



Peramalan Penjualan Harga Saham PT Bank Rakyat (Persero) Tbk BBRI Indonesia dengan Menggunakan *Recurrent Neural Network* (RNN)

Walid^{a,*}

^aJurusan Matematika FMIPA UNNES, Gedung D7 Lt 1, Sekaran Semarang 50229, Indonesia

*Alamat surel: walid.mat@mail.unnes.ac.id

Abstrak

Penelitian ini menggunakan studi literatur dan simulasi program dari data sekunder yang diambil, pengumpulan data dilakukan dengan menggunakan data saham harian PT Bank Rakyat Indonesia (Persero) Tbk (BBRI) durasi waktu Januari 2016 sampai dengan September 2018 yang diakses secara online. Metode analisis data yang digunakan dalam penelitian ini adalah metode analisis data didasarkan pada kajian teori dengan menggunakan model *recurrent neural network*, kajian komputasi dan kajian terapan. Kajian komputasi ini dilakukan untuk memperoleh hasil analisis dengan menggunakan jaringan saraf tiruan dan komputasional tahap-tahap identifikasi data yang memuat penentuan variabel, penyusunan fokus pada pemilihan indikator disetiap variabel yang menjadi objek pilihan data pada penelitian ini, menentukan beberapa nilai awal untuk bias dan bobot parameter serta nilai awal variabel pada iterasi pertama. Kemudian dikembangkan pelatihan dengan menggunakan model yang diperoleh untuk disimulasikan sampai pada perolehan titik epoch pada masing-masing jaringan model RNN yang ada. Analisis selanjutnya membandingkan hasil simulasi dan analisis komputasional model jaringan saraf tiruan dengan menggunakan model RNN Elman untuk diperoleh hasil yang terbaik, dalam analisis komputasi dan simulasi ini menggunakan program MATLAB dan R. Hasil penelitian dan pembahasan disimpulkan bahwa model RNN tipe Elman yang terbaik untuk peramalan penjualan harga saham PT Bank Rakyat Indonesia (Persero) Tbk (BBRI) adalah model RNN tipe Elman pada algoritma pelatihan Levenberg-Marquardt dengan $\mu = 0,013$ diperoleh pada arsitektur jaringan 9-10-1 pada fungsi aktivasi *logsig* dengan MSE dan MAPE pengujian masing-masing sebesar $7,6649e+10$ dan $9,2348\%$.

Kata kunci:

differensi, epoch, konvergen, MSE, noise, RNN

© 2019 Dipublikasikan oleh Jurusan Matematika, Universitas Negeri Semarang

1. Pendahuluan

RNN memiliki karakteristik yang unik, yaitu arsitekturnya memiliki minimal satu *feedback loop*, sehingga dapat menyimpan data dalam struktur jaringannya dan performansi dari RNN ini dalam melakukan prediksi bergantung pada bobot dan arsitekturnya (Hardiantho, dkk, 2011). Keunikan RNN yang lain adalah adanya koneksi umpan balik yang membawa informasi gangguan (*noise*) pada saat *input* sebelumnya yang akan diakomodasikan bagi *input* berikutnya (Salman, 2011). Penelitian ini menggunakan RNN tipe Elman. RNN tipe Elman terdiri dari atas beberapa lapisan tersembunyi dengan lapisan pertama memiliki bobot-bobot yang diperoleh dari lapisan *input*. Setiap lapisan pada jaringan ini akan menerima bobot dari lapisan sebelumnya dan semua lapisan kecuali lapisan terakhir memiliki satu bobot *recurrent*, semua lapisan memiliki bias. Tidak seperti pada BPNN, pada RNN tipe Elman fungsi aktivasi dapat berupa sembarang fungsi, baik yang kontinu maupun yang diskontinu (Kusumadewi, 2004).

Penelitian yang sudah dilakukan dengan menggunakan model RNN antara lain Walid dkk (2018) tentang Analisis Produktivitas Kinerja Dosen dan Tenaga Kependidikan dalam Mewujudkan Tahun Reputasi Universitas Negeri Semarang (UNNES) Menggunakan Jaringan Saraf Tiruan dengan menggunakan RNN. Produktivitas di lingkungan Universitas Negeri Semarang (UNNES) merupakan

suatu interkasi terpadu secara serasi dari tiga faktor esensial, yaitu optimalisasi kemampuan, manajemen dan sumber daya yang dimiliki oleh UNNES. Pemberian motivasi kepada seluruh civitas akademika UNNES sangatlah berpengaruh terhadap jalannya kemajuan UNNES dalam mewujudkan tahun reputasi yang dicanangkan pada tahun 2017 ini oleh Walid dkk (2017).

