

RBF and ARIMA Combined for Time Series Forecasting

Dian Tri Wiyanti
Teknik Informatika
Universitas Semarang
Semarang, Indonesia
deedy87@gmail.com

Reza Pulungan
Jurusan Ilmu Komputer dan Elektronika FMIPA
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Gadjah Mada Yogyakarta
pulungan@ugm.ac.id

Abstrak—Permasalahan yang ada dalam dunia nyata seringkali adalah masalah yang kompleks di mana satu metode saja mungkin tidak mampu mengatasi masalah tersebut dengan baik. Dalam penelitian ini, dua buah metode, yaitu ARIMA dan RBF, akan digabungkan. Alasan penggabungan kedua model ini adalah karena adanya asumsi bahwa model tunggal tidak dapat secara total mengidentifikasi semua karakteristik dari *time series*. Pada penelitian ini dilakukan peramalan terhadap data Indeks Harga Perdagangan Besar (IHPB), data tersebut berada pada rentang tahun 2006 hingga beberapa bulan di tahun 2012, serta memiliki 6 variabel. Hasil peramalan metode ARIMA-RBF dibandingkan dengan metode ARIMA dan metode RBF secara individual. Hasil analisa menunjukkan bahwa dengan metode penggabungan ARIMA dan RBF, model yang diberikan memiliki hasil yang lebih baik dibandingkan dengan model ARIMA saja atau model RBF saja. Hal ini terlihat dalam MAPE, RMSE, dan waktu komputasi dari semua variabel yang ada pada data uji coba tersebut.

Kata kunci— *time series*; RBF; ARIMA

I. PENDAHULUAN

Analisis *time series* dan *forecasting* adalah bidang penelitian yang aktif [1]. Artinya, keakuratan dalam *time series forecasting* menjadi pokok dari proses pengambilan keputusan. Beberapa penelitian yang melakukan riset pada *time series* adalah statistik, jaringan syaraf, wavelet, dan sistem fuzzy. Komponen utama pembentuk *soft computing* seperti jaringan syaraf memiliki keunggulan dalam penyelesaian masalah yang mengandung ketidakpastian, ketidaktepatan, dan kebenaran parsial. Komponen tersebut memiliki kelebihan dan kekurangan yang berbeda. Namun hal tersebut dapat menjadi celah bagi perkembangan ilmu pengetahuan. Karena kelebihan suatu komponen dapat melengkapi kekurangan komponen lainnya. Terlebih lagi, masalah dalam dunia nyata seringkali merupakan masalah yang kompleks dan satu model mungkin tidak mampu mengatasi masalah tersebut dengan baik. Bahkan menurut [5], kelebihan dari menggabungkan beberapa model menjadi satu adalah menghasilkan ramalan dengan tingkat akurasi yang lebih baik secara rata-rata dibandingkan dengan model individu lainnya.

Permasalahan yang ditemui di dunia nyata adalah permasalahan data yang tidak stasioner atau tidak linier. Dan dibutuhkan suatu alat yang akurat dan efektif untuk meramalkan perilaku data yang tidak stasioner atau tidak linier tersebut. Namun terdapat pula permasalahan lain dalam model jaringan syaraf yang sering ditemui yaitu permasalahan *overfitting*, dimana model yang dibuat hanya menghasilkan output yang baik untuk data yang dilatih saja dan tidak untuk data yang divalidasi (data yang tidak termasuk proses *training*). Ini adalah fenomena *overfitting* yang termasuk masalah yang sering ditemui ketika menggunakan model jaringan syaraf. Di sisi lain, permasalahan tersebut dapat diminimalisir dengan pengembangan yang mengkombinasikannya dengan model lain. Kombinasi yang pernah dicobakan untuk peramalan *time series* adalah model jaringan syaraf tiruan dengan model *Auto Regressive Integrated Moving Average* (ARIMA) [4]. Ada tiga hal yang menjadi alasan penggunaan pengkombinasian model ARIMA dan *neural network* [12]. Pertama, sering kali terjadi kesulitan untuk menerapkan penggunaan model linier atau model nonlinier pada suatu permasalahan *time series*, sehingga model kombinasi ini menjadi alternatif yang lebih mudah. Kedua, dalam kenyataannya *time series* jarang yang linier atau nonlinier dan sering mengandung keduanya, dimana tidak hanya model ARIMA dan *neural network* masing-masing dapat memodelkan setiap kasusnya, sehingga pengkombinasian ini dapat digunakan untuk memodelkan *time series* yang mengandung linier dan nonlinier. Ketiga, dalam beberapa literatur peramalan menyatakan bahwa tidak ada model tunggal yang terbaik pada setiap situasi.

