggunakan Metode Extreme Learning Machine*. Universitas Brawijaya.
- Bailliu, J. dan King, M.R., 2005. What Drives Movements in Exchange Rates ? *Bank of Canada Review*, [daring] hal.3–16. Tersedia pada: <<http://ssrn.com/abstract=896324>>.
- Chen, X., Dong, Z.Y., Meng, K., Xu, Y., Wong, K.P. dan Ngan, H.W., 2012. Electricity price forecasting with extreme learning machine and bootstrapping. *IEEE Transactions on Power Systems*, 27(4), hal.2055–2062.
- Cholissodin, I., Sutrisno, S., Soebroto, A.A., Hanum, L. dan Caesar, C.A., 2017. Optimasi Kandungan Gizi Susu Kambing Peranakan Etawa (PE) Menggunakan ELM-PSO Di UPT Pembibitan Ternak Dan Hijauan Makanan Ternak Singosari-Malang. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, 4(1).
- Darmayasa, P., 2015. *Operasi Baris Elementer (OBE) dan Penerapannya*. [daring] Tersedia pada: <<http://www.konsep-matematika.com/2015/09/operasi-baris-elementer-obe-dan-penerapannya.html>> [Diakses 29 Mar 2017].
- Ertugrul, Ö.F., 2016. Forecasting electricity load by a novel recurrent extreme learning machines approach. *International Journal of Electrical Power and Energy Systems*, 78, hal.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), hal.84–90.
- He, H. dan Shen, X., 2007. Bootstrap methods for foreign currency exchange rates prediction. *IEEE International Conference on Neural Networks - Conference Proceedings*, hal.1272–1277.
- Herdianto, 2013. *Prediksi Kerusakan Motor Induksi Menggunakan Metode Jaringan Saraf Tiruan Backpropagation*. Universitas Sumatera Utara.
- 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*, hal.22–26.
- Irawan, A.F., 2015. *Implementasi Metode Jaringan Syaraf Tiruan Backpropagation pada Peramalan Kebutuhan Air PDAM Kota Malang*. Universitas Brawijaya.
- Jauhari, D., Hanafi, A., Y, M.F.A., Satria, A.R. dan H, L.H., 2016a. Prediksi Nilai Tukar Rupiah Terhadap US Dollar Menggunakan Metode Genetic Programming. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, 3(4), hal.285–291.

- Jauhari, D., Himawan, A. dan Dewi, C., 2016b. Prediksi Distribusi Air PDAM Menggunakan Metode Jaringan Syaraf Tiruan Backpropagation Di PDAM Kota Malang. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, 3(2), hal.85–89.
- 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), hal.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), hal.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), hal.15–19.
- Nikmah, N.U., 2014. *Prediksi Kebutuhan Air PDAM Berdasarkan Jumlah Pelanggan Menggunakan Al-Alaoui Backpropagation*. Universitas Brawijaya.
- 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.
- Puspaningrum, D., 2006. *Pengantar Jaringan Syaraf Tiruan*. Yogyakarta: Andi Offset.
- Rehman, M., Khan, G.M. dan Mahmud, S.A., 2014. Foreign Currency Exchange Rates Prediction Using CGP and Recurrent Neural Network. *IERI Procedia*, 10, hal.239–244.
- S, K.M. dan W, S.P., 2015. Analisa Dan Penerapan Metode Single Exponential Smoothing Untuk Prediksi Penjualan Pada Periode Tertentu (Studi Kasus : PT. Media Cemara Kreasi). *Prosiding SNATIF Ke-2 Tahun 2015*, hal.259–266.
- 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*, [daring] 247(3), hal.831–846. Tersedia pada: <<http://dx.doi.org/10.1016/j.ejor.2015.06.052>>.
- Simorangkir, I. dan Suseno, 2004. *Sistem dan Kebijakan Nilai Tukar. Pusat Pendidikan dan Studi Kebanksentralan (PPSK) Bank Indonesia*, Jakarta: Pusat Pendidikan dan Studi Kebanksentralan (PPSK) Bank Indonesia.
- 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.

- Wan, C., Xu, Z., Pinson, P., Dong, Z.Y. dan Wong, K.P., 2014. Probabilistic forecasting of wind power generation using extreme learning machine. *IEEE Transactions on Power Systems*, 29(3), hal.1033–1044.
- 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*, hal.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), hal.3319–3326.

## LAMPIRAN A DATA NILAI TUKAR RUPIAH INDONESIA TERHADAP DOLAR AMERIKA SERIKAT

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
01 Januari 2009	11255	06 Februari 2009	11649	14 Maret 2009	11926
02 Januari 2009	11196	07 Februari 2009	11649	15 Maret 2009	11926
03 Januari 2009	11196	08 Februari 2009	11649	16 Maret 2009	11985
04 Januari 2009	11196	09 Februari 2009	11573	17 Maret 2009	12016,5
05 Januari 2009	11040	10 Februari 2009	11779,5	18 Maret 2009	11986
06 Januari 2009	11021,5	11 Februari 2009	11898,7	19 Maret 2009	11654
07 Januari 2009	11030,5	12 Februari 2009	11780,9	20 Maret 2009	11985
08 Januari 2009	10946	13 Februari 2009	11758	21 Maret 2009	11985
09 Januari 2009	11075,5	14 Februari 2009	11758	22 Maret 2009	11985
10 Januari 2009	11075,5	15 Februari 2009	11758	23 Maret 2009	11697
11 Januari 2009	11075,5	16 Februari 2009	11938,4	24 Maret 2009	11698,5
12 Januari 2009	11035	17 Februari 2009	11942,5	25 Maret 2009	11856,5
13 Januari 2009	11116,5	18 Februari 2009	12125	26 Maret 2009	11618
14 Januari 2009	11306	19 Februari 2009	11951	27 Maret 2009	11630
15 Januari 2009	11165	20 Februari 2009	11972	28 Maret 2009	11630
16 Januari 2009	11177	21 Februari 2009	11972	29 Maret 2009	11630
17 Januari 2009	11177	22 Februari 2009	11972	30 Maret 2009	11535,5
18 Januari 2009	11177	23 Februari 2009	11754,5	31 Maret 2009	11584
19 Januari 2009	11001,5	24 Februari 2009	11995,5	01 April 2009	11593
20 Januari 2009	11114,5	25 Februari 2009	11845	02 April 2009	11598
21 Januari 2009	11335,5	26 Februari 2009	11933,5	03 April 2009	11483,5
22 Januari 2009	11337,5	27 Februari 2009	12037,5	04 April 2009	11483,5
23 Januari 2009	11203,5	28 Februari 2009	12037,5	05 April 2009	11483,5
24 Januari 2009	11203,5	01 Maret 2009	12037,5	06 April 2009	11372,5
25 Januari 2009	11203,5	02 Maret 2009	12060,5	07 April 2009	11488,5
26 Januari 2009	11126,8	03 Maret 2009	12166	08 April 2009	11350,5
27 Januari 2009	11078	04 Maret 2009	11981	09 April 2009	11381,8
28 Januari 2009	11121	05 Maret 2009	11961	10 April 2009	11381,8
29 Januari 2009	11347	06 Maret 2009	12025	11 April 2009	11381,8
30 Januari 2009	11430	07 Maret 2009	12025	12 April 2009	11381,8
31 Januari 2009	11430	08 Maret 2009	12025	13 April 2009	11336
01 Februari 2009	11430	09 Maret 2009	12073,5	14 April 2009	10905,6
02 Februari 2009	11423,5	10 Maret 2009	12113	15 April 2009	10907
03 Februari 2009	11393	11 Maret 2009	11988,5	16 April 2009	10871
04 Februari 2009	11682	12 Maret 2009	11915,5	17 April 2009	10871
05 Februari 2009	11825,5	13 Maret 2009	11926	18 April 2009	10871

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
19 April 2009	10871	29 Mei 2009	10365,5	08 Juli 2009	10284,1
20 April 2009	11107	30 Mei 2009	10365,5	09 Juli 2009	10299
21 April 2009	10703,5	31 Mei 2009	10365,5	10 Juli 2009	10198,5
22 April 2009	10827,5	01 Juni 2009	10200,7	11 Juli 2009	10302
23 April 2009	10859,5	02 Juni 2009	10179,5	12 Juli 2009	10302
24 April 2009	10859,5	03 Juni 2009	10249,5	13 Juli 2009	10302
25 April 2009	10859,5	04 Juni 2009	10155	14 Juli 2009	10302
26 April 2009	10859,5	05 Juni 2009	10100,6	15 Juli 2009	10302
27 April 2009	10741	06 Juni 2009	10100,6	16 Juli 2009	10302
28 April 2009	10910,5	07 Juni 2009	10100,6	17 Juli 2009	10302
29 April 2009	10809,5	08 Juni 2009	9817,9	18 Juli 2009	10145
30 April 2009	10724,5	09 Juni 2009	10043,3	19 Juli 2009	10145
01 Mei 2009	10743,5	10 Juni 2009	10049,4	20 Juli 2009	10145
02 Mei 2009	10743,5	11 Juni 2009	10038	21 Juli 2009	10158
03 Mei 2009	10743,5	12 Juni 2009	10110,5	22 Juli 2009	10028
04 Mei 2009	10640,5	13 Juni 2009	10110,5	23 Juli 2009	9983,8
05 Mei 2009	10541,5	14 Juni 2009	10110,5	24 Juli 2009	10048,5
06 Mei 2009	10404	15 Juni 2009	10114,5	25 Juli 2009	10029,5
07 Mei 2009	10404	16 Juni 2009	10213	26 Juli 2009	10051,5
08 Mei 2009	10384	17 Juni 2009	10171,3	27 Juli 2009	10051,5
09 Mei 2009	10606	18 Juni 2009	10427	28 Juli 2009	10034,5
10 Mei 2009	10606	19 Juni 2009	10402,8	29 Juli 2009	9987,35
11 Mei 2009	10431	20 Juni 2009	10402,8	30 Juli 2009	10094
12 Mei 2009	10455,5	21 Juni 2009	10402,8	31 Juli 2009	9914,95
13 Mei 2009	10318,7	22 Juni 2009	10477,5	01 Agustus 2009	9786,85
14 Mei 2009	10382,5	23 Juni 2009	10210	02 Agustus 2009	9791,65
15 Mei 2009	10378	24 Juni 2009	10351	03 Agustus 2009	9791,65
16 Mei 2009	10370,5	25 Juni 2009	10350	04 Agustus 2009	9678,3
17 Mei 2009	10370,5	26 Juni 2009	10346	05 Agustus 2009	9693,45
18 Mei 2009	10378,8	27 Juni 2009	10277	06 Agustus 2009	9866,6
19 Mei 2009	10298	28 Juni 2009	10346	07 Agustus 2009	9920,9
20 Mei 2009	10327,5	29 Juni 2009	10277	08 Agustus 2009	9956,2
21 Mei 2009	10377,5	30 Juni 2009	10268	09 Agustus 2009	9965,4
22 Mei 2009	10201,2	01 Juli 2009	10268	10 Agustus 2009	9965,4
23 Mei 2009	10201,2	02 Juli 2009	10146,2	11 Agustus 2009	9945,45
24 Mei 2009	10201,2	03 Juli 2009	10213,9	12 Agustus 2009	9935,7
25 Mei 2009	10183,5	04 Juli 2009	10213,9	13 Agustus 2009	9956,05
26 Mei 2009	10195,5	05 Juli 2009	10213,9	14 Agustus 2009	9944,85
27 Mei 2009	10345	06 Juli 2009	10243	15 Agustus 2009	9937,43
28 Mei 2009	10643,5	07 Juli 2009	10243	16 Agustus 2009	9937,43

Tanggal	Nilai Tukar (IDR)
17 Agustus 2009	9930,01
18 Agustus 2009	10039,5
19 Agustus 2009	10068,5
20 Agustus 2009	9974,75
21 Agustus 2009	10073
22 Agustus 2009	10016
23 Agustus 2009	10010,3
24 Agustus 2009	10010,3
25 Agustus 2009	10010,8
26 Agustus 2009	10000,9
27 Agustus 2009	10063
28 Agustus 2009	10122
29 Agustus 2009	10021,5
30 Agustus 2009	10054,5
31 Agustus 2009	10016
01 September 2009	10104
02 September 2009	10135,5
03 September 2009	10094,5
04 September 2009	10138
05 September 2009	10101
06 September 2009	10134
07 September 2009	10109,5
08 September 2009	10095
09 September 2009	9882,85
10 September 2009	9922,7
11 September 2009	9891,25
12 September 2009	9904,55
13 September 2009	9904,3
14 September 2009	9902,6
15 September 2009	9887,75
16 September 2009	9844,2
17 September 2009	9701,9
18 September 2009	9484,35
19 September 2009	9695,5
20 September 2009	9704,2
21 September 2009	9704,2
22 September 2009	9724,2
23 September 2009	9654,2
24 September 2009	9613,75
25 September 2009	9709,35

Tanggal	Nilai Tukar (IDR)
26 September 2009	9693,1
27 September 2009	9679,55
28 September 2009	9679,55
29 September 2009	9719,15
30 September 2009	9695,45
01 Oktober 2009	9630,65
02 Oktober 2009	9720,1
03 Oktober 2009	9620,15
04 Oktober 2009	9634,6
05 Oktober 2009	9634,6
06 Oktober 2009	9594,15
07 Oktober 2009	9539,1
08 Oktober 2009	9444,6
09 Oktober 2009	9369,55
10 Oktober 2009	9431,2
11 Oktober 2009	9418,3
12 Oktober 2009	9418,3
13 Oktober 2009	9367,05
14 Oktober 2009	9340,55
15 Oktober 2009	9379,9
16 Oktober 2009	9359,5
17 Oktober 2009	9360,45
18 Oktober 2009	9394,3
19 Oktober 2009	9394,3
20 Oktober 2009	9398,8
21 Oktober 2009	9384,2
22 Oktober 2009	9315,8
23 Oktober 2009	9425,6
24 Oktober 2009	9408,95
25 Oktober 2009	9425,6
26 Oktober 2009	9425,6
27 Oktober 2009	9492,65
28 Oktober 2009	9561,95
29 Oktober 2009	9613,3
30 Oktober 2009	9578
31 Oktober 2009	9611
01 November 2009	9609
02 November 2009	9609
03 November 2009	9547,75
04 November 2009	9659,25