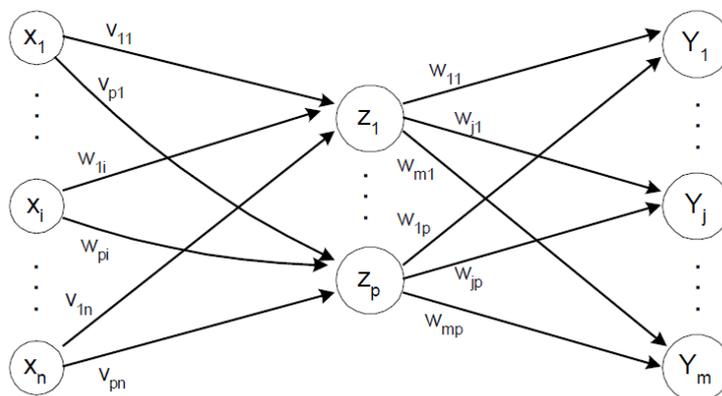
Jaringan saraf tiruan sebagai salah satu sistem pemrosesan informasi yang didesain dengan menirukan cara kerja otak manusia dalam menyelesaikan suatu masalah dengan melakukan proses belajar melalui perubahan bobot sinapsisnya dapat digunakan untuk melakukan analisis terhadap produktivitas keinerja dosen dan karyawan UNNES dalam mewujudkan tahun reputasi yang dicanangkan pada tahun 2017. Penelitian berkaitan dengan pengembangan jaringan saraf tiruan telah cukup banyak dilakukan oleh ahli dibidang ini, antara lain hasil penelitian yang dilakukan oleh Walid dkk (2015) tentang peramalan menggunakan model jaringan saraf tiruan pada data berpola long memory dengan konsentrasi ada penggunaan beban konsumsi listrik PLN di Jawa dan Bali Pergerakan harga saham Bank Rakyat Indonesia (BBRI) pada pasar modal sangatlah dinamis. Harga saham terbentuk atas banyaknya permintaan dan penawaran. Permintaan dan penawaran terjadi karena berbagai faktor, baik yang sifatnya spesifik atas saham (kinerja perusahaan dan industri di mana perusahaan tersebut bergerak), maupun faktor yang sifatnya makro seperti kondisi ekonomi negara, kondisi sosial-politik, maupun rumor-rumor yang berkembang. Kondisi-kondisi tersebut menyebabkan sebagian besar data runtun waktu harga saham tidak stasioner

Menurut Fausett (1994: 3), jaringan syaraf tiruan (JST) atau yang sering dikenal dengan *neural network* (NN) adalah sistem pemroses informasi yang memiliki karakteristik mirip dengan jaringan syaraf biologi. NN dibentuk sebagai generalisasi model matematika dari jaringan syaraf biologi, dengan asumsi sebagai berikut.

- Pemrosesan informasi terjadi pada banyak elemen sederhana yang disebut neuron.
- Sinyal dikirimkan diantara neuron-neuron melalui penghubung-penghubung.
- Setiap penghubung antar neuron memiliki bobot yang akan memperkuat atau memperlemah sinyal (bobot yang bernilai positif akan memperkuat sinyal, sedangkan bobot yang bernilai negatif akan memperlemah sinyal).
- Setiap neuron menggunakan fungsi aktivasi (biasanya bukan fungsi linier) terhadap *input* (jumlah sinyal *input* yang terboboti) untuk menentukan sinyal *output*.

1.1. Jaringan lapisan jamak (multi-layer network)

Jaringan lapisan jamak merupakan perluasan dari lapisan tunggal. Dalam jaringan ini, selain neuron *input* dan *output*, ada neuron-neuron lain (sering disebut lapisan tersembunyi). Dimungkinkan pula ada beberapa lapisan tersembunyi. Sama seperti pada neuron *input* dan *output*, neuron-neuron dalam satu lapisan tidak saling berhubungan.



(Sumber: Siang, 2004: 25)

Gambar 1. Jaringan Lapisan Jamak

Gambar 1 adalah jaringan dengan n buah *input* (x_1, x_2, \dots), sebuah lapisan tersembunyi yang terdiri dari p buah neuron (z_1, \dots), dan m buah neuron *output* (y_1, y_2, \dots). Jaringan lapisan jamak

dapat menyelesaikan masalah yang lebih kompleks dibandingkan dengan lapisan tunggal, meskipun kadangkala proses pelatihan lebih kompleks dan lama.