ARIMA sangat baik ketepatannya untuk peramalan jangka pendek dan untuk data *time series* non stasioner pada saat linier [8]. Sedangkan untuk data peramalan dalam periode yang cukup panjang ketepatannya kurang baik karena biasanya akan cenderung *flat* (datar/konstan). Selain itu ARIMA akan mengalami penurunan keakuratan apabila terdapat komponen nonlinier *time series* pada data pengamatan. [12] pun menyatakan bahwa ARIMA tidak mampu memodelkan *time series* yang nonlinier.

Berbeda dengan ARIMA, model jaringan syaraf (*neural network*) merupakan metode peramalan yang dapat digunakan untuk memprediksi *time series* nonlinier [11], selain itu stasioneritas dari data juga tidak dihiraukan [8]. Padahal dalam kehidupan nyata banyak permasalahan dengan data yang mengandung kelinieran dan ketidak-linieran sekaligus.

Penelitian yang dilakukan oleh [7] dan [8] telah membuktikan keberhasilan penggabungan dua metode untuk penyelesaian data linier dan nonlinier sekaligus. Terdapat beberapa algoritma pembelajaran dalam jaringan syaraf tiruan (*Artificial Neural Network*) yang dikembangkan untuk *time series forecasting*. Salah satunya yaitu menggunakan fungsi basis radial atau *Radial Basis Function Neural Network* (RBFNN), algoritma pembelajaran ini handal dan biasa digunakan untuk penyelesaian masalah *forecasting* dan *time series modelling*, selain itu RBF sangat baik ketika menyelesaikan permasalahan komponen non stasioner dan non linier. Disebut fungsi basis karena fungsi tersebut merupakan fungsi yang lengkap sehingga segala fungsi yang lain dapat diekspansikan ke dalam fungsi tersebut.

A. Tinjauan Pustaka

Beberapa penelitian telah membuktikan *performance* RBF, seperti pada [9] dimana model RBF yang optimal adalah kombinasi yang tepat antara jumlah *node* input dan jumlah *node hidden*, penelitian ini konsep teorinya dicobakan pada data *time series*. Sementara [1] meneliti suatu model dengan pendekatan *hybrid* RBFNN-AR untuk peramalan *time series* di mana pada kumpulan data dilakukan proses *smoothing* dengan teknik *binomial smoothing*, kemudian dimodelkan dengan RBFNN. Sedangkan [5], [7] dan [8] melakukan penelitian dengan penggabungan ARIMA dengan metode lain, seperti ARIMA-ANFIS [5], BP-ARIMA [7], dan ARIMA-NN [8].

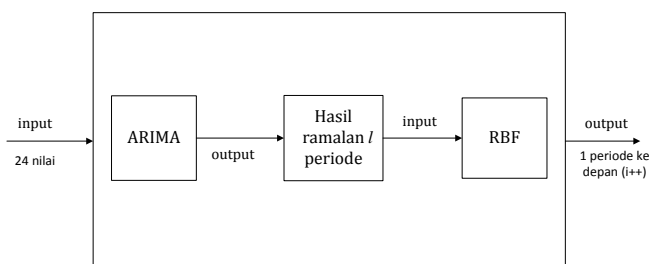
B. Kontribusi

Dalam penelitian ini, dipilih penggabungan antara metode RBFNN dengan ARIMA. Karena selain penggabungannya dapat melakukan pemodelan nonlinier maupun linier, dapat juga digunakan untuk data non stasioner, dan ARIMA dipilih karena dia tidak hanya memodelkan hubungan antara peubah terikat dengan peubah bebas yang merupakan nilai pada waktu sebelumnya, namun juga memodelkan ketergantungan peubah terikat terhadap nilai-nilai galat atau *white noise* pada waktu sebelumnya. Alasan lain adalah adanya asumsi bahwa model tunggal tidak dapat secara total mengidentifikasi semua karakteristik dari *time series* [10].

