

JARINGAN SYARAF TIRUAN ALGORITMA BACKPROPAGATION DALAM MEMPREDIKSI KETERSEDIAAN KOMODITI PANGAN PROVINSI RIAU

Eka Pandu Cynthia¹, Edi Ismanto²

UIN Sultan Syarif Kasim Riau¹, Universitas Muhammadiyah Riau²
Jl. HR. Soebrantas KM 15 Pekanbaru¹, Jl. KH. Ahmad Dahlan Pekanbaru²
081371674646¹, 081365955794²
eka.pandu.cynthia@uin-suska.ac.id¹, edi.ismanto@umri.ac.id²

ABSTRAK

Sebuah sistem untuk prediksi ketersediaan komoditi pangan dapat membantu dalam menentukan keputusan. Jaringan Syaraf Tiruan merupakan metode yang mampu melakukan proses matematis untuk prediksi ketersediaan komoditi pangan. Dengan algoritma *Backpropagation*, dilakukan proses pengolahan data terdahulu yang dijadikan *input* untuk prediksi ketersediaan komoditi pangan. Data yang diolah sebagai variabel *input* adalah Luas Area Panen, Tingkat Produktivitas, Jumlah Produksi dan Jumlah Kebutuhan Konsumsi. Sementara komoditi pangan yang diolah adalah jenis Padi, Jagung, Kedelai, Kacang Tanah, Kacang Hijau, Ubi Kayu dan Ubi Jalar. Data-data tersebut diambil dari tahun 2006 hingga tahun 2013. Tahun 2006 hingga 2012 dijadikan data *input*, sementara untuk tahun 2013 dijadikan data *target*. Beberapa tahapan *Backpropagation* yaitu dengan inisialisasi bobot, aktivasi, menghitung bobot *input* dan *bias output* dan perubahan bobot dan *bias*. Tahapan tersebut akan diperoleh *output* yang ingin dicapai dengan pendekatan *error* terkecil sehingga di peroleh hasil prediksi ketersediaan komoditi pangan. Proses pelatihan menggunakan alat bantu perangkat lunak Matlab 6.1. Hasilnya adalah prediksi jumlah ketersediaan komoditi pangan dengan proses pelatihan dan pengujian menghasilkan *actual output* sebagai *target* yang dicapai.

Kata Kunci : Jaringan Syaraf Tiruan, *Backpropagation*, Prediksi, Komoditi Pangan.

ABSTRACT

A prediction system for availability of food commodities that can assist in determining the decision. Artificial neural network is a method that is capable of performing a mathematical process for prediction availability of food commodities. With the backpropagation algorithm, performed data processing is used as input for the previous prediction availability of food commodities. The processed data as input variables is Harvested Area, Productivity Levels, Total Production and Total Consumption Needs. While processed food commodities is kind of Rice, Corn, Soybean, Peanut, Green Beans, Cassava and Sweet Potatoes. The data was taken from 2006 to 2013, 2006 to 2012 used as input data, while for the year 2013 used as target data. Some stages of the backpropagation with weight initialization, activation, calculate the weight of input and output bias, and changes in the weights and biases. Stages will be obtained outputs to be achieved with the smallest error approach so that the obtained results predicted availability of food commodities. The training process using software tools Matlab 6.1. The result is a prediction of the amount of the availability of food commodities with the training and testing process produces the actual output targets are achieved.

Keywords : Artificial Neural Network, *Backpropagation*, Prediction, Food Commodities

I. PENDAHULUAN

1. Latar Belakang Masalah

Sebuah Jaringan Syaraf Tiruan (*Artificial Neural Network*) adalah sebuah paradigma pengolahan informasi yang terinspirasi dari sistem kerja syaraf biologis, seperti kinerja otak, yang memproses suatu informasi. Elemen kunci dari paradigma ini adalah suatu struktur baru dari sistem pengolahan informasi.

Hal ini terdiri dari sejumlah besar elemen-elemen pemrosesan yang saling berubungan (*neuron*) dan saling bekerjasama untuk pemecahan masalah-masalah tertentu (Balaji dan Baskaran, 2013).