1.2. Jaringan recurrent

Model jaringan *recurrent* mirip dengan jaringan lapisan tunggal atau ganda, hanya saja, ada neuron *output* yang memberikan sinyal pada neuron *input* (sering disebut *feedback loop*).

1.3. Fungsi sigmoid biner

Fungsi ini digunakan untuk NN yang dilatih dengan menggunakan metode *backpropagation*. Fungsi sigmoid biner memiliki nilai pada *range* 0 sampai 1. Oleh karena itu, fungsi ini sering digunakan untuk NN yang membutuhkan nilai *output* yang terletak pada interval 0 sampai 1. Namun, fungsi ini bisa juga digunakan oleh NN yang nilai *output*-nya 0 atau 1.

Fungsi sigmoid biner dirumuskan sebagai:

$$y = f(x) = \frac{1}{1 + e^{-\sigma x}} \quad (11)$$

dengan: $f'(x) = \sigma f(x)[1 - f(x)]$.

Fungsi aktivasi sigmoid biner pada Matlab dikenal dengan nama *logsig*. *Syntax* untuk fungsi tersebut adalah:

$Y = \text{logsig}(x)$

1.4. Fungsi sigmoid bipolar

Fungsi sigmoid bipolar hampir sama dengan fungsi sigmoid biner, hanya saja *output* dari fungsi ini memiliki *range* antara -1 sampai 1.

Fungsi sigmoid bipolar dirumuskan sebagai:

$$y = f(x) = \frac{1 - e^{-\sigma x}}{1 + e^{-\sigma x}} \quad (2)$$

dengan: $f'(x) = \frac{\sigma}{2} [1 + f(x)][1 - f(x)]$.

Fungsi ini sangat dekat dengan fungsi *hyperbolic tangent*. Keduanya memiliki *range* antara -1 sampai 1. Untuk fungsi *hyperbolic tangent*, dirumuskan sebagai:

$$y = f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (3)$$

Atau

$$y = f(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}} \quad (4)$$

dengan: $f'(x) = [1 + f(x)][1 - f(x)]$.

Fungsi aktivasi sigmoid bipolar pada Matlab dikenal dengan nama *tansig*. *Syntax* untuk fungsi tersebut adalah:

$Y = \text{tansig}(x)$

1.5. Recurrent Neural Network

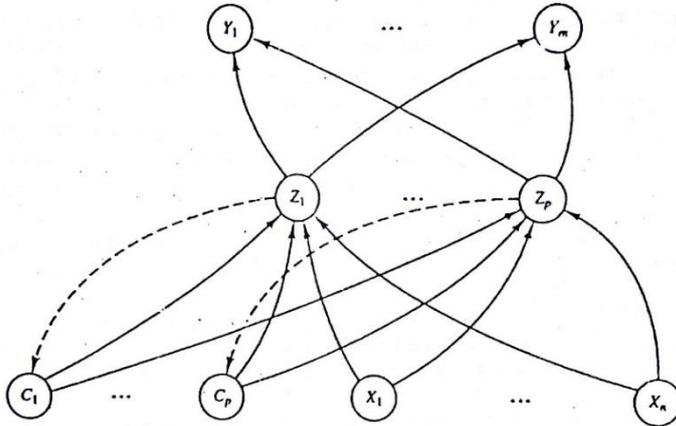
Menurut Purnomo & Kurniawan (2006: 64), *Recurrent Neural Network* (RNN) adalah NN dengan fasilitas umpan balik menuju neuron itu sendiri maupun neuron yang lain, sehingga aliran informasi dari *input* mempunyai arah banyak (*multidirectional*). *Output* NN tidak hanya tergantung pada *input* saat itu saja, tetapi juga tergantung pada kondisi *input* NN untuk waktu sebelumnya. Kondisi ini dimaksudkan untuk menampung kejadian sebelumnya diikutkan pada proses komputasi berikutnya.

Hal ini penting untuk problematika yang cukup rumit dan tanggapan *output* NN berkaitan dengan variasi waktu (*time-varying*), sehingga NN memiliki *sense* terhadap waktu dan memori kondisi sebelumnya. Hal ini erat dengan suatu cabang ilmu yang disebut automaton (*finite automata/finite state machine*). Pada prinsipnya RNN sama dengan BPNN dengan tambahan neuron konteks yang hanya menerima *input* internal (*input* balik dari lapisan tersembunyi atau *output*). Conroy, dkk (1994) menyatakan bahwa RNN didasarkan pada penyaringan *outlier* dari data dan estimasi parameter dari data yang difilter.