Tanggal	Nilai Tukar (IDR)
05 November 2009	9525,55
06 November 2009	9513,7
07 November 2009	9480,9
08 November 2009	9462,85
09 November 2009	9462,85
10 November 2009	9365,55
11 November 2009	9396,8
12 November 2009	9390,85
13 November 2009	9431,25
14 November 2009	9342,25
15 November 2009	9328,45
16 November 2009	9328,45
17 November 2009	9285,8
18 November 2009	9424,25
19 November 2009	9461,75
20 November 2009	9526
21 November 2009	9500,9
22 November 2009	9499,65
23 November 2009	9499,65
24 November 2009	9458,8
25 November 2009	9446,65
26 November 2009	9355,4
27 November 2009	9469,45
28 November 2009	9482,75
29 November 2009	9403,4
30 November 2009	9403,4
01 Desember 2009	9393,35
02 Desember 2009	9321,8
03 Desember 2009	9349,75
04 Desember 2009	9421,7
05 Desember 2009	9551,8
06 Desember 2009	9549,25
07 Desember 2009	9549,25
08 Desember 2009	9591,15
09 Desember 2009	9641,55
10 Desember 2009	9654,4
11 Desember 2009	9657,9
12 Desember 2009	9732,85
13 Desember 2009	9736,95
14 Desember 2009	9736,95



Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
15 Desember 2009	9455	24 Januari 2010	9340	05 Maret 2010	9280
16 Desember 2009	9470	25 Januari 2010	9340	06 Maret 2010	9235
17 Desember 2009	9487,85	26 Januari 2010	9335	07 Maret 2010	9235
18 Desember 2009	9659,85	27 Januari 2010	9380	08 Maret 2010	9235
19 Desember 2009	9686,6	28 Januari 2010	9395	09 Maret 2010	9175
20 Desember 2009	9640	29 Januari 2010	9350	10 Maret 2010	9199
21 Desember 2009	9640	30 Januari 2010	9350	11 Maret 2010	9165
22 Desember 2009	9658,5	31 Januari 2010	9350	12 Maret 2010	9195
23 Desember 2009	9526,65	01 Februari 2010	9350	13 Maret 2010	9160
24 Desember 2009	9460,85	02 Februari 2010	9395	14 Maret 2010	9160
25 Desember 2009	9459,65	03 Februari 2010	9380	15 Maret 2010	9160
26 Desember 2009	9439,15	04 Februari 2010	9290	16 Maret 2010	9165
27 Desember 2009	9438,85	05 Februari 2010	9330	17 Maret 2010	9165
28 Desember 2009	9438,85	06 Februari 2010	9385	18 Maret 2010	9118
29 Desember 2009	9442,85	07 Februari 2010	9385	19 Maret 2010	9125
30 Desember 2009	9472,85	08 Februari 2010	9385	20 Maret 2010	9172
31 Desember 2009	9411,55	09 Februari 2010	9395	21 Maret 2010	9172
01 Januari 2010	9427,85	10 Februari 2010	9373	22 Maret 2010	9172
02 Januari 2010	9392	11 Februari 2010	9360	23 Maret 2010	9121
03 Januari 2010	9399,5	12 Februari 2010	9370	24 Maret 2010	9110
04 Januari 2010	9399,5	13 Februari 2010	9330	25 Maret 2010	9117
05 Januari 2010	9378,35	14 Februari 2010	9330	26 Maret 2010	9134
06 Januari 2010	9320	15 Februari 2010	9330	27 Maret 2010	9120
07 Januari 2010	9253	16 Februari 2010	9344	28 Maret 2010	9120
08 Januari 2010	9235	17 Februari 2010	9322	29 Maret 2010	9120
09 Januari 2010	9215	18 Februari 2010	9270	30 Maret 2010	9089
10 Januari 2010	9215	19 Februari 2010	9313	31 Maret 2010	9090
11 Januari 2010	9215	20 Februari 2010	9355	01 April 2010	9090
12 Januari 2010	9130	21 Februari 2010	9355	02 April 2010	9065
13 Januari 2010	9156	22 Februari 2010	9355	03 April 2010	9075
14 Januari 2010	9170	23 Februari 2010	9300	04 April 2010	9075
15 Januari 2010	9165	24 Februari 2010	9303	05 April 2010	9075
16 Januari 2010	9195	25 Februari 2010	9305	06 April 2010	9070
17 Januari 2010	9195	26 Februari 2010	9343	07 April 2010	9048
18 Januari 2010	9195	27 Februari 2010	9335	08 April 2010	9050
19 Januari 2010	9230	28 Februari 2010	9335	09 April 2010	9050
20 Januari 2010	9234	01 Maret 2010	9335	10 April 2010	9028
21 Januari 2010	9315	02 Maret 2010	9265	11 April 2010	9028
22 Januari 2010	9283	03 Maret 2010	9300	12 April 2010	9028
23 Januari 2010	9340	04 Maret 2010	9288	13 April 2010	9020

Tanggal	Nilai Tukar (IDR)
14 April 2010	9035
15 April 2010	9008
16 April 2010	9006
17 April 2010	9016
18 April 2010	9016
19 April 2010	9016
20 April 2010	9040
21 April 2010	9020
22 April 2010	9010
23 April 2010	9020
24 April 2010	9010
25 April 2010	9010
26 April 2010	9010
27 April 2010	9005
28 April 2010	9006
29 April 2010	9041
30 April 2010	9020
01 Mei 2010	9012
02 Mei 2010	9012
03 Mei 2010	9012
04 Mei 2010	9029
05 Mei 2010	9023
06 Mei 2010	9100
07 Mei 2010	9210
08 Mei 2010	9190
09 Mei 2010	9190
10 Mei 2010	9190
11 Mei 2010	9075
12 Mei 2010	9089
13 Mei 2010	9085
14 Mei 2010	9090
15 Mei 2010	9090
16 Mei 2010	9160
17 Mei 2010	9090
18 Mei 2010	9130
19 Mei 2010	9105
20 Mei 2010	9155
21 Mei 2010	9250
22 Mei 2010	9270
23 Mei 2010	9270

Tanggal	Nilai Tukar (IDR)
24 Mei 2010	9270
25 Mei 2010	9275
26 Mei 2010	9360
27 Mei 2010	9335
28 Mei 2010	9260
29 Mei 2010	9260
30 Mei 2010	9260
31 Mei 2010	9260
01 Juni 2010	9162
02 Juni 2010	9218
03 Juni 2010	9225
04 Juni 2010	9185
05 Juni 2010	9270
06 Juni 2010	9180
07 Juni 2010	9180
08 Juni 2010	9255
09 Juni 2010	9250
10 Juni 2010	9250
11 Juni 2010	9245
12 Juni 2010	9200
13 Juni 2010	9200
14 Juni 2010	9200
15 Juni 2010	9178
16 Juni 2010	9160
17 Juni 2010	9152
18 Juni 2010	9153
19 Juni 2010	9118
20 Juni 2010	9118
21 Juni 2010	9118
22 Juni 2010	9015
23 Juni 2010	9025
24 Juni 2010	9045
25 Juni 2010	9045
26 Juni 2010	9065
27 Juni 2010	9065
28 Juni 2010	9055
29 Juni 2010	9023
30 Juni 2010	9070
01 Juli 2010	9060
02 Juli 2010	9072

Tanggal	Nilai Tukar (IDR)
03 Juli 2010	9055
04 Juli 2010	9055
05 Juli 2010	9055
06 Juli 2010	9063
07 Juli 2010	9058
08 Juli 2010	9071
09 Juli 2010	9065
10 Juli 2010	9060
11 Juli 2010	9060
12 Juli 2010	9060
13 Juli 2010	9051
14 Juli 2010	9050
15 Juli 2010	9049
16 Juli 2010	9050
17 Juli 2010	9075
18 Juli 2010	9052
19 Juli 2010	9052
20 Juli 2010	9060
21 Juli 2010	9058
22 Juli 2010	9053
23 Juli 2010	9050
24 Juli 2010	9040
25 Juli 2010	9049
26 Juli 2010	9032,5
27 Juli 2010	9015,5
28 Juli 2010	9013
29 Juli 2010	9004
30 Juli 2010	9005
31 Juli 2010	8952
01 Agustus 2010	8952
02 Agustus 2010	8952
03 Agustus 2010	8945
04 Agustus 2010	8947
05 Agustus 2010	8950
06 Agustus 2010	8938
07 Agustus 2010	8938
08 Agustus 2010	8938
09 Agustus 2010	8938
10 Agustus 2010	8931
11 Agustus 2010	8955

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
12 Agustus 2010	8972	21 September 2010	8978	31 Oktober 2010	8927
13 Agustus 2010	9010	22 September 2010	8966	01 November 2010	8927
14 Agustus 2010	8970	23 September 2010	8958	02 November 2010	8920
15 Agustus 2010	8970	24 September 2010	8953	03 November 2010	8928,3
16 Agustus 2010	8970	25 September 2010	8955	04 November 2010	8840,7
17 Agustus 2010	8980	26 September 2010	8955	05 November 2010	8925,1
18 Agustus 2010	8980	27 September 2010	8953	06 November 2010	8908,4
19 Agustus 2010	8973	28 September 2010	8953	07 November 2010	8905
20 Agustus 2010	8963	29 September 2010	8954	08 November 2010	8905
21 Agustus 2010	8973	30 September 2010	8920	09 November 2010	8920,8
22 Agustus 2010	8973	01 Oktober 2010	8922	10 November 2010	8904,2
23 Agustus 2010	8973	02 Oktober 2010	8923	11 November 2010	8923,8
24 Agustus 2010	8967	03 Oktober 2010	8925	12 November 2010	8905,2
25 Agustus 2010	8978	04 Oktober 2010	8923	13 November 2010	8934,2
26 Agustus 2010	8978	05 Oktober 2010	8922	14 November 2010	8935,8
27 Agustus 2010	8979	06 Oktober 2010	8945	15 November 2010	8935,8
28 Agustus 2010	9015	07 Oktober 2010	8921	16 November 2010	8965
29 Agustus 2010	9015	08 Oktober 2010	8924	17 November 2010	8945,4
30 Agustus 2010	9015	09 Oktober 2010	8933	18 November 2010	8943,6
31 Agustus 2010	9018	10 Oktober 2010	8933	19 November 2010	8946
01 September 2010	9049	11 Oktober 2010	8933	20 November 2010	8980
02 September 2010	9013	12 Oktober 2010	9010	21 November 2010	8944
03 September 2010	9008	13 Oktober 2010	8930	22 November 2010	8944
04 September 2010	9004	14 Oktober 2010	8925	23 November 2010	8929,2
05 September 2010	9004	15 Oktober 2010	8923	24 November 2010	8959,2
06 September 2010	9004	16 Oktober 2010	8923	25 November 2010	8962
07 September 2010	8986	17 Oktober 2010	8923	26 November 2010	8962,1
08 September 2010	9015	18 Oktober 2010	8923	27 November 2010	9015
09 September 2010	9000	19 Oktober 2010	8933	28 November 2010	9015
10 September 2010	9000	20 Oktober 2010	8926	29 November 2010	9009,9
11 September 2010	8950	21 Oktober 2010	8933	30 November 2010	9060
12 September 2010	9000	22 Oktober 2010	8935	01 Desember 2010	9030,6
13 September 2010	9000	23 Oktober 2010	8937	02 Desember 2010	9010
14 September 2010	9000	24 Oktober 2010	8935	03 Desember 2010	9016
15 September 2010	8958	25 Oktober 2010	8937	04 Desember 2010	9005,2
16 September 2010	8988	26 Oktober 2010	8908	05 Desember 2010	9016,2
17 September 2010	8978	27 Oktober 2010	8910	06 Desember 2010	9016,2
18 September 2010	8982	28 Oktober 2010	8935	07 Desember 2010	9012,4
19 September 2010	8982	29 Oktober 2010	8933	08 Desember 2010	8995
20 September 2010	8982	30 Oktober 2010	8927	09 Desember 2010	9026

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
10 Desember 2010	9023,8	19 Januari 2011	9062,2	28 Februari 2011	8842
11 Desember 2010	9016,3	20 Januari 2011	9043,7	01 Maret 2011	8822,7
12 Desember 2010	9018	21 Januari 2011	9069,5	02 Maret 2011	8813,5
13 Desember 2010	9018	22 Januari 2011	9060,9	03 Maret 2011	8803
14 Desember 2010	9023	23 Januari 2011	9062	04 Maret 2011	8797,2
15 Desember 2010	9010	24 Januari 2011	9062	05 Maret 2011	8776,4
16 Desember 2010	9036	25 Januari 2011	9063,2	06 Maret 2011	8770,1
17 Desember 2010	9031,8	26 Januari 2011	9036,8	07 Maret 2011	8799,6
18 Desember 2010	9043	27 Januari 2011	9038	08 Maret 2011	8786
19 Desember 2010	9030	28 Januari 2011	9026,6	09 Maret 2011	8790
20 Desember 2010	9030	29 Januari 2011	9028,9	10 Maret 2011	8758,3
21 Desember 2010	9043	30 Januari 2011	9033,6	11 Maret 2011	8788,2
22 Desember 2010	9048,6	31 Januari 2011	9033,6	12 Maret 2011	8771
23 Desember 2010	9040,7	01 Februari 2011	9048	13 Maret 2011	8771
24 Desember 2010	9061,1	02 Februari 2011	9034	14 Maret 2011	8753,8
25 Desember 2010	9042,5	03 Februari 2011	9005,5	15 Maret 2011	8762,9
26 Desember 2010	9039,6	04 Februari 2011	9039,2	16 Maret 2011	8791,4
27 Desember 2010	9039,6	05 Februari 2011	8997,2	17 Maret 2011	8816
28 Desember 2010	9047,6	06 Februari 2011	9010	18 Maret 2011	8775
29 Desember 2010	9017,4	07 Februari 2011	8978,1	19 Maret 2011	8765
30 Desember 2010	9002,1	08 Februari 2011	8955,3	20 Maret 2011	8765
31 Desember 2010	8992	09 Februari 2011	8917,9	21 Maret 2011	8765
01 Januari 2011	8992	10 Februari 2011	8917	22 Maret 2011	8711,1
02 Januari 2011	8990,5	11 Februari 2011	8927	23 Maret 2011	8715,1
03 Januari 2011	9001,5	12 Februari 2011	8932,5	24 Maret 2011	8722,2
04 Januari 2011	8985,4	13 Februari 2011	8936,6	25 Maret 2011	8717
05 Januari 2011	8985,7	14 Februari 2011	8936,6	26 Maret 2011	8717
06 Januari 2011	8972,7	15 Februari 2011	8911,4	27 Maret 2011	8717
07 Januari 2011	8991	16 Februari 2011	8905,5	28 Maret 2011	8717
08 Januari 2011	9026,8	17 Februari 2011	8894,7	29 Maret 2011	8705
09 Januari 2011	9015,3	18 Februari 2011	8899,3	30 Maret 2011	8711
10 Januari 2011	9015,6	19 Februari 2011	8861,9	31 Maret 2011	8715
11 Januari 2011	9029,8	20 Februari 2011	8839,2	01 April 2011	8703
12 Januari 2011	9043,9	21 Februari 2011	8839,2	02 April 2011	8685
13 Januari 2011	9032,3	22 Februari 2011	8856,2	03 April 2011	8675
14 Januari 2011	9050,1	23 Februari 2011	8871,9	04 April 2011	8699
15 Januari 2011	9051,3	24 Februari 2011	8858,2	05 April 2011	8668
16 Januari 2011	9085	25 Februari 2011	8883,5	06 April 2011	8665
17 Januari 2011	9074	26 Februari 2011	8842,7	07 April 2011	8653
18 Januari 2011	9059,4	27 Februari 2011	8842,7	08 April 2011	8665