II. METODE PENELITIAN

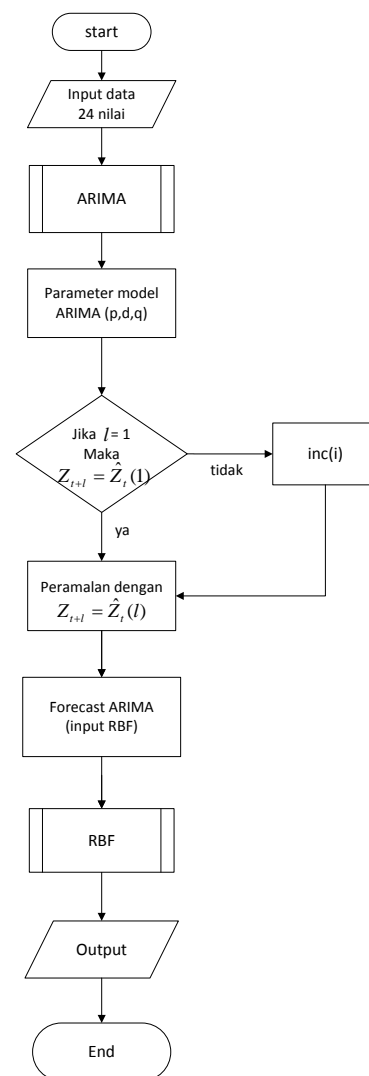
Untuk menunjang konsep teori yang telah dibahas, maka digunakan data *time series* Indeks Harga Perdagangan Besar Indonesia (IHPB) dari bulan Januari 2006–April 2012 [1]. Secara garis besar sistem peramalan ARIMA-RBF yang akan dibangun ditunjukkan dengan Gambar 1.

Sistem



Gambar 1. Arsitektur Sistem Prediksi ARIMA-RBF

Gambar 1 menggambarkan mengenai sistem prediksi dari penggabungan metode ARIMA dan RBF. Dalam sistem tersebut data berproses dalam ARIMA terlebih dahulu untuk data masukan selama 2 tahun yang artinya 24 nilai *input*, hal ini terkait dengan prosedur ARIMA yang memiliki syarat minimum jumlah masukan. Data yang keluar dari proses ARIMA adalah sebuah ramalan untuk beberapa periode ke depan. Namun, meskipun telah diproses dengan ARIMA, diasumsikan masih terdapat ketidaklinieran dalam data tersebut. Sehingga proses selanjutnya adalah menginputkan data ramalan dari ARIMA ke dalam RBF, dimana jumlah *input*nya akan dimodifikasi sedemikian sehingga didapatkan model RBF terbaik dengan nilai *error* yang minimum. Hasil *output* dari sistem adalah ramalan untuk beberapa periode ke depan yang dilakukan secara iteratif. Langkah-langkah kerja sistem ini dapat digambarkan secara lebih detail dalam diagram alir pada Gambar 2.