RNN dapat dianggap sebagai jaringan “*recurrent* sebagian”, dimana sebagian besar koneksi hanya *feedforward*. Sekelompok neuron tertentu menerima sinyal umpan balik (*feedback*) dari langkah waktu sebelumnya. Neuron itu diketahui sebagai neuron konteks (Fausett, 1994: 374).

RNN adalah jaringan yang mengakomodasi *output* jaringan untuk menjadi *input* pada jaringan itu lagi dalam rangka menghasilkan *output* jaringan berikutnya. Ada dua macam RNN pada *toolbox* Matlab, yaitu Jaringan Elman dan Jaringan Hopfield (Kusumadewi, 2004).

Arsitektur untuk RNN sederhana seperti yang ditunjukkan pada Gambar 2 (Fausett, 1994: 375).



(Sumber: Fausett, 1994: 375)

Gambar 2. Arsitektur *Recurrent Neural Network* Sederhana

Aktivasi dari neuron-neuron konteks pada waktu adalah aktivasi (sinyal *output*) dari neuron-neuron tersembunyi pada langkah waktu sebelumnya. Bobot dari neuron konteks ke neuron tersembunyi dilatih dengan cara yang sama persis dengan bobot dari neuron *input* ke neuron tersembunyi. Dengan demikian, pada setiap langkah waktu, algoritma pelatihan adalah sama seperti untuk BPNN sederhana.

Model umum jaringan RNN dengan satu lapisan tersembunyi, dengan neuron *input* dan neuron pada lapisan tersembunyi adalah (Sani, 2014):

$$\hat{Y} = f^o \left(\beta_0 + \sum_{j=1}^p \left(\beta_j f^h \left(\gamma_{j0} + \gamma_{dj} + \sum_{i=1}^q \gamma_{ji} X_i \right) \right) \right) \quad (2)$$

dengan:

\hat{Y} : variabel *output*

β_0 : bobot bias pada lapisan tersembunyi, $j = 1, 2, \dots, p$

β_j : bobot untuk neuron tersembunyi ke- j , $j = 1, 2, \dots, p$

γ_{ji} : bobot dari neuron *input* ke- i menuju neuron tersembunyi ke- j

γ_{dj} : bobot *delay* atau neuron konteks

γ_{j0} : bobot bias pada lapisan *input*, $i = 1, 2, \dots, n$

X_i : neuron *input*, $i = 1, 2, \dots, n$

$f^h(x)$: fungsi aktivasi pada lapisan tersembunyi

$f^o(x)$: fungsi aktivasi pada lapisan *output*

Jaringan Elman biasanya menggunakan fungsi aktivasi sigmoid bipolar untuk lapisan tersembunyi (*recurrent*) dan linear untuk lapisan *output*. Tidak seperti pada BPNN, pada jaringan Elman ini, fungsi aktivasi dapat berupa sembarang fungsi, baik yang kontinu maupun yang diskontinu. Neuron pada setiap jaringan, ditetapkan cukup banyak, demikian pula jumlah lapisan tersembunyi disesuaikan dengan kompleksitas permasalahan. Delay yang terjadi pada waktu sebelumnya ($t-1$) dapat digunakan untuk waktu saat ini (t). Sehingga apabila dipilih dua RNN dengan bobot-bobot awal yang sama dan diberikan pada input yang sama, bisa jadi akan menghasilkan output jaringan yang berbeda. Seperti halnya pada BPNN, proses pelatihan pada jaringan Elman dapat dilakukan dengan dua cara, yaitu dengan fungsi *train* dan *adapt*.

Berdasarkan latar belakang di atas permasalahan yang akan diteliti adalah bagaimana Peramalan penjualan harga saham PT Bank Rakyat Indonesia (Persero) Tbk (BBRI) dengan Menggunakan Recurrent Neural Network (RNN)

2. Metode Penelitian

Penelitian ini dibagi menjadi beberapa metode tahapan penelitian, secara lengkap dijabarkan sebagai berikut.

2.1. Metode Pengumpulan Data

Pengumpulan data dilakukan dengan menggunakan data sekunder yaitu data saham harian PT Bank Rakyat Indonesia (Persero) Tbk (BBRI) durasi waktu januari 2016 sampai dengan september 2018 yang diakses secara online dari <https://finance.yahoo.com/>.