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
09 April 2011	8641	19 Mei 2011	8558	28 Juni 2011	8605,1
10 April 2011	8641	20 Mei 2011	8557	29 Juni 2011	8627
11 April 2011	8641	21 Mei 2011	8562	30 Juni 2011	8605
12 April 2011	8655	22 Mei 2011	8537,6	01 Juli 2011	8577,1
13 April 2011	8660	23 Mei 2011	8537,6	02 Juli 2011	8545
14 April 2011	8655	24 Mei 2011	8605	03 Juli 2011	8530,8
15 April 2011	8656	25 Mei 2011	8569	04 Juli 2011	8530,8
16 April 2011	8668	26 Mei 2011	8593,8	05 Juli 2011	8523,1
17 April 2011	8665	27 Mei 2011	8580	06 Juli 2011	8535
18 April 2011	8663	28 Mei 2011	8562	07 Juli 2011	8540,5
19 April 2011	8675	29 Mei 2011	8547,9	08 Juli 2011	8533,7
20 April 2011	8675	30 Mei 2011	8547,9	09 Juli 2011	8512,5
21 April 2011	8655	31 Mei 2011	8543,7	10 Juli 2011	8512,5
22 April 2011	8628	01 Juni 2011	8541,4	11 Juli 2011	8520,5
23 April 2011	8628	02 Juni 2011	8545	12 Juli 2011	8525,8
24 April 2011	8628	03 Juni 2011	8545,3	13 Juli 2011	8546
25 April 2011	8628	04 Juni 2011	8532	14 Juli 2011	8555
26 April 2011	8628	05 Juni 2011	8533	15 Juli 2011	8540,9
27 April 2011	8653	06 Juni 2011	8518,5	16 Juli 2011	8532,3
28 April 2011	8635	07 Juni 2011	8507	17 Juli 2011	8540,2
29 April 2011	8580	08 Juni 2011	8517,7	18 Juli 2011	8540,2
30 April 2011	8568	09 Juni 2011	8516	19 Juli 2011	8555
01 Mei 2011	8568	10 Juni 2011	8523	20 Juli 2011	8547,5
02 Mei 2011	8568	11 Juni 2011	8534,1	21 Juli 2011	8532,4
03 Mei 2011	8548	12 Juni 2011	8521	22 Juli 2011	8543
04 Mei 2011	8548	13 Juni 2011	8521	23 Juli 2011	8530
05 Mei 2011	8565	14 Juni 2011	8570	24 Juli 2011	8530
06 Mei 2011	8563	15 Juni 2011	8535,4	25 Juli 2011	8528
07 Mei 2011	8590	16 Juni 2011	8548	26 Juli 2011	8523
08 Mei 2011	8580	17 Juni 2011	8615	27 Juli 2011	8505
09 Mei 2011	8580	18 Juni 2011	8585	28 Juli 2011	8485
10 Mei 2011	8570	19 Juni 2011	8580	29 Juli 2011	8502,1
11 Mei 2011	8560	20 Juni 2011	8580	30 Juli 2011	8505
12 Mei 2011	8540	21 Juni 2011	8616,7	31 Juli 2011	8502,3
13 Mei 2011	8570	22 Juni 2011	8608	01 Agustus 2011	8502,3
14 Mei 2011	8590	23 Juni 2011	8585	02 Agustus 2011	8460
15 Mei 2011	8590	24 Juni 2011	8607,9	03 Agustus 2011	8434,5
16 Mei 2011	8555	25 Juni 2011	8600	04 Agustus 2011	8485,9
17 Mei 2011	8573	26 Juni 2011	8600	05 Agustus 2011	8498,9
18 Mei 2011	8575	27 Juni 2011	8600	06 Agustus 2011	8515,3

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
07 Agustus 2011	8519,2	16 September 2011	8815	26 Oktober 2011	8851,3
08 Agustus 2011	8519,2	17 September 2011	8805,5	27 Oktober 2011	8873,5
09 Agustus 2011	8504,5	18 September 2011	8805,5	28 Oktober 2011	8858,1
10 Agustus 2011	8544	19 September 2011	8805,5	29 Oktober 2011	8813,1
11 Agustus 2011	8524,5	20 September 2011	8849,9	30 Oktober 2011	8813,1
12 Agustus 2011	8535,6	21 September 2011	8840,1	31 Oktober 2011	8785,9
13 Agustus 2011	8540,9	22 September 2011	8868	01 November 2011	8852,5
14 Agustus 2011	8548,6	23 September 2011	8762,8	02 November 2011	8893,9
15 Agustus 2011	8548,6	24 September 2011	8780,5	03 November 2011	8928
16 Agustus 2011	8548	25 September 2011	8796,6	04 November 2011	8969,4
17 Agustus 2011	8530	26 September 2011	8796,6	05 November 2011	8956
18 Agustus 2011	8533,5	27 September 2011	9082,8	06 November 2011	8950
19 Agustus 2011	8535	28 September 2011	8903,9	07 November 2011	8950
20 Agustus 2011	8545	29 September 2011	8895,4	08 November 2011	8965
21 Agustus 2011	8555	30 September 2011	8961,8	09 November 2011	8921,7
22 Agustus 2011	8555	01 Oktober 2011	9029,5	10 November 2011	8889,3
23 Agustus 2011	8545,2	02 Oktober 2011	9029,5	11 November 2011	8969,5
24 Agustus 2011	8544	03 Oktober 2011	9029,5	12 November 2011	8962,1
25 Agustus 2011	8558	04 Oktober 2011	8980	13 November 2011	8962,1
26 Agustus 2011	8605	05 Oktober 2011	8905,2	14 November 2011	8971,9
27 Agustus 2011	8549	06 Oktober 2011	8962	15 November 2011	8968,5
28 Agustus 2011	8517,3	07 Oktober 2011	8905,1	16 November 2011	8994,5
29 Agustus 2011	8517,3	08 Oktober 2011	4497	17 November 2011	8993
30 Agustus 2011	8530	09 Oktober 2011	4497	18 November 2011	9000,4
31 Agustus 2011	8525	10 Oktober 2011	8889,5	19 November 2011	9019,2
01 September 2011	8509,7	11 Oktober 2011	8891,7	20 November 2011	9025
02 September 2011	8533	12 Oktober 2011	8904,3	21 November 2011	9025
03 September 2011	8510	13 Oktober 2011	8861,2	22 November 2011	9024,6
04 September 2011	8527,2	14 Oktober 2011	8892,5	23 November 2011	9045
05 September 2011	8527,2	15 Oktober 2011	8852,8	24 November 2011	9045
06 September 2011	8542,3	16 Oktober 2011	8844,6	25 November 2011	8990,5
07 September 2011	8557	17 Oktober 2011	8844,6	26 November 2011	9054,9
08 September 2011	8552,4	18 Oktober 2011	8816	27 November 2011	9055
09 September 2011	8562,3	19 Oktober 2011	8873,1	28 November 2011	9055
10 September 2011	8572,5	20 Oktober 2011	8848,1	29 November 2011	9130
11 September 2011	8571,6	21 Oktober 2011	8881,2	30 November 2011	9165
12 September 2011	8574	22 Oktober 2011	8857,5	01 Desember 2011	9082,3
13 September 2011	8599,8	23 Oktober 2011	8860,8	02 Desember 2011	9000
14 September 2011	8638,3	24 Oktober 2011	8860,8	03 Desember 2011	9033
15 September 2011	8790	25 Oktober 2011	8840,2	04 Desember 2011	9078,8

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
05 Desember 2011	9019,4	14 Januari 2012	9085,3	23 Februari 2012	9069,6
06 Desember 2011	9022,2	15 Januari 2012	9082,5	24 Februari 2012	9045
07 Desember 2011	9043	16 Januari 2012	9082,5	25 Februari 2012	9044
08 Desember 2011	9033,5	17 Januari 2012	9129,5	26 Februari 2012	9048,8
09 Desember 2011	8995,1	18 Januari 2012	9080,5	27 Februari 2012	9048,8
10 Desember 2011	9065,6	19 Januari 2012	9030	28 Februari 2012	9145,6
11 Desember 2011	9073	20 Januari 2012	8976,5	29 Februari 2012	9030
12 Desember 2011	9073	21 Januari 2012	8945	01 Maret 2012	9019
13 Desember 2011	9077,5	22 Januari 2012	8930,7	02 Maret 2012	9085
14 Desember 2011	9117,4	23 Januari 2012	8945	03 Maret 2012	9120
15 Desember 2011	9210	24 Januari 2012	8926,5	04 Maret 2012	9080
16 Desember 2011	9090,1	25 Januari 2012	8963,8	05 Maret 2012	9080
17 Desember 2011	9080	26 Januari 2012	9056,4	06 Maret 2012	9141
18 Desember 2011	9080	27 Januari 2012	8979,6	07 Maret 2012	9135
19 Desember 2011	9045,3	28 Januari 2012	8983	08 Maret 2012	9166,6
20 Desember 2011	9120,1	29 Januari 2012	8977	09 Maret 2012	9138
21 Desember 2011	9076,3	30 Januari 2012	8977	10 Maret 2012	9119,5
22 Desember 2011	9072,5	31 Januari 2012	9029	11 Maret 2012	9130
23 Desember 2011	9142,5	01 Februari 2012	8990	12 Maret 2012	9132,5
24 Desember 2011	9053	02 Februari 2012	8999	13 Maret 2012	9173,5
25 Desember 2011	9067,2	03 Februari 2012	8938,7	14 Maret 2012	9167,2
26 Desember 2011	9067,2	04 Februari 2012	8962,7	15 Maret 2012	9160
27 Desember 2011	9067,2	05 Februari 2012	8956,8	16 Maret 2012	9180
28 Desember 2011	9061,2	06 Februari 2012	8956,8	17 Maret 2012	9121,5
29 Desember 2011	9073,4	07 Februari 2012	8985	18 Maret 2012	9114,6
30 Desember 2011	9072,8	08 Februari 2012	8978,9	19 Maret 2012	9130
31 Desember 2011	9143	09 Februari 2012	8928,6	20 Maret 2012	9182
01 Januari 2012	9080	10 Februari 2012	9020	21 Maret 2012	9185
02 Januari 2012	9080	11 Februari 2012	9071	22 Maret 2012	9199
03 Januari 2012	9081,5	12 Februari 2012	8990	23 Maret 2012	9226,5
04 Januari 2012	9122	13 Februari 2012	8990	24 Maret 2012	9209
05 Januari 2012	9145	14 Februari 2012	9052	25 Maret 2012	9128,9
06 Januari 2012	9120,6	15 Februari 2012	9005,3	26 Maret 2012	9129,6
07 Januari 2012	9095	16 Februari 2012	9059	27 Maret 2012	9184
08 Januari 2012	9087,8	17 Februari 2012	9005,1	28 Maret 2012	9170,1
09 Januari 2012	9087,8	18 Februari 2012	9050	29 Maret 2012	9139,3
10 Januari 2012	9153,1	19 Februari 2012	9011,7	30 Maret 2012	9143,5
11 Januari 2012	9150	20 Februari 2012	9011,7	31 Maret 2012	9176
12 Januari 2012	9197,5	21 Februari 2012	9000,5	01 April 2012	9140,1
13 Januari 2012	9159	22 Februari 2012	9049	02 April 2012	9140,1

Tanggal	Nilai Tukar (IDR)
03 April 2012	9155,1
04 April 2012	9140,1
05 April 2012	9167,5
06 April 2012	9165
07 April 2012	9207,8
08 April 2012	9113,5
09 April 2012	9110
10 April 2012	9165
11 April 2012	9199,5
12 April 2012	9190
13 April 2012	9162,9
14 April 2012	9187
15 April 2012	9202,2
16 April 2012	9198,5
17 April 2012	9171,5
18 April 2012	9185
19 April 2012	9172,6
20 April 2012	9182
21 April 2012	9187,5
22 April 2012	9191,95
23 April 2012	9196,4
24 April 2012	9200,85
25 April 2012	9205,3
26 April 2012	9189,4
27 April 2012	9190
28 April 2012	9189
29 April 2012	9189
30 April 2012	9177
01 Mei 2012	9191,2
02 Mei 2012	9187
03 Mei 2012	9220
04 Mei 2012	9264,2
05 Mei 2012	9243,6
06 Mei 2012	9225,1
07 Mei 2012	9223,4
08 Mei 2012	9230
09 Mei 2012	9195,2
10 Mei 2012	9270
11 Mei 2012	9261,2
12 Mei 2012	9245,2

Tanggal	Nilai Tukar (IDR)
13 Mei 2012	9208,6
14 Mei 2012	9208
15 Mei 2012	9250
16 Mei 2012	9290,8
17 Mei 2012	9279
18 Mei 2012	9242
19 Mei 2012	9350
20 Mei 2012	9175
21 Mei 2012	9175
22 Mei 2012	9365,4
23 Mei 2012	9333,8
24 Mei 2012	9405
25 Mei 2012	9445
26 Mei 2012	9456,2
27 Mei 2012	9290
28 Mei 2012	9290
29 Mei 2012	9333
30 Mei 2012	9481
31 Mei 2012	9455
01 Juni 2012	9402,2
02 Juni 2012	9386,6
03 Juni 2012	9400
04 Juni 2012	9306
05 Juni 2012	9400
06 Juni 2012	9415
07 Juni 2012	9301,3
08 Juni 2012	9425
09 Juni 2012	9470
10 Juni 2012	9340,4
11 Juni 2012	9341,2
12 Juni 2012	9465
13 Juni 2012	9435
14 Juni 2012	9421,4
15 Juni 2012	9445
16 Juni 2012	9395
17 Juni 2012	9475
18 Juni 2012	9475
19 Juni 2012	9387,3
20 Juni 2012	9405,3
21 Juni 2012	9420,3