Jaringan Syaraf Tiruan, seperti manusia, belajar dari contoh atau pola. Sebuah jaringan syaraf tiruan adalah hasil konfigurasi untuk aplikasi tertentu, seperti pengenalan pola atau

klasifikasi data, melalui suatu proses pembelajaran. Pembelajaran dalam sistem biologis menambahkan penyesuaian koneksi sinaptik yang ada diantara *neuron*. Jaringan syaraf tiruan memiliki kemampuan untuk mengekstrak hubungan antara *input* dan *output* dari sebuah proses, tanpa adanya keberadaan fisik (Devi *et al*, 2012).

Backpropagation, atau propagasi *error*, adalah metode umum dari pembelajaran Jaringan Syaraf Tiruan bagaimana menyelesaikan suatu tugas yang diberikan. Ini merupakan suatu proses pembelajaran terawasi dan merupakan implementasi dari *delta rule* (Vamsidhar *et al*, 2010). Hal ini juga menyediakan metode komputasi yang efisien untuk perubahan bobot dalam jaringan umpan maju (*feed forward*) dengan unit-unit fungsi aktivasi terdiferensial untuk pembelajaran sebuah *set* pola *input output* (Rebello *et al*, 2011).

Jaringan perambatan galat mundur (*backpropagation*) merupakan salah satu algoritma Jaringan Syaraf Tiruan yang sering digunakan untuk menyelesaikan masalah-masalah yang rumit berkaitan dengan identifikasi, prediksi, pengenalan pola dan sebagainya (Anwar, 2011).

Ketersediaan pangan yang cukup dan merata merupakan salah satu pilar perwujudan ketahanan pangan yang mantap dan berkesinambungan (Undang-Undang No 7 Tahun 1996 tentang pangan, dan penjelasan PP no 68 Tahun 2002). Tersedianya data-data tentang ketahanan pangan merupakan hal yang sangat mendasar untuk digunakan sebagai tolak ukur dalam mengestimasi dan menilai keberhasilan pembangunan ketahanan pangan serta memprediksi situasi ketahanan pangan sebagai isyarat dini untuk upaya perbaikan.

Dari gambaran wacana di atas, penulis akan mencoba mempelajari dan melakukan analisa pengujian menggunakan Jaringan Syaraf Tiruan Algoritma *Backpropagation* dalam memprediksi ketersediaan komoditi pangan Provinsi Riau.

Tujuan penulisan penelitian ini adalah :

1. Memahami konsep Jaringan Syaraf Tiruan dan proses Algoritma *Backpropagation*.

2. Menganalisa hal-hal yang berkenaan dengan ketersediaan komoditi pangan.
3. Merancang suatu arsitektur Jaringan Syaraf Tiruan untuk prediksi ketersediaan komoditi pangan.
4. Membangun suatu arsitektur Jaringan Syaraf Tiruan menggunakan Algoritma *Backpropagation* untuk prediksi ketersediaan komoditi pangan.
5. Melakukan pengujian arsitektur Jaringan Syaraf Tiruan Algoritma *Backpropagation* dalam memprediksi ketersediaan komoditi pangan.

II. STUDI PUSTAKA

1. Kecerdasan Buatan (*Artificial Intelligence*)

Kecerdasan Buatan (*Artificial Intelligence* atau *AI*) didefinisikan sebagai kecerdasan yang ditunjukkan oleh suatu entitas buatan. Sistem seperti ini umumnya dianggap komputer. Kecerdasan diciptakan dan dimasukkan ke dalam suatu mesin (komputer) agar dapat melakukan pekerjaan seperti yang dapat dilakukan manusia. Beberapa macam bidang yang menggunakan kecerdasan buatan antara lain sistem pakar, permainan komputer (*games*), logika *fuzzy*, jaringan syaraf tiruan dan robotika (Nasution, 2012).

2. Jaringan Syaraf Tiruan (*Artificial Neural Networks*)

Jaringan Syaraf Tiruan adalah merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf ini diimplemintasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran (Andrijasa dan Mistianingsih, 2010).