2.2. Metode Analisis Data

Metode analisis data yang digunakan dalam penelitian ini adalah metode analisis data didasarkan pada kajian teori, kajian komputasi dan kajian terapan. Berdasarkan fokus penelitian tersebut, tahapan penelitian ini dijabarkan secara detail dalam pemaparan berikut.

2.2.1. Kajian Teori

Studi literature dan kajian teoritis mendalam dilakukan untuk memperoleh kajian identifikasi data yang memuat model jaringan saraf tiruan, untuk penentuan nilai bias awal, bobot awal dan model jaringan RNN yang paling baik. Selain itu untuk memperoleh kajian pemetaan terhadap penggunaan aplikasi model jaringan saraf tiruan yang berkembangnya saat ini dikaji secara mendalam sebagai dasar penggunaan keilmuaan untuk menganalisis kondisi yang sebenarnya.

2.2.2. Kajian Komputasi

Kajian komputasi ini dilakukan untuk memperoleh hasil analisis dengan menggunakan jaringan saraf tiruan dan komputasional tahap-tahap identifikasi data yang memuat penentuan variabel, penyusunan fokus pada pemilihan indikator disetiap variabel yang menjadi objek pilihan data pada penelitian ini, menentukan beberapa nilai awal untuk bias dan bobot parameter serta nilai awal variabel pada iterasi pertama. Kemudian dikembangkan pelatihan dengan menggunakan model yang diperoleh untuk disimulasikan sampai pada perolehan titik epoch pada masing-masing jaringan model RNN yang ada. Analisis selanjutnya membandingkan hasil simulasi dan analisis komputasional model jaringan saraf tiruan dengan menggunakan model RNN Elman untuk diperoleh hasil yang terbaik, dalam analisis komputasi dan simulasi ini menggunakan program MATLAB dan R.

3. Hasil Penelitian dan Pembahasan

Peramalan penjualan harga saham PT Bank Rakyat Indonesia (Persero) Tbk (BBRI) dengan Menggunakan *Recurrent Neural Network* (RNN) tipe Elman dilakukan dengan menggunakan langkah-langkah pemodelan RNN tipe Elman antara lain.

3.1. Penentuan Input dan Target Jaringan

Plot PACF menunjukkan lag-lag yang signifikan, yaitu lag 2, lag 5, lag 9, lag 12, lag 13, sehingga jaringan yang dibangun memiliki *input*, x_{t-2} , x_{t-5} , x_{t-9} , x_{t-12} , x_{t-13} , dengan target x_t atau dapat dikatakan bahwa x_t dan banyaknya data menjadi 76 data. Pembagian data *training* sebanyak 64 dan data uji sebanyak 8 data.

3.2. Normalisasi Data

Data *input* dan target pada data latih harus dinormalisasikan terlebih dahulu. Perintah untuk menormalisasi data *input* dan target pada data latih sebagai berikut.

[pn,ps] = mapstd(inputlatih)
 [tn,ts] = mapstd(targetlatih)

3.3. Tahap Pembentukan Model

Pembentukan model terbaik dalam membangun jaringan RNN tipe Elman dilakukan dengan menggunakan langkah-langkah sebagai berikut.

3.3.1. Penentuan jaringan yang optimum

Lapisan tersembunyi yang digunakan ada satu lapisan tersembunyi, dan untuk neuron tersembunyi diperoleh berdasarkan *trial and error* pada jumlah neuron tersembunyi untuk pelatihan RNN tipe Elman, yaitu 5 neuron, 10 neuron, 15 neuron. Fungsi aktivasi yang digunakan pada lapisan tersembunyi dilakukan *trial and error* pada satu fungsi aktivasi, yaitu fungsi sigmoid biner yang pada Matlab dikenal dengan nama *logsig*, sedangkan pada lapisan *output* digunakan fungsi linear atau identitas yang pada Matlab dikenal dengan nama *purelin*.

3.3.2. Penentuan algoritma pelatihan

Algoritma pelatihan yang digunakan untuk membangun jaringan RNN dilakukan *trial and error* pada dua algoritma pelatihan, yaitu *gradient descent* dengan momentum dan *adaptive learning rate* yang pada Matlab dikenal dengan nama *traingdx* dan *Levenberg-Marquardt* yang pada Matlab dikenal dengan nama *trainlm*.