Gambar 2. Diagram alir proses RBF-ARIMA

Gambar 2 akan didapatkan dengan langkah-langkah sebagai berikut:

- Tujuan pertama dapat dicapai dengan memasukkan data selama 2 tahun atau 24 nilai untuk diproses dengan metode ARIMA dan menggunakan bantuan *software* statistik Minitab 14. Untuk proses ARIMA sendiri ditunjukkan oleh Gambar 3.
- Setelah didapatkan model ARIMA (p,d,q) yang terbaik maka proses selanjutnya adalah peramalan untuk beberapa periode ke depan dan dilakukan secara iteratif dimana untuk 1 periode ke depan yang dilambangkan dengan Z_{t+1} . Ukuran *window* dari hasil peramalan ARIMA bergantung pada model ARIMA (p,d,q) yang terpilih.
- Hasil peramalan ARIMA menjadi input untuk proses RBF untuk selanjutnya diproses menggunakan metode RBF.
- Output dari Gambar 2 adalah hasil peramalan yang telah melalui proses ARIMA dan RBF.

Pada tahap pengkombinasian model ARIMA-RBF ini terdapat tiga tahapan yang harus diikuti [12] dalam [8]. Pengkombinasian dilakukan karena dalam dunia nyata jarang ditemukan kejadian *time series* yang murni linier ataupun murni nonlinier. Secara umum kombinasi dari model *time series* yang memiliki struktur autokorelasi linier dan nonlinier dapat dituliskan sebagai :

$$Z_t = L_t + N_t \quad (1)$$

dimana Z_t merupakan *time series* murni, L_t merupakan komponen linier dan N_t merupakan komponen nonlinier. Tiga tahapan tersebut yaitu :

- Memodelkan komponen linier.
- Memodelkan komponen nonlinier.
- Kombinasi komponen linier dan nonlinier.

A. Memodelkan Komponen Linier

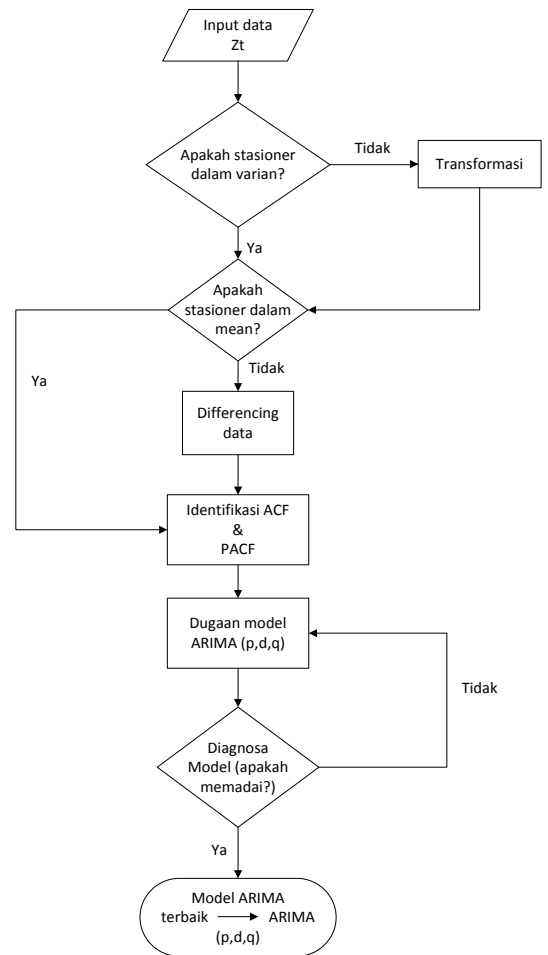
Model ARIMA digunakan pada tahapan pertama, yaitu pemodelan secara linier. Model ARIMA digunakan untuk menyelesaikan kasus yang linier, dimana residual dari model yang linier masih mengandung informasi hubungan non linier. Secara matematis dapat dituliskan sebagai :

$$e_t = Z_t - \hat{L}_t \quad (2)$$

dimana \hat{L}_t adalah nilai peramalan pada waktu t dan Z_t adalah data awal waktu ke- t . Menurut [1], karena RBF tidak mampu menangkap struktur linier dari data, residual dari kenonlinieran yang diberikan akan memberikan informasi kelinieran. Dengan memodelkan residual menggunakan model ARIMA, maka kelinieran tersebut ditemukan. Dengan model ARIMA maka residualnya adalah sebagai berikut [2] :

$$e_t = f(e_{t-1}, e_{t-2}, \dots, e_{t-n}) + \varepsilon_t \quad (3)$$

Gambaran langkah-langkah analisa ARIMA *Box Jenkins* ada pada Gambar 3.