Tanggal	Nilai Tukar (IDR)
22 Juni 2012	9428,8
23 Juni 2012	9436,8
24 Juni 2012	9506
25 Juni 2012	9431,5
26 Juni 2012	9439,1
27 Juni 2012	9440
28 Juni 2012	9446,6
29 Juni 2012	9444,5
30 Juni 2012	9391,2
01 Juli 2012	9393
02 Juli 2012	9393
03 Juli 2012	9392
04 Juli 2012	9322,6
05 Juli 2012	9361
06 Juli 2012	9394,5
07 Juli 2012	9415,2
08 Juli 2012	9483
09 Juli 2012	9483
10 Juli 2012	9409,8
11 Juli 2012	9430
12 Juli 2012	9415,5
13 Juli 2012	9444,6
14 Juli 2012	9465
15 Juli 2012	9448
16 Juli 2012	9448
17 Juli 2012	9450,1
18 Juli 2012	9455
19 Juli 2012	9450,1
20 Juli 2012	9447,8
21 Juli 2012	9449
22 Juli 2012	9450
23 Juli 2012	9450
24 Juli 2012	9535
25 Juli 2012	9467,5
26 Juli 2012	9484,5
27 Juli 2012	9477,8
28 Juli 2012	9456,3
29 Juli 2012	9455
30 Juli 2012	9455
31 Juli 2012	9454,8



Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
01 Agustus 2012	9445,8	10 September 2012	9571	20 Oktober 2012	9592,6
02 Agustus 2012	9473	11 September 2012	9577,4	21 Oktober 2012	9592,6
03 Agustus 2012	9466,9	12 September 2012	9577,5	22 Oktober 2012	9592,6
04 Agustus 2012	9468,4	13 September 2012	9567	23 Oktober 2012	9604,9
05 Agustus 2012	9461,2	14 September 2012	9577,8	24 Oktober 2012	9607,5
06 Agustus 2012	9461,2	15 September 2012	9506	25 Oktober 2012	9628
07 Agustus 2012	9465,6	16 September 2012	9505	26 Oktober 2012	9605,3
08 Agustus 2012	9467,7	17 September 2012	9482,6	27 Oktober 2012	9605,4
09 Agustus 2012	9470	18 September 2012	9462,1	28 Oktober 2012	9605
10 Agustus 2012	9471,3	19 September 2012	9502,2	29 Oktober 2012	9605
11 Agustus 2012	9476,5	20 September 2012	9536,9	30 Oktober 2012	9606,2
12 Agustus 2012	9478	21 September 2012	9540,3	31 Oktober 2012	9614,2
13 Agustus 2012	9478	22 September 2012	9552,5	01 November 2012	9605
14 Agustus 2012	9485,3	23 September 2012	9545	02 November 2012	9617,5
15 Agustus 2012	9482,3	24 September 2012	9547,4	03 November 2012	9615
16 Agustus 2012	9496,8	25 September 2012	9561,9	04 November 2012	9615
17 Agustus 2012	9489,4	26 September 2012	9575,7	05 November 2012	9648,5
18 Agustus 2012	9490	27 September 2012	9627	06 November 2012	9619,6
19 Agustus 2012	9490	28 September 2012	9575,3	07 November 2012	9619,3
20 Agustus 2012	9510,4	29 September 2012	9569,6	08 November 2012	9622
21 Agustus 2012	9489,8	30 September 2012	9570	09 November 2012	9629,5
22 Agustus 2012	9490,1	01 Oktober 2012	9570	10 November 2012	9619,6
23 Agustus 2012	9490,5	02 Oktober 2012	9580,5	11 November 2012	9620
24 Agustus 2012	9494,8	03 Oktober 2012	9581,1	12 November 2012	9624,5
25 Agustus 2012	9505,8	04 Oktober 2012	9583,8	13 November 2012	9619
26 Agustus 2012	9505	05 Oktober 2012	9590,1	14 November 2012	9624
27 Agustus 2012	9505	06 Oktober 2012	9589,5	15 November 2012	9625,3
28 Agustus 2012	9515,3	07 Oktober 2012	9589	16 November 2012	9620
29 Agustus 2012	9534,8	08 Oktober 2012	9583,5	17 November 2012	9635,4
30 Agustus 2012	9547,3	09 Oktober 2012	9590,7	18 November 2012	9625
31 Agustus 2012	9544,3	10 Oktober 2012	9592	19 November 2012	9625
01 September 2012	9534,1	11 Oktober 2012	9592,5	20 November 2012	9628,6
02 September 2012	9535	12 Oktober 2012	9583	21 November 2012	9632,8
03 September 2012	9535	13 Oktober 2012	9576,8	22 November 2012	9635
04 September 2012	9535,5	14 Oktober 2012	9578	23 November 2012	9627,8
05 September 2012	9570,9	15 Oktober 2012	9578	24 November 2012	9614
06 September 2012	9569,9	16 Oktober 2012	9589,5	25 November 2012	9556,2
07 September 2012	9565,5	17 Oktober 2012	9583,5	26 November 2012	9556,2
08 September 2012	9569,3	18 Oktober 2012	9573,7	27 November 2012	9599
09 September 2012	9571	19 Oktober 2012	9618,7	28 November 2012	9600,35

Tanggal	Nilai Tukar (IDR)
29 November 2012	9600,5
30 November 2012	9623,6
01 Desember 2012	9603
02 Desember 2012	9594
03 Desember 2012	9594
04 Desember 2012	9595,35
05 Desember 2012	9599,75
06 Desember 2012	9620,45
07 Desember 2012	9649,7
08 Desember 2012	9621,15
09 Desember 2012	9620
10 Desember 2012	9620
11 Desember 2012	9630,6
12 Desember 2012	9635,5
13 Desember 2012	9629,95
14 Desember 2012	9635,45
15 Desember 2012	9638,25
16 Desember 2012	9638
17 Desember 2012	9637,25
18 Desember 2012	9639,75
19 Desember 2012	9644
20 Desember 2012	9632,65
21 Desember 2012	9652,35
22 Desember 2012	9657,35
23 Desember 2012	9612,5
24 Desember 2012	9612,5
25 Desember 2012	9657,25
26 Desember 2012	9657,65
27 Desember 2012	9658,25
28 Desember 2012	9661,95
29 Desember 2012	9638
30 Desember 2012	9641
31 Desember 2012	9641
01 Januari 2013	9664,95
02 Januari 2013	9652
03 Januari 2013	9650
04 Januari 2013	9657,5
05 Januari 2013	9660,7
06 Januari 2013	9660
07 Januari 2013	9785

Tanggal	Nilai Tukar (IDR)
08 Januari 2013	9795
09 Januari 2013	9669,2
10 Januari 2013	9660
11 Januari 2013	9659,9
12 Januari 2013	9635
13 Januari 2013	9632,5
14 Januari 2013	9632,5
15 Januari 2013	9635,5
16 Januari 2013	9651,05
17 Januari 2013	9656,1
18 Januari 2013	9649,25
19 Januari 2013	9645
20 Januari 2013	9627,95
21 Januari 2013	9854,5
22 Januari 2013	9618,05
23 Januari 2013	9619,75
24 Januari 2013	9624,6
25 Januari 2013	9625,5
26 Januari 2013	9652,5
27 Januari 2013	9651,25
28 Januari 2013	9753,25
29 Januari 2013	9676
30 Januari 2013	9677,55
31 Januari 2013	9687,95
01 Februari 2013	9700
02 Februari 2013	9728,45
03 Februari 2013	9721
04 Februari 2013	9713
05 Februari 2013	9670,25
06 Februari 2013	9675
07 Februari 2013	9712,5
08 Februari 2013	9690
09 Februari 2013	9666,9
10 Februari 2013	9666
11 Februari 2013	9666
12 Februari 2013	9618,4
13 Februari 2013	9646,45
14 Februari 2013	9632,15
15 Februari 2013	9660
16 Februari 2013	9669,6

Tanggal	Nilai Tukar (IDR)
17 Februari 2013	9666,25
18 Februari 2013	9670
19 Februari 2013	9680,6
20 Februari 2013	9691,5
21 Februari 2013	9682,75
22 Februari 2013	9712,5
23 Februari 2013	9708,7
24 Februari 2013	9709
25 Februari 2013	9709
26 Februari 2013	9703,5
27 Februari 2013	9707,4
28 Februari 2013	9686,5
01 Maret 2013	9663,6
02 Maret 2013	9681
03 Maret 2013	9678
04 Maret 2013	9678
05 Maret 2013	9720
06 Maret 2013	9692,3
07 Maret 2013	9698
08 Maret 2013	9684,3
09 Maret 2013	9680
10 Maret 2013	9679,75
11 Maret 2013	9678,95
12 Maret 2013	9696,45
13 Maret 2013	9696,3
14 Maret 2013	9690,75
15 Maret 2013	9705,2
16 Maret 2013	9721
17 Maret 2013	9704,5
18 Maret 2013	9704,5
19 Maret 2013	9711,3
20 Maret 2013	9711,15
21 Maret 2013	9720,4
22 Maret 2013	9755
23 Maret 2013	9742,6
24 Maret 2013	9743
25 Maret 2013	9743
26 Maret 2013	9735
27 Maret 2013	9732,25
28 Maret 2013	9724,45

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
29 Maret 2013	9718,25	08 Mei 2013	9734,05	17 Juni 2013	9923,5
30 Maret 2013	9612,5	09 Mei 2013	9729,25	18 Juni 2013	9890
31 Maret 2013	9718	10 Mei 2013	9729,45	19 Juni 2013	9907,1
01 April 2013	9718	11 Mei 2013	9733,8	20 Juni 2013	9907,35
02 April 2013	9720,25	12 Mei 2013	9734	21 Juni 2013	9924,05
03 April 2013	9736,5	13 Mei 2013	9734	22 Juni 2013	9924,35
04 April 2013	9747,6	14 Mei 2013	9737,5	23 Juni 2013	9927,25
05 April 2013	9749,15	15 Mei 2013	9739,1	24 Juni 2013	9927,25
06 April 2013	9747,1	16 Mei 2013	9747,75	25 Juni 2013	9929,3
07 April 2013	9809,35	17 Mei 2013	9749,7	26 Juni 2013	9920,65
08 April 2013	9757,25	18 Mei 2013	9754,6	27 Juni 2013	9924,8
09 April 2013	9748,25	19 Mei 2013	9756	28 Juni 2013	9925,25
10 April 2013	9721,75	20 Mei 2013	9756	29 Juni 2013	9924,35
11 April 2013	9689,45	21 Mei 2013	9755,6	30 Juni 2013	9925
12 April 2013	9702,05	22 Mei 2013	9763,35	01 Juli 2013	9925
13 April 2013	9711,75	23 Mei 2013	9765,25	02 Juli 2013	9924,25
14 April 2013	9713,6	24 Mei 2013	9770,4	03 Juli 2013	9932,8
15 April 2013	9712,1	25 Mei 2013	9772,1	04 Juli 2013	9939,8
16 April 2013	9712,25	26 Mei 2013	9773	05 Juli 2013	9940,25
17 April 2013	9716,5	27 Mei 2013	9773	06 Juli 2013	9942,9
18 April 2013	9710,35	28 Mei 2013	9789,2	07 Juli 2013	9943
19 April 2013	9714,25	29 Mei 2013	9792,25	08 Juli 2013	9943
20 April 2013	9712,6	30 Mei 2013	9801,75	09 Juli 2013	9955,25
21 April 2013	9715,35	31 Mei 2013	9804,6	10 Juli 2013	9958,8
22 April 2013	9713	01 Juni 2013	9794,8	11 Juli 2013	9970,45
23 April 2013	9717,7	02 Juni 2013	9795	12 Juli 2013	9970,05
24 April 2013	9724,1	03 Juni 2013	9935	13 Juli 2013	9992,55
25 April 2013	9717,75	04 Juni 2013	9805,55	14 Juli 2013	10101,9
26 April 2013	9720	05 Juni 2013	9800,4	15 Juli 2013	9995,15
27 April 2013	9720	06 Juni 2013	9791,95	16 Juli 2013	10017,5
28 April 2013	9721	07 Juni 2013	9791,75	17 Juli 2013	10329,5
29 April 2013	9721	08 Juni 2013	9805,45	18 Juli 2013	10045,5
30 April 2013	9720,55	09 Juni 2013	9994,75	19 Juli 2013	10060
01 Mei 2013	9722,3	10 Juni 2013	9805	20 Juli 2013	10080
02 Mei 2013	9722,9	11 Juni 2013	9800,55	21 Juli 2013	10078
03 Mei 2013	9735,9	12 Juni 2013	9825,05	22 Juli 2013	10078
04 Mei 2013	9734,8	13 Juni 2013	9859,75	23 Juli 2013	10065
05 Mei 2013	9735	14 Juni 2013	9883,5	24 Juli 2013	10200
06 Mei 2013	9735	15 Juni 2013	9872,25	25 Juli 2013	10259,5
07 Mei 2013	9734,1	16 Juni 2013	9923,5	26 Juli 2013	10304,5

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
27 Juli 2013	10265	05 September 2013	11124,5	15 Oktober 2013	11364,5
28 Juli 2013	10091	06 September 2013	11124	16 Oktober 2013	11364,5
29 Juli 2013	10265	07 September 2013	11561,5	17 Oktober 2013	10945
30 Juli 2013	10272,5	08 September 2013	11175	18 Oktober 2013	11329,5
31 Juli 2013	10322	09 September 2013	11175	19 Oktober 2013	11325
01 Agustus 2013	10285,5	10 September 2013	11159	20 Oktober 2013	11323
02 Agustus 2013	10263,5	11 September 2013	11234,5	21 Oktober 2013	11323
03 Agustus 2013	10288,5	12 September 2013	11343,5	22 Oktober 2013	11335,5
04 Agustus 2013	10285	13 September 2013	11070,5	23 Oktober 2013	11291,5
05 Agustus 2013	10202	14 September 2013	11410,5	24 Oktober 2013	11269,5
06 Agustus 2013	10284,5	15 September 2013	11409,5	25 Oktober 2013	11156
07 Agustus 2013	10280	16 September 2013	11409,5	26 Oktober 2013	11014
08 Agustus 2013	10285	17 September 2013	11377	27 Oktober 2013	11015
09 Agustus 2013	10284	18 September 2013	11447	28 Oktober 2013	10735,5
10 Agustus 2013	10280	19 September 2013	11474	29 Oktober 2013	11054
11 Agustus 2013	10285	20 September 2013	11282,5	30 Oktober 2013	11104,5
12 Agustus 2013	10285	21 September 2013	11349,5	31 Oktober 2013	10843
13 Agustus 2013	10290	22 September 2013	11350	01 November 2013	10950
14 Agustus 2013	10289,5	23 September 2013	11350	02 November 2013	11332
15 Agustus 2013	10290	24 September 2013	11444,5	03 November 2013	11333
16 Agustus 2013	10355	25 September 2013	11469,5	04 November 2013	11333
17 Agustus 2013	10385,5	26 September 2013	11484,5	05 November 2013	11160
18 Agustus 2013	10385	27 September 2013	11457,5	06 November 2013	11352,5
19 Agustus 2013	10540,5	28 September 2013	11537,5	07 November 2013	11410
20 Agustus 2013	10490	29 September 2013	11536,5	08 November 2013	11270
21 Agustus 2013	10808,5	30 September 2013	11536,5	09 November 2013	11428,5
22 Agustus 2013	10775	01 Oktober 2013	11579,5	10 November 2013	11410
23 Agustus 2013	10818,5	02 Oktober 2013	11505,5	11 November 2013	11407
24 Agustus 2013	10780	03 Oktober 2013	11531	12 November 2013	11560,5
25 Agustus 2013	10778	04 Oktober 2013	11527,5	13 November 2013	11420
26 Agustus 2013	10974	05 Oktober 2013	11522,5	14 November 2013	11605
27 Agustus 2013	11040,5	06 Oktober 2013	11246	15 November 2013	11385
28 Agustus 2013	11304,5	07 Oktober 2013	11523	16 November 2013	11621
29 Agustus 2013	11222,5	08 Oktober 2013	11523,5	17 November 2013	11620
30 Agustus 2013	11130,5	09 Oktober 2013	11515,5	18 November 2013	11620
31 Agustus 2013	11278,5	10 Oktober 2013	11531,5	19 November 2013	11633
01 September 2013	10915,5	11 Oktober 2013	11490	20 November 2013	11596,5
02 September 2013	10920	12 Oktober 2013	11365	21 November 2013	11656
03 September 2013	10977,5	13 Oktober 2013	11365	22 November 2013	11525
04 September 2013	11065,5	14 Oktober 2013	9612,5	23 November 2013	11491