3.3.3. Denormalisasi Data

Setelah proses pelatihan selesai dilakukan, maka hasil *output* jaringan yang ternormalisasi dikembalikan lagi seperti semula yang disebut dengan denormalisasi data. Perintah untuk mendenormalisasi *output* pada tahap pelatihan sebagai berikut.

a = mapstd('reverse',an,ts)

Perintah untuk mendenormalisasi *output* pada tahap pengujian sebagai berikut.

anew = mapstd('reverse',anewn,ts)

3.3.4. Tahap Penentuan Model

Tahap penentuan model terdiri dari penentuan model terbaik dan uji kesesuaian model.

- Penentuan model terbaik

Model terbaik terbentuk berdasarkan *trial and error* terhadap beberapa macam arsitektur jaringan. Model terbaik adalah model dengan MSE dan MAPE terkecil dengan jaringan paling sederhana. Berikut adalah hasil pelatihan jaringan RNN tipe Elman dengan menggunakan algoritma pelatihan *gradient descent* dengan momentum dan *adaptive learning rate* dan algoritma pelatihan *Levenberg-Marquardt*.

Hasil pelatihan pada jaringan RNN tipe Elman menggunakan algoritma pelatihan *Levenberg-Marquardt* (*trainlm*) berdasarkan model terbaik dari masing-masing μ .

Tabel 1. Hasil Pelatihan RNN Tipe Elman dengan $\mu = 0,002$

Arsitektur Jaringan	$f(x)$	Epoch	Perf	Pelatihan			Pengujian		
				MSE	MAPE (%)	R	MSE	MAPE (%)	R
9-5-1	Logsig	1224	0,0273	3,9232e+09	2,6322	0,98612	8,8855e+10	10,6724	0,63127

Berdasarkan Tabel 1 dapat dilihat bahwa model terbaik RNN tipe Elman pada algoritma pelatihan *Levenberg-Marquardt* dengan $\mu = 0,002$ diperoleh pada arsitektur jaringan 9-5-1 pada fungsi aktivasi *logsig* dengan MSE dan MAPE pengujian masing-masing sebesar 8,8855e+10 dan 10,6724%.

Tabel 2. Hasil Pelatihan RNN Tipe Elman dengan $\mu = 0,003$

Arsitektur Jaringan	$f(x)$	Epoch	Perf	Pelatihan			Pengujian		
				MSE	MAPE	R	MSE	MAPE	R

		0,000			1,1124e+0		0,3748		0,999		6,8949e+1		10,3349		0,703	
		774			8				61		0				24	
9-10-1	Logsig	17														

Berdasarkan Tabel 2 dapat dilihat bahwa model terbaik RNN tipe Elman pada algoritma pelatihan *Levenberg-Marquardt* dengan $\mu = 0,003$ diperoleh pada arsitektur jaringan 9-10-1 pada fungsi aktivasi logsig dengan MSE dan MAPE pengujian masing-masing sebesar $7,6649e+10$ dan $9,2348\%$.

Berdasarkan hasil pelatihan RNN tipe Elman dengan algoritma pelatihan *gradient descent* dengan momentum dan *adaptive learning rate* dan algoritma pelatihan *Levenberg-Marquardt* diperoleh MSE dan MAPE pengujian terkecil dengan nilai MSE serta momentum yang digunakan adalah $0,2$, *learning rate* yang digunakan adalah $0,2$, dan fungsi aktivasi yang digunakan adalah logsig. Oleh karena itu, model yang akan digunakan untuk peramalan dalam penelitian ini adalah model RNN tipe Elman adalah model dengan arsitektur jaringan 9-10-1 yang terdiri dari 9 neuron pada lapisan *input*, 5 neuron pada lapisan tersembunyi, dan 1 neuron pada lapisan *output*.

Proses peramalan menggunakan model RNN terbangun dari 9 neuron pada lapisan *input*, 5 neuron pada lapisan tersembunyi, dan 1 neuron pada lapisan *output*. Diperolehnya input selanjutnya *input* dinormalisasikan. Perhitungan pada lapisan *input* ke lapisan tersembunyi sebagai berikut.

$$[x_1, v_{11}][x_1, v_{21}][x_1, v_{31}][x_1, v_{41}][x_1, v_{51}] \\ = [7,3994][11,7329][21,7788][21,7788][133,4931][-0,9173]$$

Diperoleh,

$$\sum_{i=1}^9 x_i \cdot v_{1i} = x_1 \cdot v_{11} + x_2 \cdot v_{12} + x_3 \cdot v_{13} + x_4 \cdot v_{14} + x_5 \cdot v_{15} = 16,8032$$

Sedangkan operasi keluaran pada lapisan input ke lapisan tersembunyi sebagai berikut.