Pada penelitian ini akan dirancang jaringan syaraf tiruan model lapisan banyak yang mana arsitektur tipe ini memiliki satu atau lebih lapisan yang terletak di antara lapisan masukan dan lapisan keluaran, memiliki juga satu atau

lebih lapisan tersembunyi. Umumnya, ada lapisan bobot-bobot yang terletak antara dua lapisan yang bersebelahan. Jaringan dengan banyak lapisan ini dapat menyelesaikan permasalahan yang lebih sulit daripada lapisan dengan lapisan tunggal, tentu saja dengan pembelajaran yang lebih rumit juga. Gambar 1 merupakan salah satu contoh model *neuron* dengan banyak lapisan dan hanya memiliki satu lapisan tersembunyi. V_{ij} adalah bobot-bobot yang menghubungkan antara *neuron-neuron* pada lapisan masukan dengan lapisan tersembunyi. Sedangkan W_{jk} adalah bobot-bobot yang menghubungkan antara *neuron-neuron* pada lapisan tersembunyi dengan lapisan keluaran. z_{in_j} adalah hasil pengolahan data pada lapisan tersembunyi dengan fungsi aktivasi $F1$ untuk menghasilkan z_j ($j=1, \dots, k$);

$$z_{in_j} = \sum_{i=1}^n x_i \cdot w_{ij}$$

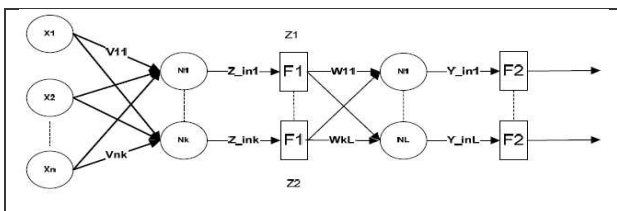
$$z_j = F1(z_{in_j})$$

Sedangkan y_{in_k} adalah hasil pengolahan data pada lapisan keluaran dengan fungsi aktivasi $F2$ untuk menghasilkan keluaran jaringan.

$$y_k; (k = 1, \dots, L)$$

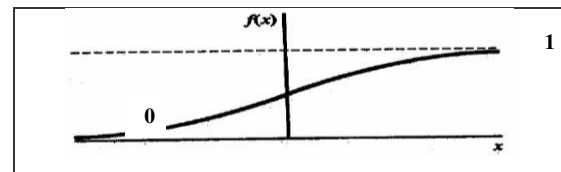
$$y_{in_k} = \sum_{j=1}^n z_j \cdot w_{jk}$$

$$z_j = F2(y_{in_k})$$



Gambar 1. Model *Neuron* Dengan Banyak Lapisan

Fungsi aktivasi yang digunakan untuk jaringan syaraf tiruan pada penelitian ini adalah fungsi *sigmoid biner*. Fungsi ini dilatih menggunakan metode *Backpropagation*. Fungsi *sigmoid biner* yang tampak pada Gambar 2 memiliki nilai antara 0 sampai 1. Karena itu, fungsi ini sering digunakan untuk jaringan syaraf yang membutuhkan nilai keluaran yang terletak pada interval 0 sampai 1. Fungsi *sigmoid biner* dirumuskan sebagai berikut :



Gambar 2. Fungsi Aktivasi *Sigmoid Biner*

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

$$y = f(x) = \sigma f(x)[1 - f(x)]$$

Untuk menormalisasikan/transformasi data dan nilai agar dapat bernilai 0 hingga 1, dapat menggunakan rumus sebagai berikut (Sya'diyah, 2011) :

$$\text{Nilai } X_{\text{baru}} = \frac{\text{Nilai } X_{\text{Lama}} - \text{Nilai } X_{\text{Minimum}}}{\text{Nilai } X_{\text{Maximum}} - \text{Nilai } X_{\text{Minimum}}}$$

3. Algoritma *Backpropagation*

Jaringan syaraf tiruan dengan *layer* tunggal memiliki keterbatasan dalam pengenalan pola. Kelemahan ini bisa ditanggulangi dengan menambahkan satu atau beberapa *layer* tersembunyi di antara *layer* masukan dan *layer* keluaran. Jaringan Syaraf Tiruan *Backpropagation* (JST-BP) melatih jaringan mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan untuk memberikan respon yang benar terhadap pola masukan yang serupa dengan pola yang dipakai selama pelatihan (Jumarwanto. *et al*, 2009).