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
24 November 2013	11694,5	03 Januari 2014	12102,5	12 Februari 2014	11940,5
25 November 2013	11694,5	04 Januari 2014	12115,5	13 Februari 2014	11913
26 November 2013	11747	05 Januari 2014	12110,5	14 Februari 2014	11865
27 November 2013	11765	06 Januari 2014	12110,5	15 Februari 2014	11607,5
28 November 2013	11765	07 Januari 2014	12151	16 Februari 2014	11615,5
29 November 2013	11843,5	08 Januari 2014	12056	17 Februari 2014	11615,5
30 November 2013	11890	09 Januari 2014	12042,5	18 Februari 2014	11647,5
01 Desember 2013	11794,5	10 Januari 2014	12017,5	19 Februari 2014	11760,5
02 Desember 2013	11794,5	11 Januari 2014	11886	20 Februari 2014	11720
03 Desember 2013	11896	12 Januari 2014	11825,5	21 Februari 2014	11735,5
04 Desember 2013	11741	13 Januari 2014	11825,5	22 Februari 2014	11712,5
05 Desember 2013	11890	14 Januari 2014	11787	23 Februari 2014	11690,5
06 Desember 2013	11930	15 Januari 2014	11810,5	24 Februari 2014	11690,5
07 Desember 2013	11956	16 Januari 2014	11847	25 Februari 2014	11650
08 Desember 2013	11846	17 Januari 2014	11900,5	26 Februari 2014	11615,5
09 Desember 2013	11846	18 Januari 2014	11860	27 Februari 2014	11647,5
10 Desember 2013	11825	19 Januari 2014	11860,5	28 Februari 2014	11615
11 Desember 2013	11850,5	20 Januari 2014	11860,5	01 Maret 2014	11610
12 Desember 2013	11948	21 Januari 2014	11905,5	02 Maret 2014	11578,5
13 Desember 2013	12035,5	22 Januari 2014	12037,5	03 Maret 2014	11578,5
14 Desember 2013	11973	23 Januari 2014	12060	04 Maret 2014	11655,5
15 Desember 2013	12020,5	24 Januari 2014	12075,5	05 Maret 2014	11580,5
16 Desember 2013	12020,5	25 Januari 2014	12175,5	06 Maret 2014	11590,5
17 Desember 2013	11965	26 Januari 2014	12129,5	07 Maret 2014	11383
18 Desember 2013	12001	27 Januari 2014	12129,5	08 Maret 2014	11440
19 Desember 2013	12042,5	28 Januari 2014	12195,5	09 Maret 2014	11395,5
20 Desember 2013	12115,5	29 Januari 2014	12160	10 Maret 2014	11395,5
21 Desember 2013	12110	30 Januari 2014	12169	11 Maret 2014	11463
22 Desember 2013	12155,5	31 Januari 2014	12165,5	12 Maret 2014	11478
23 Desember 2013	12155,5	01 Februari 2014	12147,5	13 Maret 2014	11458
24 Desember 2013	12125,5	02 Februari 2014	12165	14 Maret 2014	11435,5
25 Desember 2013	12200	03 Februari 2014	12165	15 Maret 2014	11305,5
26 Desember 2013	12150,5	04 Februari 2014	12095,5	16 Maret 2014	11305,5
27 Desember 2013	12145,5	05 Februari 2014	12010,5	17 Maret 2014	11305,5
28 Desember 2013	12159,5	06 Februari 2014	12030,5	18 Maret 2014	11257,5
29 Desember 2013	12144,5	07 Februari 2014	11970	19 Maret 2014	11300,5
30 Desember 2013	12144,5	08 Februari 2014	11990	20 Maret 2014	11377,5
31 Desember 2013	12210	09 Februari 2014	12010,5	21 Maret 2014	11465
01 Januari 2014	12170	10 Februari 2014	12010,5	22 Maret 2014	11385
02 Januari 2014	12160,5	11 Februari 2014	12015	23 Maret 2014	11410,5

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
24 Maret 2014	11410,5	03 Mei 2014	11530	12 Juni 2014	11815
25 Maret 2014	11379	04 Mei 2014	11525	13 Juni 2014	11767
26 Maret 2014	11405	05 Mei 2014	11525	14 Juni 2014	11810
27 Maret 2014	11425,5	06 Mei 2014	11510	15 Juni 2014	11795
28 Maret 2014	11392,5	07 Mei 2014	11505	16 Juni 2014	11795
29 Maret 2014	11325	08 Mei 2014	11545	17 Juni 2014	11837
30 Maret 2014	11314,5	09 Mei 2014	11520	18 Juni 2014	11935
31 Maret 2014	11314,5	10 Mei 2014	11532	19 Juni 2014	11957,5
01 April 2014	11253	11 Mei 2014	11550	20 Juni 2014	11947
02 April 2014	11253	12 Mei 2014	11550	21 Juni 2014	11955
03 April 2014	11335	13 Mei 2014	11515	22 Juni 2014	11960
04 April 2014	11338	14 Mei 2014	11465	23 Juni 2014	11960
05 April 2014	11288	15 Mei 2014	11366	24 Juni 2014	11975
06 April 2014	11320	16 Mei 2014	11432	25 Juni 2014	12007
07 April 2014	11320	17 Mei 2014	11340	26 Juni 2014	12107
08 April 2014	11328	18 Mei 2014	11337,5	27 Juni 2014	12135
09 April 2014	11230	19 Mei 2014	11337,5	28 Juni 2014	12028
10 April 2014	11358	20 Mei 2014	11403	29 Juni 2014	12030
11 April 2014	11475	21 Mei 2014	11493	30 Juni 2014	12030
12 April 2014	11460	22 Mei 2014	11503	01 Juli 2014	11870
13 April 2014	11405	23 Mei 2014	11525	02 Juli 2014	11875
14 April 2014	11405	24 Mei 2014	11603	03 Juli 2014	11930
15 April 2014	11411	25 Mei 2014	11605	04 Juli 2014	11920
16 April 2014	11485	26 Mei 2014	11605	05 Juli 2014	11820
17 April 2014	11405	27 Mei 2014	11560	06 Juli 2014	11870
18 April 2014	11423	28 Mei 2014	11610	07 Juli 2014	11870
19 April 2014	11410	29 Mei 2014	11638	08 Juli 2014	11815
20 April 2014	11455	30 Mei 2014	11606	09 Juli 2014	11682
21 April 2014	11455	31 Mei 2014	11743	10 Juli 2014	11655
22 April 2014	11438	01 Juni 2014	11675	11 Juli 2014	11590
23 April 2014	11548	02 Juni 2014	11675	12 Juli 2014	11635
24 April 2014	11618	03 Juni 2014	11827	13 Juli 2014	11625
25 April 2014	11628	04 Juni 2014	11823	14 Juli 2014	11625
26 April 2014	11575	05 Juni 2014	11890	15 Juli 2014	11680
27 April 2014	11575	06 Juni 2014	11817	16 Juli 2014	11775
28 April 2014	11575	07 Juni 2014	11755	17 Juli 2014	11680
29 April 2014	11588	08 Juni 2014	11825	18 Juli 2014	11682
30 April 2014	11523	09 Juni 2014	11825	19 Juli 2014	11615
01 Mei 2014	11538	10 Juni 2014	11795	20 Juli 2014	11625
02 Mei 2014	11523	11 Juni 2014	11775	21 Juli 2014	11625

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
22 Juli 2014	11515	31 Agustus 2014	11704	10 Oktober 2014	12200
23 Juli 2014	11482	01 September 2014	11704	11 Oktober 2014	12212
24 Juli 2014	11487	02 September 2014	11718	12 Oktober 2014	12220
25 Juli 2014	11564	03 September 2014	11765	13 Oktober 2014	12220
26 Juli 2014	11590	04 September 2014	11775	14 Oktober 2014	12210
27 Juli 2014	11571	05 September 2014	11767	15 Oktober 2014	12185
28 Juli 2014	11571	06 September 2014	11732	16 Oktober 2014	12202
29 Juli 2014	11580	07 September 2014	11732	17 Oktober 2014	12230
30 Juli 2014	11635	08 September 2014	11732	18 Oktober 2014	12072
31 Juli 2014	11705	09 September 2014	11754	19 Oktober 2014	12112
01 Agustus 2014	11860	10 September 2014	11812	20 Oktober 2014	12112
02 Agustus 2014	11871	11 September 2014	11842	21 Oktober 2014	11973
03 Agustus 2014	11871	12 September 2014	11835	22 Oktober 2014	11985
04 Agustus 2014	11871	13 September 2014	11820	23 Oktober 2014	12018
05 Agustus 2014	11750	14 September 2014	11820	24 Oktober 2014	12058
06 Agustus 2014	11747	15 September 2014	11927,5	25 Oktober 2014	12045
07 Agustus 2014	11788	16 September 2014	11965	26 Oktober 2014	12085
08 Agustus 2014	11802	17 September 2014	11965	27 Oktober 2014	12085
09 Agustus 2014	11740	18 September 2014	11985	28 Oktober 2014	12100
10 Agustus 2014	11755	19 September 2014	11982	29 Oktober 2014	12125
11 Agustus 2014	11755	20 September 2014	12002	30 Oktober 2014	12097
12 Agustus 2014	11705	21 September 2014	11993	31 Oktober 2014	12115
13 Agustus 2014	11725	22 September 2014	11993	01 November 2014	12062
14 Agustus 2014	11680	23 September 2014	12030	02 November 2014	12080
15 Agustus 2014	11697	24 September 2014	11980	03 November 2014	12080
16 Agustus 2014	11700	25 September 2014	11997,5	04 November 2014	12112
17 Agustus 2014	11677	26 September 2014	12015	05 November 2014	12108
18 Agustus 2014	11677	27 September 2014	12130	06 November 2014	12192
19 Agustus 2014	11690	28 September 2014	12060	07 November 2014	12177
20 Agustus 2014	11696	29 September 2014	12060	08 November 2014	12157
21 Agustus 2014	11720	30 September 2014	12225	09 November 2014	12185
22 Agustus 2014	11660	01 Oktober 2014	12185	10 November 2014	12185
23 Agustus 2014	11698	02 Oktober 2014	12160	11 November 2014	12177
24 Agustus 2014	11684	03 Oktober 2014	12176,5	12 November 2014	12208
25 Agustus 2014	11670	04 Oktober 2014	12176,5	13 November 2014	12185
26 Agustus 2014	11703	05 Oktober 2014	12176,5	14 November 2014	12190
27 Agustus 2014	11693	06 Oktober 2014	12210	15 November 2014	12165
28 Agustus 2014	11675	07 Oktober 2014	12202,5	16 November 2014	12210
29 Agustus 2014	11740	08 Oktober 2014	12202,5	17 November 2014	12210
30 Agustus 2014	11739	09 Oktober 2014	12265	18 November 2014	12130

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
19 November 2014	12083	29 Desember 2014	12430	07 Februari 2015	12675
20 November 2014	12140	30 Desember 2014	12415	08 Februari 2015	12673
21 November 2014	12125	31 Desember 2014	12410	09 Februari 2015	12673
22 November 2014	12103	01 Januari 2015	12435	10 Februari 2015	12682
23 November 2014	12143	02 Januari 2015	12460	11 Februari 2015	12722
24 November 2014	12143	03 Januari 2015	12485	12 Februari 2015	12855
25 November 2014	12132	04 Januari 2015	12510	13 Februari 2015	12745
26 November 2014	12139	05 Januari 2015	12510	14 Februari 2015	12747
27 November 2014	12165	06 Januari 2015	12615	15 Februari 2015	12729,5
28 November 2014	12167	07 Januari 2015	12630	16 Februari 2015	12729,5
29 November 2014	12255	08 Januari 2015	12700	17 Februari 2015	12763
30 November 2014	12185	09 Januari 2015	12670	18 Februari 2015	12817
01 Desember 2014	12185	10 Januari 2015	12567,5	19 Februari 2015	12883
02 Desember 2014	12287	11 Januari 2015	12592,5	20 Februari 2015	12878
03 Desember 2014	12290	12 Januari 2015	12592,5	21 Februari 2015	12868
04 Desember 2014	12310	13 Januari 2015	12575	22 Februari 2015	12843
05 Desember 2014	12295	14 Januari 2015	12575	23 Februari 2015	12843
06 Desember 2014	12380	15 Januari 2015	12570	24 Februari 2015	12920
07 Desember 2014	12295	16 Januari 2015	12524	25 Februari 2015	12897
08 Desember 2014	12295	17 Januari 2015	12615	26 Februari 2015	12860
09 Desember 2014	12385	18 Januari 2015	12590	27 Februari 2015	12852
10 Desember 2014	12362	19 Januari 2015	12590	28 Februari 2015	13000
11 Desember 2014	12365	20 Januari 2015	12652	01 Maret 2015	12988
12 Desember 2014	12424	21 Januari 2015	12575	02 Maret 2015	12988
13 Desember 2014	12615	22 Januari 2015	12470	03 Maret 2015	12985
14 Desember 2014	12622	23 Januari 2015	12420	04 Maret 2015	12967,5
15 Desember 2014	12622	24 Januari 2015	12468	05 Maret 2015	13115
16 Desember 2014	12880	25 Januari 2015	12473	06 Maret 2015	13007
17 Desember 2014	12880	26 Januari 2015	12473	07 Maret 2015	13030
18 Desember 2014	12685	27 Januari 2015	12488	08 Maret 2015	13087,5
19 Desember 2014	12510	28 Januari 2015	12465	09 Maret 2015	13087,5
20 Desember 2014	12405	29 Januari 2015	12500	10 Maret 2015	13073
21 Desember 2014	12450	30 Januari 2015	12618	11 Maret 2015	13212,5
22 Desember 2014	12450	31 Januari 2015	12790	12 Maret 2015	13185
23 Desember 2014	12453	01 Februari 2015	12760	13 Maret 2015	13193
24 Desember 2014	12485	02 Februari 2015	12760	14 Maret 2015	13235
25 Desember 2014	12420	03 Februari 2015	12662	15 Maret 2015	13235
26 Desember 2014	12485	04 Februari 2015	12600	16 Maret 2015	13235
27 Desember 2014	12485	05 Februari 2015	12630	17 Maret 2015	13205
28 Desember 2014	12430	06 Februari 2015	12595	18 Maret 2015	13180



Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
19 Maret 2015	13142	28 April 2015	12955	07 Juni 2015	13398
20 Maret 2015	13050	29 April 2015	12937,5	08 Juni 2015	13398
21 Maret 2015	13012	30 April 2015	12905	09 Juni 2015	13355
22 Maret 2015	13035	01 Mei 2015	12972	10 Juni 2015	13340
23 Maret 2015	13035	02 Mei 2015	13000	11 Juni 2015	13225
24 Maret 2015	12975	03 Mei 2015	13010	12 Juni 2015	13292
25 Maret 2015	12948	04 Mei 2015	13010	13 Juni 2015	13317,5
26 Maret 2015	12985	05 Mei 2015	13017	14 Juni 2015	13320
27 Maret 2015	13022,5	06 Mei 2015	13027	15 Juni 2015	13320
28 Maret 2015	13050	07 Mei 2015	13025	16 Juni 2015	13332,5
29 Maret 2015	13085	08 Mei 2015	13130	17 Juni 2015	13337,5
30 Maret 2015	13085	09 Mei 2015	13060	18 Juni 2015	13337,5
31 Maret 2015	13090	10 Mei 2015	13057	19 Juni 2015	13340
01 April 2015	13075	11 Mei 2015	13057	20 Juni 2015	13320
02 April 2015	12980	12 Mei 2015	13160	21 Juni 2015	13320
03 April 2015	13020	13 Mei 2015	13170	22 Juni 2015	13320
04 April 2015	12935	14 Mei 2015	13082	23 Juni 2015	13292
05 April 2015	12980	15 Mei 2015	13023	24 Juni 2015	13277
06 April 2015	12980	16 Mei 2015	13070	25 Juni 2015	13320
07 April 2015	12940	17 Mei 2015	13062	26 Juni 2015	13303
08 April 2015	12995	18 Mei 2015	13062	27 Juni 2015	13312,5
09 April 2015	12973	19 Mei 2015	13172	28 Juni 2015	13332
10 April 2015	12967	20 Mei 2015	13165	29 Juni 2015	13332
11 April 2015	12918	21 Mei 2015	13102,5	30 Juni 2015	13327
12 April 2015	12920	22 Mei 2015	13123	01 Juli 2015	13352
13 April 2015	12920	23 Mei 2015	13148	02 Juli 2015	13335
14 April 2015	13070	24 Mei 2015	13160	03 Juli 2015	13293
15 April 2015	12937,5	25 Mei 2015	13160	04 Juli 2015	13355
16 April 2015	12930	26 Mei 2015	13183	05 Juli 2015	13350
17 April 2015	12832	27 Mei 2015	13222,5	06 Juli 2015	13350
18 April 2015	12850	28 Mei 2015	13180	07 Juli 2015	13295
19 April 2015	12842,5	29 Mei 2015	13195	08 Juli 2015	13358
20 April 2015	12842,5	30 Mei 2015	13220	09 Juli 2015	13365
21 April 2015	13031	31 Mei 2015	13210	10 Juli 2015	13320
22 April 2015	12953	01 Juni 2015	13210	11 Juli 2015	13317,5
23 April 2015	12935	02 Juni 2015	13210	12 Juli 2015	13308
24 April 2015	12940	03 Juni 2015	13200	13 Juli 2015	13308
25 April 2015	12952	04 Juni 2015	13240	14 Juli 2015	13288
26 April 2015	12952	05 Juni 2015	13285	15 Juli 2015	13295
27 April 2015	12952	06 Juni 2015	13388	16 Juli 2015	13344,5

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
17 Juli 2015	13345	26 Agustus 2015	14010	05 Oktober 2015	14722,5
18 Juli 2015	13355	27 Agustus 2015	14105	06 Oktober 2015	14450
19 Juli 2015	13360	28 Agustus 2015	13997	07 Oktober 2015	14157
20 Juli 2015	13360	29 Agustus 2015	13995	08 Oktober 2015	13860
21 Juli 2015	13350	30 Agustus 2015	14010	09 Oktober 2015	13750
22 Juli 2015	13369	31 Agustus 2015	14010	10 Oktober 2015	13420
23 Juli 2015	13422	01 September 2015	14110	11 Oktober 2015	13420
24 Juli 2015	13440	02 September 2015	14090	12 Oktober 2015	13420
25 Juli 2015	13470	03 September 2015	14133	13 Oktober 2015	13512
26 Juli 2015	13440	04 September 2015	14170	14 Oktober 2015	13598
27 Juli 2015	13440	05 September 2015	14288	15 Oktober 2015	13470
28 Juli 2015	13478	06 September 2015	14290	16 Oktober 2015	13475
29 Juli 2015	13443	07 September 2015	14290	17 Oktober 2015	13535
30 Juli 2015	13448	08 September 2015	14310	18 Oktober 2015	13525
31 Juli 2015	13521	09 September 2015	14270	19 Oktober 2015	13525
01 Agustus 2015	13490	10 September 2015	14265	20 Oktober 2015	13660
02 Agustus 2015	13505	11 September 2015	14295	21 Oktober 2015	13720
03 Agustus 2015	13505	12 September 2015	14295	22 Oktober 2015	13790
04 Agustus 2015	13467	13 September 2015	14320	23 Oktober 2015	13535
05 Agustus 2015	13512	14 September 2015	14320	24 Oktober 2015	13610
06 Agustus 2015	13512,5	15 September 2015	14335	25 Oktober 2015	13605
07 Agustus 2015	13545	16 September 2015	14423	26 Oktober 2015	13605
08 Agustus 2015	13525	17 September 2015	14428	27 Oktober 2015	13560
09 Agustus 2015	13513	18 September 2015	14452	28 Oktober 2015	13652
10 Agustus 2015	13513	19 September 2015	14368	29 Oktober 2015	13525
11 Agustus 2015	13505	20 September 2015	14413	30 Oktober 2015	13635
12 Agustus 2015	13698	21 September 2015	14413	31 Oktober 2015	13719
13 Agustus 2015	13895	22 September 2015	14460	01 November 2015	13685
14 Agustus 2015	13777	23 September 2015	14625	02 November 2015	13685
15 Agustus 2015	13815	24 September 2015	14683	03 November 2015	13597,5
16 Agustus 2015	13815	25 September 2015	14765	04 November 2015	13575
17 Agustus 2015	13815	26 September 2015	14647	05 November 2015	13540
18 Agustus 2015	13868	27 September 2015	14667	06 November 2015	13555
19 Agustus 2015	13853	28 September 2015	14667	07 November 2015	13715
20 Agustus 2015	13820	29 September 2015	14707	08 November 2015	13660
21 Agustus 2015	13900	30 September 2015	14682	09 November 2015	13660
22 Agustus 2015	13985	01 Oktober 2015	14637	10 November 2015	13702
23 Agustus 2015	14015	02 Oktober 2015	14660	11 November 2015	13595
24 Agustus 2015	14015	03 Oktober 2015	14663	12 November 2015	13607,5
25 Agustus 2015	14110	04 Oktober 2015	14722,5	13 November 2015	13645

Tanggal	Nilai Tukar (IDR)
14 November 2015	13765
15 November 2015	13773
16 November 2015	13773
17 November 2015	13773
18 November 2015	13737
19 November 2015	13803
20 November 2015	13695
21 November 2015	13607
22 November 2015	13635
23 November 2015	13635
24 November 2015	13697
25 November 2015	13637,5
26 November 2015	13690
27 November 2015	13780
28 November 2015	13833
29 November 2015	13788
30 November 2015	13788
01 Desember 2015	13870
02 Desember 2015	13830
03 Desember 2015	13822
04 Desember 2015	13838
05 Desember 2015	13810
06 Desember 2015	13808
07 Desember 2015	13808
08 Desember 2015	13862
09 Desember 2015	14035,5
10 Desember 2015	14005
11 Desember 2015	13963
12 Desember 2015	14125
13 Desember 2015	14170
14 Desember 2015	14170
15 Desember 2015	14075
16 Desember 2015	14042
17 Desember 2015	14010
18 Desember 2015	14030
19 Desember 2015	13843
20 Desember 2015	13905
21 Desember 2015	13905
22 Desember 2015	13740
23 Desember 2015	13640

Tanggal	Nilai Tukar (IDR)
24 Desember 2015	13642
25 Desember 2015	13650
26 Desember 2015	13645
27 Desember 2015	13645
28 Desember 2015	13645
29 Desember 2015	13593,5
30 Desember 2015	13685
31 Desember 2015	13858
01 Januari 2016	13875
02 Januari 2016	13790
03 Januari 2016	13790
04 Januari 2016	13790
05 Januari 2016	13975
06 Januari 2016	13868
07 Januari 2016	13875
08 Januari 2016	13840
09 Januari 2016	13930
10 Januari 2016	13930
11 Januari 2016	13930
12 Januari 2016	13837,5
13 Januari 2016	13850
14 Januari 2016	13802
15 Januari 2016	13880
16 Januari 2016	13987
17 Januari 2016	13983
18 Januari 2016	13983
19 Januari 2016	13930
20 Januari 2016	13842
21 Januari 2016	13975
22 Januari 2016	13839
23 Januari 2016	13800
24 Januari 2016	13797
25 Januari 2016	13797
26 Januari 2016	13863
27 Januari 2016	13830
28 Januari 2016	13847
29 Januari 2016	13820
30 Januari 2016	13703
31 Januari 2016	13685
01 Februari 2016	13685

Tanggal	Nilai Tukar (IDR)
02 Februari 2016	13645
03 Februari 2016	13717
04 Februari 2016	13750
05 Februari 2016	13610
06 Februari 2016	13635
07 Februari 2016	13635
08 Februari 2016	13635
09 Februari 2016	13688
10 Februari 2016	13610
11 Februari 2016	13455
12 Februari 2016	13495
13 Februari 2016	13487
14 Februari 2016	13505
15 Februari 2016	13505
16 Februari 2016	13395
17 Februari 2016	13510
18 Februari 2016	13435
19 Februari 2016	13540
20 Februari 2016	13480
21 Februari 2016	13477
22 Februari 2016	13477
23 Februari 2016	13400
24 Februari 2016	13425
25 Februari 2016	13432,5
26 Februari 2016	13457
27 Februari 2016	13405
28 Februari 2016	13435
29 Februari 2016	13435
01 Maret 2016	13345
02 Maret 2016	13335
03 Maret 2016	13261,5
04 Maret 2016	13177
05 Maret 2016	13015
06 Maret 2016	13038
07 Maret 2016	13038
08 Maret 2016	13040
09 Maret 2016	13183
10 Maret 2016	13112
11 Maret 2016	13192,5
12 Maret 2016	13015

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
13 Maret 2016	12993	22 April 2016	13170	01 Juni 2016	13638
14 Maret 2016	12993	23 April 2016	13205	02 Juni 2016	13678
15 Maret 2016	13060	24 April 2016	13207,5	03 Juni 2016	13660
16 Maret 2016	13183	25 April 2016	13207,5	04 Juni 2016	13467,5
17 Maret 2016	13298	26 April 2016	13217,5	05 Juni 2016	13423
18 Maret 2016	12990	27 April 2016	13183	06 Juni 2016	13423
19 Maret 2016	13092	28 April 2016	13185	07 Juni 2016	13367
20 Maret 2016	13088	29 April 2016	13175	08 Juni 2016	13217
21 Maret 2016	13088	30 April 2016	13213	09 Juni 2016	13265
22 Maret 2016	13188	01 Mei 2016	13190	10 Juni 2016	13267
23 Maret 2016	13141	02 Mei 2016	13190	11 Juni 2016	13343
24 Maret 2016	13228	03 Mei 2016	13138	12 Juni 2016	13353
25 Maret 2016	13296	04 Mei 2016	13254	13 Juni 2016	13353
26 Maret 2016	13265	05 Mei 2016	13362	14 Juni 2016	13317
27 Maret 2016	13270	06 Mei 2016	13355	15 Juni 2016	13405
28 Maret 2016	13270	07 Mei 2016	13329	16 Juni 2016	13348
29 Maret 2016	13375	08 Mei 2016	13357,5	17 Juni 2016	13395
30 Maret 2016	13315	09 Mei 2016	13357,5	18 Juni 2016	13342
31 Maret 2016	13248	10 Mei 2016	13380	19 Juni 2016	13337
01 April 2016	13142	11 Mei 2016	13315	20 Juni 2016	13337
02 April 2016	13120	12 Mei 2016	13305	21 Juni 2016	13240
03 April 2016	13137	13 Mei 2016	13305	22 Juni 2016	13237,5
04 April 2016	13137	14 Mei 2016	13340	23 Juni 2016	13250
05 April 2016	13140	15 Mei 2016	13337	24 Juni 2016	13102
06 April 2016	13260	16 Mei 2016	13337	25 Juni 2016	13420
07 April 2016	13195	17 Mei 2016	13330	26 Juni 2016	13425
08 April 2016	13207	18 Mei 2016	13322	27 Juni 2016	13425
09 April 2016	13127,5	19 Mei 2016	13408	28 Juni 2016	13385
10 April 2016	13125	20 Mei 2016	13615	29 Juni 2016	13215,5
11 April 2016	13125	21 Mei 2016	13610	30 Juni 2016	13153
12 April 2016	13105	22 Mei 2016	13617,5	01 Juli 2016	13195
13 April 2016	13078	23 Mei 2016	13617,5	02 Juli 2016	13115
14 April 2016	13188	24 Mei 2016	13575	03 Juli 2016	13120
15 April 2016	13179	25 Mei 2016	13652	04 Juli 2016	13120
16 April 2016	13185	26 Mei 2016	13611	05 Juli 2016	13100
17 April 2016	13182,5	27 Mei 2016	13555	06 Juli 2016	13173
18 April 2016	13182,5	28 Mei 2016	13610	07 Juli 2016	13125
19 April 2016	13119	29 Mei 2016	13605	08 Juli 2016	13220
20 April 2016	13092	30 Mei 2016	13605	09 Juli 2016	13110
21 April 2016	13169	31 Mei 2016	13641	10 Juli 2016	13125