$$z_{in5} = v_{50} + \sum_{i=1}^9 x_i v_{5i} = -1,6325 + (-1,8668) = -4,5763$$

Selanjutnya menggunakan fungsi aktivasi sigmoid biner, diperoleh

$$z_5 = \frac{1}{1 + e^{-z_{in5}}} = \frac{1}{1 + e^{-(-3,4993)}} = 0,13453$$

Perhitungan pada lapisan tersembunyi ke lapisan output sebagai berikut.

$$\sum_{j=1}^5 w_{1j} \cdot z_j = 3,2320$$

Operasi keluaran pada lapisan tersembunyi ke lapisan output sebagai berikut.

$$y_{in1} = w_{10} + \sum_{j=1}^5 w_{1j} \cdot z_j = 0,84732$$

Dengan menggunakan fungsi aktivasi linear, diperoleh

$$y_t = 0,84732$$

Kemudian dinormalisasikan sehingga diperoleh

$$y_t = 1959259$$

Jadi, hasil peramalan penjualan harga saham PT Bank Rakyat Indonesia (Persero) Tbk (BBRI) pada Bulan minggu pertama Agustus 2018 adalah 1959259 saham. Peramalan untuk selanjutnya dengan cara yang sama sampai pada minggu kedua bulan September 2018 dapat dilihat pada tabel berikut.

Tabel 3. Hasil Peramalan penjualan harga saham PT Bank Rakyat Indonesia (Persero) Tbk (BBRI) minggu pertama bulan agustus sampai minggu kedua bulan September 2018

Waktu	Hasil Peramalan (dalam Ton)
Minggu ke. 2018	1959259

Mei 2018	2205390
Juni 2018	2255167
Juli 2018	2280011
Agustus 2018	2282722
September 2018	2289387

4. Simpulan

Berdasarkan hasil penelitian dan pembahasan disimpulkan bahwa model RNN tipe Elman yang terbaik untuk peramalan penjualan harga saham PT Bank Rakyat Indonesia (Persero) Tbk (BBRI) adalah model RNN tipe Elman pada algoritma pelatihan *Levenberg-Marquardt* dengan $\mu = 0,013$ diperoleh pada arsitektur jaringan 9-10-1 pada fungsi aktivasi logsig dengan MSE dan MAPE pengujian masing-masing sebesar $7,6649e+10$ dan $9,2348\%$.

Daftar Pustaka

- Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Birmingham: Oxford University Press.
- Chakraborty, K., Mehrotra, K., Mohan, C. K., & Ranka, S. (1992). Forecasting the Behavior of Multivariate Time Series Using Neural Networks. *Neural Networks* .(online). <https://cyber.sci-hub.mu/MTAuMTAxNi9zMDg5My02MDgwKDA1KTgwMDkyLTk=/chakraborty1992.pdf#view=FitH> Vol.V: 961—970. (diunduh tanggal 1 Agustus 2018).
- Connor, J. T., Martin, D. G., & Atlas, L. E. (1994). Recurrent Neural Networks and Robust Time Series Prediction”. *IEEE Transactions on Neural Networks* (online). <https://dacemirror.sci-hub.mu/journal-article/861dde49215aa7182c8b6193a6dcc20a/connor1994.pdf#view=FitH> Vol.V (2): 240—254.
- Fausett, L. (1994). *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*. New Jersey: Prentice-Hall.
- Gers F.A., Schmidhuber, J (2001). LSTM Recurrent Networks Learn Simple Context Free and Context Sensitive Languages. *IEEE Transactions on Neural Network* 12 (6), 1333-1340.
- Hermawan, Arief. (2006). *Jaringan Saraf Tiruan: Teori dan Aplikasi*. Yogyakarta: ANDI.
- Hikmah, A. (2017). Peramalan Deret Waktu Menggunakan *Autoregressive* (AR), Jaringan Syaraf Tiruan *Radial Basis Function* (RBF), dan Hibrid AR-RBF pada Inflasi Indonesia. *Skripsi*. Universitas Negeri Semarang.
- Kaastra, I., & Boyd, M. (1996). Designing a Neural Network for Forecasting Financial and Economic Time Series. *Neurocomputing* . (online). <https://cyber.sci-hub.mu/MTAuMTAxNi8wOTI1LTZMTIoOTUpMDAwMzktOQ==/kaastra1996.pdf#view=FitH> Nomor 10: 215—236. (diunduh tanggal 1 Agustus 2018).
- Kohzadi, N., Boyd, M. S., Kermanshahi, B., & Kaastra, I. (1996). A Comparison of Artificial Neural Network and Time Series Models for Forecasting Commodity Prices. *Neurocomputing* . (online). <https://cyber.scihub.mu/MTAuMTAxNi8wOTI1LTZMTIoOTUpMDAwMjAtOA==/kohzadi1996.pdf#view=FitH> Nomor 10: 169—181. (diunduh tanggal 1 Agustus 2018).
- Kontan. (2017). Permintaan Semen Nasional Tumbuh 7,8%. <http://industri.kontan.co.id/news/permintaan-semen-nasional-tumbuh-78>. (diakses tanggal 21 April 2018).
- Kuncoro, M. (2011). *Metode Kuantitatif: Teori dan Aplikasi untuk Bisnis dan Ekonomi*. Yogyakarta: UPP-STIM YKPN.
- Kurniawan, M. A., Kharis, M., & Sugiharti, E. (2017). Penerapan Metode Feed Forward Neural Network (FFNN) Backpropagation untuk Meramalkan Harga Saham. *Unnes Journal of Mathematics*. (online). <http://skripsi.unnes.ac.id/v2/skripsi/baca/188335/302.aspx>. (diunduh tanggal 17 Juli 2018).