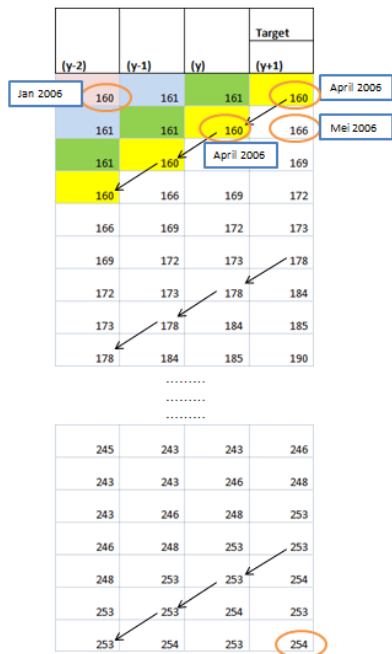


Gambar 3. Diagram alir proses ARIMA

B. Memodelkan Komponen Nonlinier

Jaringan RBF digunakan dalam tahap memodelkan komponen nonlinier, dan total jaringan syaraf ada 6 karena masing-masing untuk memprediksi komoditas yang berbeda. Langkah-langkah analisa RBF adalah:

- Input data untuk proses *training*. Data yang akan digunakan dalam proses pelatihan adalah sebarang n jumlah data yang diinputkan oleh *user*. Data diatur sedemikian sehingga membentuk matriks. Ilustrasi proses pembentukan matriks ditunjukkan oleh Gambar 4.



Gambar 4. Ilustrasi matriks

2. Menentukan jumlah hidden neuron.
3. Proses *training* dimulai dengan melakukan normalisasi data sedemikian sehingga data akan berada pada [0,1]. Rumus normalisasi yang digunakan adalah :

$$\text{data normalisasi} = \frac{\text{nilai } x - \text{nilai minimum}}{\text{nilai maksimum} - \text{nilai minimum}} \quad (4)$$

4. Langkah proses *training* selanjutnya adalah melakukan perhitungan bobot (w) antara lapisan *input* ke lapisan tersembunyi menggunakan algoritma *unsupervised learning* yaitu algoritma SOM.

Berikut langkah-langkah algoritma SOM :

- a) Inisialisasi bobot secara random, *learning rate* yang digunakan, dan parameter topologi tetangga.
- b) Selama kondisi berhenti masih salah, lakukan langkah 2-8.
- c) Untuk setiap vektor masukan x , lakukan langkah 3-5.
- d) Untuk setiap j , hitung jarak Euclidean

$$D(j) = \sum (w_{ij} - x_i)^2 \quad (5)$$

dengan w_{ij} adalah bobot dari *input* i ke *hidden* j , dan x_i adalah vektor *input* [6].

- e) Menemukan indeks J sehingga $D(J)$ minimum.
- f) Untuk setiap unit j yang termasuk di dalam tetangga dari J , dan untuk setiap i :

$$w_{ij}(t+1) = w_{ij}(t) + \alpha(t)[x_i - w_{ij}(t)] \quad (6)$$

dengan $w_{ij}(t+1)$ adalah bobot baru, $w_{ij}(t)$ adalah bobot saat ini, $\alpha(t)$ adalah laju pembelajaran.

- g) Memperbaiki nilai laju pembelajaran α .

$$\alpha_{\text{baru}} = \alpha_{\text{lama}} - (\alpha_{\text{lama}} \times \text{momentum}) \quad (7)$$

- h) Mengurangi radius tetangga topologis pada waktu yang sudah ditentukan.
- i) Tes kondisi berhenti.