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
11 Juli 2016	13125	20 Agustus 2016	13178	29 September 2016	12957
12 Juli 2016	13120	21 Agustus 2016	13155	30 September 2016	13004
13 Juli 2016	13090	22 Agustus 2016	13172	01 Oktober 2016	13024
14 Juli 2016	13045	23 Agustus 2016	13216	02 Oktober 2016	13021
15 Juli 2016	13045	24 Agustus 2016	13223	03 Oktober 2016	13019
16 Juli 2016	13122	25 Agustus 2016	13241,5	04 Oktober 2016	12982
17 Juli 2016	13107	26 Agustus 2016	13265	05 Oktober 2016	13033
18 Juli 2016	13107	27 Agustus 2016	13269	06 Oktober 2016	13035
19 Juli 2016	13093	28 Agustus 2016	13301	07 Oktober 2016	13021
20 Juli 2016	13095	29 Agustus 2016	13308	08 Oktober 2016	12971
21 Juli 2016	13123	30 Agustus 2016	13262	09 Oktober 2016	12974
22 Juli 2016	13105,5	31 Agustus 2016	13297	10 Oktober 2016	12969
23 Juli 2016	13123	01 September 2016	13292	11 Oktober 2016	13058
24 Juli 2016	13118	02 September 2016	13247	12 Oktober 2016	13037
25 Juli 2016	13118	03 September 2016	13204	13 Oktober 2016	13070
26 Juli 2016	13140	04 September 2016	13202	14 Oktober 2016	13071
27 Juli 2016	13170	05 September 2016	13202	15 Oktober 2016	13076
28 Juli 2016	13142	06 September 2016	13133	16 Oktober 2016	13085
29 Juli 2016	13073	07 September 2016	13072	17 Oktober 2016	13059
30 Juli 2016	13057	08 September 2016	13058	18 Oktober 2016	13018
31 Juli 2016	13067,5	09 September 2016	13096	19 Oktober 2016	12992
01 Agustus 2016	13067,5	10 September 2016	13173	20 Oktober 2016	13018
02 Agustus 2016	13020	11 September 2016	13184	21 Oktober 2016	13037
03 Agustus 2016	13092	12 September 2016	13099,5	22 Oktober 2016	13053
04 Agustus 2016	13154	13 September 2016	13170	23 Oktober 2016	13053
05 Agustus 2016	13110	14 September 2016	13250	24 Oktober 2016	13037
06 Agustus 2016	13123	15 September 2016	13209	25 Oktober 2016	12994
07 Agustus 2016	13127	16 September 2016	13143	26 Oktober 2016	13017
08 Agustus 2016	13127	17 September 2016	13177	27 Oktober 2016	13045
09 Agustus 2016	13132	18 September 2016	13170	28 Oktober 2016	13070
10 Agustus 2016	13105	19 September 2016	13170	29 Oktober 2016	13070
11 Agustus 2016	13083	20 September 2016	13146	30 Oktober 2016	13070
12 Agustus 2016	13087	21 September 2016	13145	31 Oktober 2016	13046
13 Agustus 2016	13105	22 September 2016	13110	01 November 2016	13058
14 Agustus 2016	13115	23 September 2016	13045	02 November 2016	13053
15 Agustus 2016	13115	24 September 2016	13079	03 November 2016	13074
16 Agustus 2016	13066	25 September 2016	13085	04 November 2016	13108
17 Agustus 2016	13105	26 September 2016	13085	05 November 2016	13120
18 Agustus 2016	13147	27 September 2016	13029	06 November 2016	13120
19 Agustus 2016	13105	28 September 2016	12930	07 November 2016	13110

Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)	Tanggal	Nilai Tukar (IDR)
08 November 2016	13077	18 Desember 2016	13416	27 Januari 2017	13347
09 November 2016	13153	19 Desember 2016	13387	28 Januari 2017	13364
10 November 2016	13461	20 Desember 2016	13455,3	29 Januari 2017	13364
11 November 2016	13510,5	21 Desember 2016	13450,3	30 Januari 2017	13327,3
12 November 2016	13365	22 Desember 2016	13463	31 Januari 2017	13350
13 November 2016	13365	23 Desember 2016	13435,8	01 Februari 2017	13397,8
14 November 2016	13515,5	24 Desember 2016	13440	02 Februari 2017	13330,3
15 November 2016	13328	25 Desember 2016	13440	03 Februari 2017	13315,3
16 November 2016	13508	26 Desember 2016	13429,5	04 Februari 2017	13325
17 November 2016	13397	27 Desember 2016	13444,3	05 Februari 2017	13325
18 November 2016	13340	28 Desember 2016	13485,3	06 Februari 2017	13305,8
19 November 2016	13370	29 Desember 2016	13471,3	07 Februari 2017	13320,8
20 November 2016	13370	30 Desember 2016	13486	08 Februari 2017	13314,3
21 November 2016	13445,5	31 Desember 2016	13525	09 Februari 2017	13297
22 November 2016	13437	01 Januari 2017	13525	10 Februari 2017	13302,8
23 November 2016	13615,5	02 Januari 2017	13535,5	11 Februari 2017	13320
24 November 2016	13550	03 Januari 2017	13470	12 Februari 2017	13319
25 November 2016	13528	04 Januari 2017	13345,5	13 Februari 2017	13328
26 November 2016	13515	05 Januari 2017	13332,5	14 Februari 2017	13324
27 November 2016	13541	06 Januari 2017	13329	15 Februari 2017	13320,3
28 November 2016	13516	07 Januari 2017	13349	16 Februari 2017	13347,3
29 November 2016	13543	08 Januari 2017	13349	17 Februari 2017	13342
30 November 2016	13600	09 Januari 2017	13300	18 Februari 2017	13345
01 Desember 2016	13530	10 Januari 2017	13271	19 Februari 2017	13345
02 Desember 2016	13469,5	11 Januari 2017	13315,5	20 Februari 2017	13342,5
03 Desember 2016	13467	12 Januari 2017	13275	21 Februari 2017	13360
04 Desember 2016	13467	13 Januari 2017	13344	22 Februari 2017	13342
05 Desember 2016	13437	14 Januari 2017	13320	23 Februari 2017	13311
06 Desember 2016	13382,5	15 Januari 2017	13320	24 Februari 2017	13340
07 Desember 2016	13290	16 Januari 2017	13346	25 Februari 2017	13330
08 Desember 2016	13301	17 Januari 2017	13305	26 Februari 2017	13330
09 Desember 2016	13346	18 Januari 2017	13353	27 Februari 2017	13355
10 Desember 2016	13352	19 Januari 2017	13390	28 Februari 2017	13346
11 Desember 2016	13352	20 Januari 2017	13427	01 Maret 2017	13340
12 Desember 2016	13292	21 Januari 2017	13400	02 Maret 2017	13383
13 Desember 2016	13267,8	22 Januari 2017	13400	03 Maret 2017	13373
14 Desember 2016	13285	23 Januari 2017	13341	04 Maret 2017	13323
15 Desember 2016	13427	24 Januari 2017	13322	05 Maret 2017	13323
16 Desember 2016	13402,3	25 Januari 2017	13343	06 Maret 2017	13322,8
17 Desember 2016	13416	26 Januari 2017	13395	07 Maret 2017	13337,8

Tanggal	Nilai Tukar (IDR)
08 Maret 2017	13407
09 Maret 2017	13394

Tanggal	Nilai Tukar (IDR)
10 Maret 2017	13363
11 Maret 2017	13356

## LAMPIRAN B HASIL UJI COBA

### B.1 Hasil Uji Coba Jumlah *Hidden Neuron*

Jumlah <i>Hidden Neuron</i>	Nilai MAPE (%) Percobaan Ke- <i>i</i>										Rata-rata MAPE (%)
	1	2	3	4	5	6	7	8	9	10	
2	4,194604	0,817861	0,824841	0,985986	12,737960	0,504016	1,414587	0,689417	1,653970	0,388327	<b>2,421157</b>
3	0,494568	0,527909	0,513291	0,746500	0,500886	0,467152	0,492382	0,481483	0,441589	0,491982	<b>0,515774</b>
4	0,461076	0,484359	0,532880	0,509796	0,425336	0,385920	0,434843	0,459743	0,415652	0,446810	<b>0,455642</b>
5	0,483653	0,402289	0,430690	0,398275	0,425442	0,405229	0,477475	0,537490	0,438730	0,479428	<b>0,447870</b>
6	0,423355	0,399917	0,437309	0,436906	0,442334	0,463682	0,423447	0,410515	0,443312	0,440056	<b>0,432083</b>
7	0,463171	0,418358	0,418353	0,435787	0,424314	0,416705	0,463058	0,445711	0,413895	0,413200	<b>0,431255</b>
8	0,395567	0,409887	0,426667	0,407232	0,399194	0,420484	0,425748	0,462095	0,476103	0,413006	<b>0,423598</b>
9	0,415378	0,416715	0,416166	0,408276	0,404456	0,431174	0,411427	0,404304	0,417544	0,415131	<b>0,414057</b>
10	0,421155	0,409799	0,416234	0,407822	0,408994	0,408937	0,407315	0,402593	0,410357	0,408321	<b>0,410153</b>
11	0,409599	0,414757	0,405008	0,409792	0,410048	0,418648	0,406847	0,409526	0,398803	0,409052	<b>0,409208</b>
12	0,410396	0,412163	0,409206	0,409725	0,411965	0,409944	0,412224	0,412141	0,416620	0,413590	<b>0,411797</b>
13	0,412895	0,409128	0,409317	0,407672	0,412899	0,406468	0,416472	0,409664	0,405843	0,414406	<b>0,410476</b>
14	0,410539	0,411644	0,407630	0,408731	0,406405	0,403240	0,408971	0,408694	0,414388	0,405912	<b>0,408615</b>
15	0,408395	0,410156	0,409469	0,412887	0,410778	0,402007	0,409906	0,409654	0,410374	0,414316	<b>0,409794</b>
16	0,410152	0,403252	0,411284	0,407614	0,405512	0,413819	0,404453	0,405515	0,409882	0,411031	<b>0,408251</b>
17	0,412265	0,404297	0,408259	0,403854	0,402022	0,401386	0,406331	0,405502	0,410311	0,402056	<b>0,405628</b>
18	0,402874	0,407599	0,410834	0,406986	0,401167	0,402604	0,402904	0,414493	0,403766	0,402318	<b>0,405555</b>
19	0,402683	0,414387	0,405218	0,409242	0,403769	0,405265	0,404552	0,404928	0,403073	0,403268	<b>0,405639</b>
20	0,403148	0,402040	0,404558	0,409677	0,404908	0,403684	0,402581	0,405428	0,407429	0,409225	<b>0,405268</b>



## B.2 Hasil Uji Coba Jumlah Fitur

Jumlah Fitur	Nilai MAPE (%) Percobaan Ke- <i>i</i>										Rata-rata MAPE (%)
	1	2	3	4	5	6	7	8	9	10	
3	0,409799	0,409330	0,407776	0,410937	0,407943	0,412228	0,407860	0,408065	0,411206	0,411012	<b>0,409616</b>
4	0,408314	0,407615	0,410759	0,410289	0,408955	0,408419	0,411921	0,407262	0,408858	0,412793	<b>0,409519</b>
5	0,405412	0,410826	0,408745	0,421181	0,407420	0,407214	0,410580	0,412357	0,413157	0,414362	<b>0,411125</b>
6	0,411955	0,413854	0,406034	0,404791	0,434606	0,410765	0,417000	0,412142	0,409066	0,415022	<b>0,413524</b>
7	0,424462	0,408854	0,417926	0,423238	0,449028	0,391956	0,417200	0,412185	0,406462	0,408994	<b>0,416031</b>

## B.3 Hasil Uji Coba Jangka Waktu Prediksi

Jangka Waktu	Data Latih (hari)	Data Uji (hari)	Nilai MAPE (%) Percobaan Ke- <i>i</i>										Rata-rata MAPE (%)
			1	2	3	4	5	6	7	8	9	10	
1 hari	1890	1	0,131507	0,097036	0,061702	0,098360	0,103080	0,071810	0,098598	0,086132	0,053308	0,097046	<b>0,089858</b>
1 minggu	1890	7	0,410546	0,414849	0,418404	0,419957	0,381399	0,410845	0,420671	0,450540	0,432703	0,425062	<b>0,418498</b>
1 bulan	1890	30	0,408250	0,418982	0,408516	0,409113	0,410108	0,411355	0,422171	0,416123	0,409600	0,410150	<b>0,412437</b>
3 bulan	1890	91	0,359210	0,364372	0,367847	0,370680	0,364369	0,363658	0,363249	0,365049	0,362395	0,361371	<b>0,364220</b>
4 bulan	1890	121	0,375936	0,376387	0,376653	0,377272	0,376168	0,375995	0,375299	0,377795	0,377495	0,376369	<b>0,376537</b>
6 bulan	1890	182	0,348519	0,346427	0,346981	0,346113	0,353819	0,347777	0,348931	0,347731	0,347299	0,348457	<b>0,348205</b>
1 tahun	1890	365	0,386505	0,358192	0,370999	0,351672	0,382175	0,385997	0,397002	0,438740	0,403543	0,382200	<b>0,385703</b>
2 tahun	1890	730	0,668624	0,471223	1,191993	0,559296	0,643020	0,653873	0,538075	0,418767	0,477879	0,490973	<b>0,611372</b>
3 tahun	1890	1095	0,576049	0,605966	0,797463	0,641929	0,623066	0,421355	0,914752	0,675845	0,545073	0,629944	<b>0,643144</b>