- Kusumadewi, F. (2014). Peramalan Harga Emas Menggunakan *Feedforward Neural Networks* Dengan Algoritma *Backpropagation*. *Skripsi*. Universitas Negeri Yogyakarta.
- Kusumadewi, S. (2004). *Membangun Jaringan Syaraf Tiruan: menggunakan MATLAB & Excel Link*. Yogyakarta: Graha Ilmu.
- Makridakis, S., Wheelwright, S. C., & McGee, V.E. (1999). *Metode dan Aplikasi Peramalan*. Edisi 2 Jilid 1. Terjemahan. Untung Sus Adriyanto dan Abdul Basith. Jakarta: Erlangga.
- Misriati, T. (2016). Peramalan Jumlah Kunjungan Wisatawan Mancanegara ke Lombok Menggunakan Jaringan Syaraf Tiruan. *Jurnal Seminal Nasional Ilmu Pengetahuan dan Teknologi Komputer Nusa Mandiri*. (online). <https://konferensi.nusamandiri.ac.id/prosiding/index.php/snipstek/article/download/6/3/>. (diunduh tanggal 18 Juli 2018).
- Muis, S. (2006). *Teknik Jaringan Syaraf Tiruan*. Edisi Pertama. Yogyakarta: Graha Ilmu.
- Walid, Subanar, Dedi. R., Suhartono (2012). Recurrent Neural Network (RNN) pada Beban Listrik di Semarang. *Prosiding Seminar Nasional Matematika*. Unnes.
- Walid dkk (2015). Fractional Integrated Recurrent Neural Network (FIRNN) for Forecasting of Time Series Data with Applications in Electric Load in Java-Bali. *Contemporary Engineering Sciences*, Vol. 8, 2015, no. 31, 1485 - 1500
- Walid, Sugiman, Sunarmi, Dian Tri Wiyanti (2018), Analisis Produktivitas Kinerja Dosen dan Tenaga Kependidikan dalam Mewujudkan Tahun Reputasi Universitas Negeri Semarang (UNNES) Menggunakan Jaringan Saraf Tiruan, *PRISMA 1 journal.unnes.ac.id/sju/index.php/prisma*, vol 1, 2018, no.1 , 919-927.
- Wei, W. W. S. (1994). *Time Series Analysis: Univariate and Multivariate Method*. Canada: Addison-Wesley Publishing Company.
- Wikipedia. (2018). Semen Indonesia. http://id.wikipedia.org/wiki/Semen_Indonesia. (diakses tanggal 18 April 2018).
- Zhang, G. P. (2004). *Business Forecasting with Artificial Neural Networks: An Overview*. Hershey, PA: Idea Group Publishing.
- Zhang, G., & Hu, M. Y. (1998). Neural Network Forecasting of the British Pound/US Dollar Exchange Rate. *Omega* (online). <https://twin.sci-hub.tw/6545/62e451225bbfe1f3da87c81b1bd5a671/zhang1998.pdf#view=FitH> Vol. XXVI (4): 495—506. (diunduh tanggal 1 Agustus 2018).
- Zulkarnain. (2012). *Ilmu Menjual: Pendekatan Teoritis dan Kecakapan Menjual*. Yogyakarta: Graha Ilmu.