5. Mencari besarnya nilai *spread* yang akan digunakan.

$$\sigma = \frac{\text{jarak maksimum antara 2 pusat}}{\sqrt{\text{banyaknya pusat}}} = \frac{d_{\text{max}}}{\sqrt{m}} \quad (8)$$

6. Perhitungan nilai aktivasi dengan fungsi *Gaussian*.

$$\phi = \exp\left(-\frac{\sum(x-w)^2}{2\sigma^2}\right) \quad (9)$$

7. Membentuk matriks *Gaussian* dari hasil perhitungan pada langkah 6.

$$G = \begin{bmatrix} \phi_{11} & \phi_{12} & \dots & \phi_{1C} \\ \phi_{21} & \phi_{22} & \dots & \phi_{2C} \\ \dots & \dots & \dots & \dots \\ \phi_{M1} & \phi_{M2} & \dots & \phi_{MC} \end{bmatrix}$$

M → vektor *input* ke -M
C → *hidden unit* ke-C

Kolom terakhir, biasanya diisi nilai bias

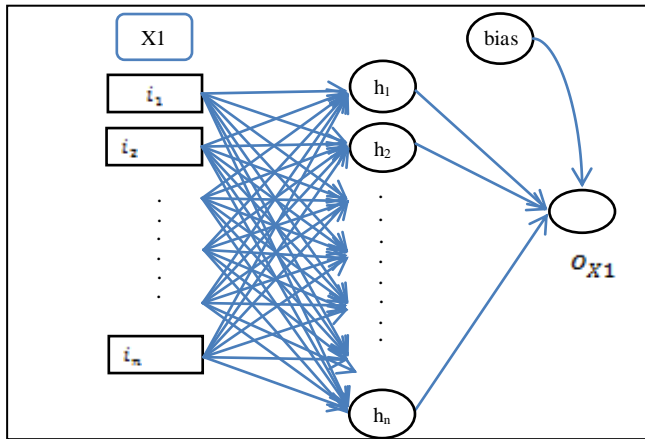
8. Menghitung *pseudo invers* dari matriks *Gaussian*. Perhitungan bobot dari lapisan tersembunyi ke lapisan output menggunakan sistem persamaan linear *pseudo invers matriks*, dimana bobot baru (w) dihitung dengan mengalikan *pseudo invers* dari matriks G , dengan vektor target (d) dari data *training*.

$$w = G^+ \cdot d = (G^T \cdot G)^{-1} \cdot G^T \cdot d \quad (10)$$

G^T adalah transpose dari matriks G dimana setiap kolom dari matriks G menjadi baris pada G^T . Sedangkan $(G^T \cdot G)^{-1}$ adalah invers dari $G^T \cdot G$ sehingga $(G^T \cdot G) \cdot (G^T \cdot G)^{-1} = I$, dengan I adalah matriks identitas [3].

9. Training data dilakukan dengan *trial and error* sedemikian sehingga didapat nilai MAPE (*Mean Absolute Percentage Error*) dan RMSE (*Root Mean Square Error*) terkecil dengan jaringan yang telah mencapai optimal.
10. Dilakukan proses testing dengan memasukkan bobot pelatihan, data pelatihan, data riil. Selanjutnya output dari proses testing adalah bobot baru, hasil peramalan RBF, serta RMSE dan MAPE.
11. Model optimal didapatkan dari perhitungan error terkecil dan tampilan grafik dimana titik-titik plot dari hasil prediksi mendekati data aktual.

Jaringan yang akan dilatih memiliki arsitektur seperti ditunjukkan dengan Gambar 5, dicontohkan untuk variabel X_1 . i_1, i_2, \dots, i_n adalah *input* sebanyak n sebarang jumlah data, h_1, h_2, \dots, h_n adalah *hidden neuron*, sedangkan O_{X_1} adalah nilai prediksi X_1 untuk 1 periode ke depan. Ukuran *window* yang digunakan dalam peramalan dengan RBF ini bergantung pada banyaknya *input*.