#### B.4 Hasil Uji Coba Jumlah *Context Neuron*

Jumlah Context Neuron	Nilai MAPE (%) Percobaan Ke- <i>i</i>										Rata-rata MAPE (%)
	1	2	3	4	5	6	7	8	9	10	
1	0,071246	0,076168	0,071492	0,077735	0,102105	0,059461	0,066715	0,072885	0,075213	0,077212	<b>0,075023</b>
2	0,080920	0,082270	0,091132	0,076665	0,083517	0,061913	0,064735	0,092740	0,094374	0,099021	<b>0,082729</b>
3	0,102717	0,092137	0,085597	0,051533	0,091962	0,079026	0,096677	0,074833	0,074996	0,102798	<b>0,085228</b>
4	0,101466	0,068160	0,036046	0,174097	0,110679	0,053002	0,201255	0,069674	0,145309	0,050698	<b>0,101039</b>
5	0,009337	0,131366	0,086204	0,108608	0,071398	0,051730	0,181809	0,080729	0,108340	0,119445	<b>0,094897</b>
6	0,180178	0,108780	0,102611	0,257456	0,142299	0,082249	0,036604	0,238822	0,271442	0,139981	<b>0,156042</b>
7	0,301898	0,278966	0,177707	0,000560	0,039942	0,236873	0,008245	0,046500	0,335577	0,224930	<b>0,165120</b>
8	0,469265	0,124220	0,180003	0,299365	0,231876	0,157481	0,118098	0,003815	0,307845	0,082155	<b>0,197412</b>
9	0,129554	0,071091	0,165559	0,007502	0,192544	0,119604	0,138729	0,165957	0,205283	0,640905	<b>0,183673</b>
10	0,143206	0,139186	0,147279	0,240970	0,025350	0,250239	0,155851	0,182314	0,538470	0,236599	<b>0,205946</b>
11	0,391520	0,125995	0,106897	0,237789	0,447201	0,247853	0,240110	0,464159	0,004046	0,098521	<b>0,236409</b>
12	0,135382	0,132533	0,067419	0,415798	0,240993	0,070339	0,443842	0,207540	0,136701	0,017767	<b>0,186831</b>
13	0,305822	0,504034	0,026216	0,199863	0,310011	0,962463	0,178415	0,601892	0,088048	0,115886	<b>0,329265</b>
14	0,288916	0,463146	0,294471	0,254428	0,185230	0,297719	0,192309	0,115391	0,138576	0,018387	<b>0,224857</b>
15	0,296255	0,159392	0,191865	0,123050	0,687318	0,262527	0,207979	0,376771	0,042799	0,450507	<b>0,279846</b>
16	0,475175	0,002208	0,087375	0,259085	0,009616	0,093942	0,051711	0,202968	0,029648	0,206279	<b>0,141801</b>
17	0,371534	0,039979	0,561387	0,224244	0,148261	0,751075	0,237759	0,016635	0,360864	0,279395	<b>0,299113</b>
18	0,541836	0,054831	0,219748	0,180695	0,133521	0,435182	0,289621	0,459481	0,114088	0,349023	<b>0,277803</b>
19	0,383842	0,193903	0,261909	0,168307	0,428409	0,170241	0,486569	0,542248	0,108688	0,296979	<b>0,304110</b>
20	0,422098	0,175980	0,290490	0,444320	0,256675	0,450480	0,911231	0,018256	0,185848	0,271589	<b>0,342697</b>

### B.5 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Nilai MAPE

Metode	Nilai MAPE (%) Percobaan Ke- <i>i</i>										Rata-rata MAPE (%)
	1	2	3	4	5	6	7	8	9	10	
<b>RELMNN (<i>datasets</i> terdapat <i>outliers</i>)</b>	0,062141	0,027497	0,230403	0,128221	0,081167	0,124351	0,024460	0,147684	0,444289	0,016901	<b>0,128711</b>
<b>ELM (<i>datasets</i> terdapat <i>outliers</i>)</b>	0,066954	0,016593	0,045698	0,225751	0,038132	0,032059	0,045399	0,311371	0,326467	0,114762	<b>0,122319</b>
<b>RELMNN (<i>datasets</i> tanpa <i>outliers</i>)</b>	0,056300	0,078954	0,062583	0,058205	0,077705	0,062198	0,074580	0,077558	0,068967	0,077971	<b>0,069502</b>
<b>ELM (<i>datasets</i> tanpa <i>outliers</i>)</b>	0,065234	0,111507	0,079631	0,089092	0,080008	0,094162	0,080399	0,100044	0,119824	0,084324	<b>0,090423</b>

### B.6 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Selisih Nilai Tukar Rupiah Indonesia terhadap Dolar Amerika

Metode	Rata-rata Selisih Nilai Tukar antara Target dengan Prediksi (IDR) Percobaan Ke- <i>i</i>										Rata-rata Selisih (IDR)
	1	2	3	4	5	6	7	8	9	10	
<b>RELMNN</b>	6,450820	9,046567	7,170730	6,669101	8,903491	7,126643	8,545380	8,886580	7,902225	8,933864	<b>7,963540</b>
<b>ELM</b>	7,474515	12,776487	9,124114	10,208116	9,167282	10,789040	9,212146	11,463026	13,729437	9,661816	<b>10,360598</b>

**B.7 Hasil Uji Coba Perbandingan Metode RELMNN dengan Metode ELM Terhadap Waktu yang Dibutuhkan Pada Proses *Training* dan *Testing***

Metode	Waktu yang Digunakan (Detik) Percobaan Ke- <i>i</i>										Rata-rata Waktu (detik)
	1	2	3	4	5	6	7	8	9	10	
<b>RELMNN (<i>training</i>)</b>	0,024035	0,021425	0,012197	0,012964	0,011580	0,014488	0,022603	0,014725	0,013894	0,011156	<b>0,015907</b>
<b>ELM (<i>training</i>)</b>	0,015163	0,016483	0,011473	0,014142	0,014667	0,012451	0,022667	0,011770	0,012624	0,012559	<b>0,014400</b>
<b>RELMNN (<i>testing</i>)</b>	0,001871	0,002013	0,001096	0,001031	0,001658	0,001031	0,001041	0,001303	0,001792	0,000753	<b>0,001359</b>
<b>ELM (<i>testing</i>)</b>	0,000878	0,000866	0,004333	0,000795	0,002395	0,001442	0,002394	0,003226	0,001015	0,001143	<b>0,001849</b>

## LAMPIRAN C VISUALISASI HASIL UJI COBA

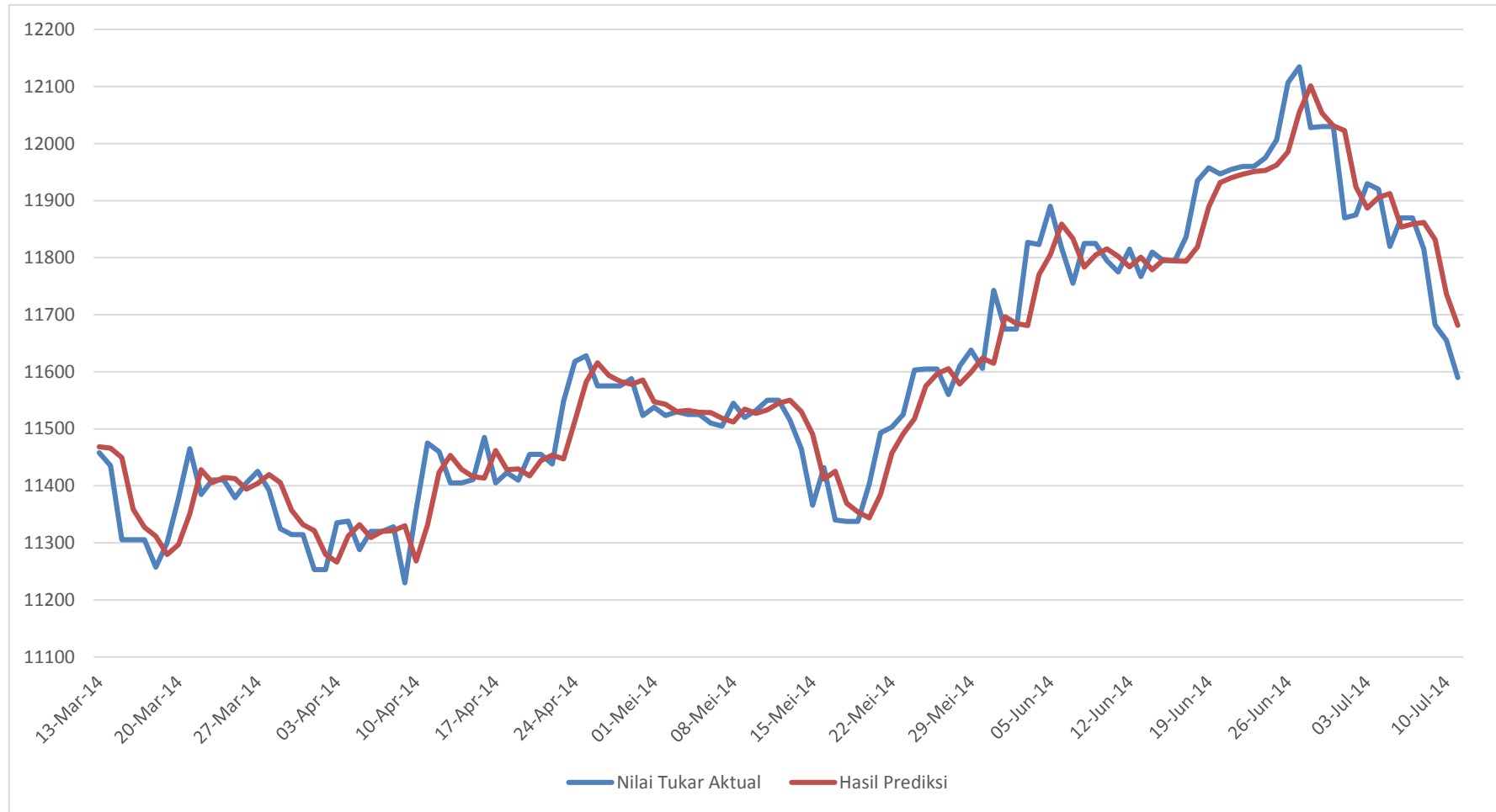
### C.1 Visualisasi Hasil Uji Coba Jangka Waktu 1 Bulan

Grafik di bawah ini merupakan visualisasi hasil prediksi ketika melakukan prediksi pada jangka waktu 1 bulan dan menggunakan data latih 1890 data, 4 fitur, 12 *hidden neuron* dan 1 *context neuron*. Pada hasil uji coba ini didapatkan MAPE sebesar 0,415267.



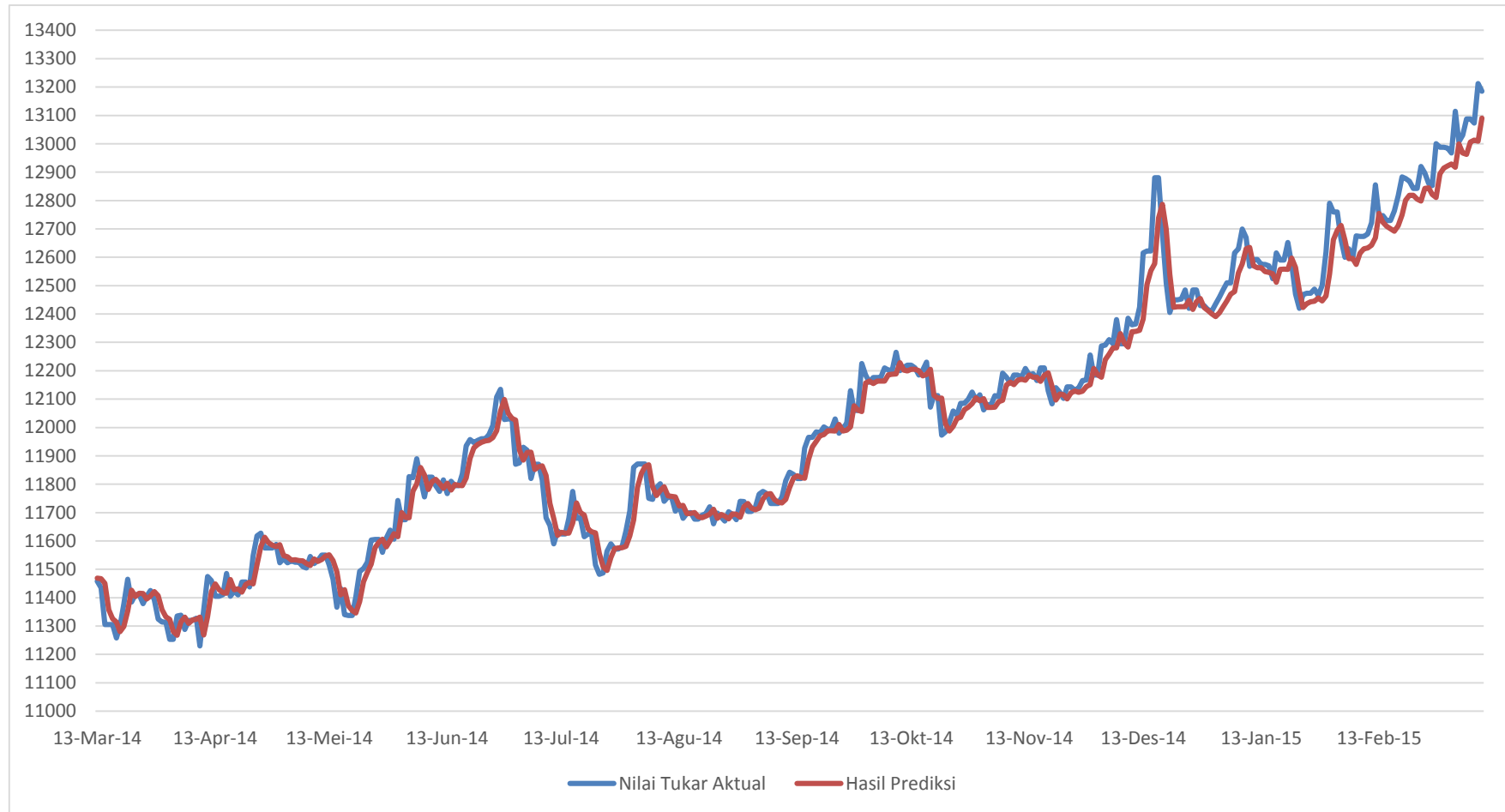
## C.2 Visualisasi Hasil Uji Coba Jangka Waktu 4 Bulan

Grafik di bawah ini merupakan visualisasi hasil prediksi ketika melakukan prediksi pada jangka waktu 4 bulan dan menggunakan data latih 1890 data, 4 fitur, 12 *hidden neuron* dan 1 *context neuron*. Pada hasil uji coba ini didapatkan MAPE sebesar 0.379049.



### C.3 Visualisasi Hasil Uji Coba Jangka Waktu 1 Tahun

Grafik di bawah ini merupakan visualisasi hasil prediksi ketika melakukan prediksi pada jangka waktu 1 tahun dan menggunakan data latih 1890 data, 4 fitur, 12 *hidden neuron* dan 1 *context neuron*. Pada hasil uji coba ini didapatkan MAPE sebesar 0,385587.



#### C.4 Visualisasi Hasil Uji Coba Jumlah Data Latih dan Data Uji yang Sama.

Grafik di bawah ini merupakan visualisasi hasil prediksi ketika melakukan prediksi dengan data uji dan data latih yang sama yaitu sejumlah 1890 data, yang di uji dengan menggunakan 4 fitur, 12 *hidden neuron* dan 1 *context neuron*. Pada hasil uji coba ini didapatkan MAPE sebesar 0,337936.