Gambar 5. Arsitektur RBF dengan input n -data

C. Kombinasi komponen linier dan nonlinier

Tahapan ini adalah tahap untuk memodelkan residual dari model ARIMA dengan menggunakan metode RBF. Oleh karena itu tahap ini disebut sebagai tahapan kombinasi komponen linier dan nonlinier. Hasil ramalan dari metode ARIMA kemudian dikombinasikan dengan hasil ramalan dari model RBF. Secara matematis, hasil ramalan secara keseluruhan yang diperoleh adalah sebagai berikut :

$$\hat{z}_t = \hat{L}_t + \hat{N}_t \tag{11}$$

\hat{z}_t merupakan hasil peramalan yang merupakan gabungan nilai ramalan dari model ARIMA dan nilai ramalan dari model RBF.

III. HASIL DAN PEMBAHASAN

Langkah-langkah penelitian pada peramalan data IHPB menggunakan model kombinasi ARIMA-RBF diawali dengan melakukan pemodelan ARIMA hingga diperoleh model ARIMA terbaik. Residual yang diperoleh dari model ARIMA terbaik di tiap-tiap variabel digunakan sebagai *input* pada model RBF. Langkah selanjutnya adalah melakukan peramalan dengan menggunakan model RBF. Hasil ramalan terakhir merupakan hasil peramalan menggunakan model kombinasi ARIMA-RBF. Hasil perbandingan dari ketiga metode yaitu ARIMA, RBF, dan kombinasi ARIMA-RBF ditampilkan dalam Tabel 1.

Dalam Tabel 1 diperlihatkan bahwa hasil kombinasi ARIMA-RBF lebih baik dibandingkan dengan hasil ARIMA dan RBF secara individu. Hal ini terlihat terutama pada nilai RMSE dan MAPE yang secara keseluruhan lebih kecil nilainya jika dibandingkan dengan nilai RMSE dan MAPE dari kedua model lainnya.

Sedangkan untuk waktu komputasi yaitu pada Tabel 2, apabila dilihat dari cepat tidaknya waktu komputasi, hasil kombinasi ARIMA-RBF memiliki nilai yang bagus.

TABEL 1. PERBANDINGAN NILAI ERROR DARI KETIGA MODEL

Variabel	ARIMA			RBF			ARIMA-RBF		
	Model Optimal (p,d,q)	RMSE	MAPE	Model Optimal (i,h,o)	RMSE	MAPE	Model Optimal (i,h,o)	RMSE	MAPE
1. Pertanian (X1)	(0,1,1)	2.96	0.58	(14,37,1)	6.244	0.0116	(6,8,1)	0.2084	0.000461
2. Pertambangan dan Penggalian (X2)	(1,1,1)	1.28	0.46	(7,23,1)	1.925	0.0068	(6,24,1)	0.1612	0.000397
3. Industri (X3)	(0,1,1)	1.39	0.88	(8,30,1)	1.568	0.00579	(5,12,1)	0.8832	0.002447
4. Impor (X4)	(1,1,2)	3.86	0.40	(7,24,1)	3.087	0.01175	(6,13,1)	0.0906	0.000238
5. Ekspor Non Migas (X5)	(1,0,0)	5.17	0.26	(9,14,1)	5.430	0.01354	(5,23,1)	0.6322	0.002389
6. Ekspor Migas (X6)	(1,0,1)	10.01	0.27	(6,21,1)	7.534	0.03356	(2,25,1)	2.4387	0.003761

TABEL 2. PERBANDINGAN WAKTU KOMPUTASI RBF DAN ARIMA-RBF

Variabel	RBF	ARIMA-RBF
	Waktu Komputasi (second)	Waktu Komputasi (second)
1. Pertanian (X1)	0.64	0.06
2. Pertambangan dan Penggalian (X2)	0.39	0.16
3. Industri (X3)	0.19	0.09
4. Impor (X4)	0.34	0.13
5. Ekspor Non Migas (X5)	0.23	0.19
6. Ekspor Migas (X6)	0.28	0.06

Dari Tabel 1 dan Tabel 2 dapat dikatakan bahwa semakin sedikit jumlah neuron yang digunakan maka akan semakin cepat waktu komputasinya.

IV. KESIMPULAN

Peramalan dengan menggunakan kombinasi metode ARIMA dan RBF dapat ditarik sejumlah kesimpulan bahwa:

- a. Pada proses *training*, diketahui bahwa banyaknya neuron pada *hidden layer* sangat mempengaruhi hasil prediksi, baik MAPE, RMSE maupun waktu komputasi. Hasil yang ditunjukkan memperlihatkan bahwa berbedanya jumlah *hidden neuron* pada masing-masing variabel diperlukan untuk mendapatkan arsitektur terbaiknya.
- b. Berbedanya jumlah hidden neuron pada masing-masing variabel dipengaruhi juga oleh inisialisasi bobot secara random dan algoritma SOM yang kompetitif, dimana hal tersebut yang menyebabkan perbedaan banyaknya neuron saat input dan saat pengujian, karena neuron yang tidak memiliki anggota akan langsung dieliminasi pada saat pelatihan.
- c. Model kombinasi ARIMA-RBF adalah model terbaik saat digunakan untuk data uji coba IHPB dengan kriteria RMSE, MAPE, maupun waktu komputasi.

REFERENSI

- [1] Zheng, F. and Zhong, S., 2011, "Time Series Forecasting Using A Hybrid RBF Neural Network and AR Model Based on Binomial Smoothing", World Academy of Science, Engineering and Technology, 75, 1471-1475.
- [2] Box, G. and Jenkins, G., 1976, "Time Series Analysis Forecasting and Control", Holden-Day, San Fransisco.
- [3] Broomhead, D.S. and Lowe, D., 1988, "Multivariable Functional Interpolation And Adaptive Network", Complex Sys-tems, 2, 321-355.

- [4] Faruk, D. O., 2010, A Hybrid Neural Network And ARIMA Model For Water Quality Time Series Prediction, "Engineering Applications of Artificial Intelligence", vol. 23, 586-594.
- [5] Fauziah, L. dan Suhartono., 2012. Peramalan Jumlah Kedatangan Wisatawan Mancanegara ke Indonesia Melalui Lima Pintu Kedatangan Utama Menggunakan Model hibrida ARIMA-ANFIS, Makalah Tugas Akhir, Jurusan Statistika FMIPA-ITS, Surabaya.
- [6] Fu, L., 1994. Neural Network In Computer Intelligence, McGraw-Hill International, Singapore.
- [7] Merh, N., Saxena, V.P., and Pardasani, K.R., 2010, "A Comparison Between Hybrid Approaches of ANN And ARIMA for Indian Stock Trend Forecasting", Business Intelligence Journal - July, 2, 3, 23-43.
- [8] Munarsih, E., 2011, "Penerapan Model ARIMA-Neural Network Hybrid Untuk Peramalan Time Series", Thesis, S2 Matematika FMIPA Universitas Gadjah Mada, Yogyakarta.
- [9] Sutjo, B., Subanar, dan Guritno, S., 2006, "Pemilihan Hubungan Input-Node Pada Jaringan Syaraf Fungsi Radial Basis. Berkala MIPA", 16(1), 55-61.
- [10] Terui, N. and Van Dijk, H.K., 2002, "Combined Forecast from Linear and Nonlinear Time Series Model", International Journal of Forecasting, vol.18, no.3, pp 421-438.
- [11] Zhang, G.P., Patuwo, B.E., Hu, M.Y., 2001, "A Simulation Study Of Artificial Neural Networks For Nonlinear Time-Series Forecasting", Computers & Operations Research, 28, 381-396.
- [12] Zhang, G., 2003, "Time Series Forecasting Using a Hybrid ARIMA and Neural Network Model", Journal of Neurocomputing, 50, pp.159-175.
- [13] Anonim, 2012, "Indeks Harga Perdagangan Besar Indonesia", BPS, http://www.bps.go.id/tab_sub/view.php?tabel=1&daftar=1&id_subyek=20¬ab=1, diakses 6 Mei 2012.