4. Pelatihan Algoritma *Backpropagation*

Pelatihan *Backpropagation* dilakukan melalui langkah-langkah berikut ini : (Anike. *et al*, 2012).

0. Inisialisasi bobot;
1. Selama kondisi berhenti bernilai salah, kerjakan langkah 2 – 9;
2. Untuk setiap data *training*, lakukan langkah 3 – 8.
3. Untuk langkah 3 hingga 5 merupakan Proses Umpun Maju (*Feedforward*).

Setiap unit *input* ($X_i, i = 1, \dots, n$); menerima sinyal *input* dan menyebarkan sinyal tersebut ke seluruh unit tersembunyi.

4. Pada setiap unit tersembunyi ($Z_j, j = 1, \dots, p$); menjumlahkan sinyal-sinyal *input* yang sudah berbobot (termasuk bias nya)

$$z_{in_j} = v_{0j} + \sum_{i=1}^n x_i \cdot v_{ij}$$

Lalu menghitung sinyal *output* dari unit tersembunyi dengan menggunakan fungsi aktivasi yang telah ditentukan $z_j = f(z_{in_j})$. Sinyal *output* ini selanjutnya dikirim ke seluruh unit pada unit diatas (unit *output*).

5. Tiap-tiap *output* ($Y_k, k = 1, \dots, m$); menjumlahkan bobot sinyal *input* :

$$y_{in_k} = w_{0k} + \sum_{i=1}^p z_i \cdot w_{jk}$$

Lalu menghitung sinyal *output* dari unit *output* bersangkutan dengan menggunakan fungsi aktivasi yang telah ditentukan $y_k = f(y_{in_k})$. Sinyal *output* ini selanjutnya dikirim ke seluruh unit *output*.

6. Untuk langkah 6 hingga 7 merupakan Proses Umpan Mundur (*Backward*) / Propagasi *Error*. Setiap unit *output* ($Y_k, k = 1, \dots, m$); menerima suatu pola *target* yang sesuai dengan pola *input* pelatihan, untuk menghitung kesalahan (*error*) antara target dengan *output* yang dihasilkan jaringan;

$$\delta_k = (t_k - y_k) f'(y_{in_k})$$

Faktor δ_k digunakan untuk menghitung koreksi *error* (Δw_{jk}) yang nantinya akan dipakai untuk memperbaiki w_{jk} dimana :

$$\Delta w_{jk} = \alpha \delta_k z_j$$

Selain itu juga dihitung koreksi bias (Δw_{ok}) yang nantinya akan dipakai untuk memperbaiki w_{ok} dimana :

$$\Delta w_{ok} = \alpha \delta_k$$

Faktor δ_k kemudian dikirimkan ke lapisan yang berada pada langkah 7.

7. Setiap unit tersembunyi ($Z_j, j = 1, \dots, p$); menerima *input* delta (dari langkah ke-6) yang sudah berbobot :

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk}$$

Kemudian hasilnya dikalikan dengan turunan dari fungsi aktivasi yang digunakan untuk menghitung informasi kesalahan *error* δ_j dimana :

$$\delta_j = \delta_{in_j} f'(z_{in_j})$$

Kemudian hitunglah koreksi bobot dengan :

$$\Delta v_{ij} = \alpha \delta_j x_i$$

Kemudian hitunglah koreksi bias :

$$\Delta v_{0j} = \alpha \delta_j$$

8. Untuk langkah 6 hingga 7 merupakan Proses *Update* Bobot dan *Bias*. Setiap unit *output* ($Y_k, k = 1, \dots, m$); memperbaiki bobot dan bias dari seti ap unit tersembunyi ($j = 0, \dots, p$);

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk}$$

9. Tes kondisi berhenti apabila *error* ditemukan. Jika kondisi berhenti terpenuhi, maka pelatihan jaringan dapat dihentikan.

5. Pengujian *Backpropagation*

Pengujian dilakukan melalui *feedforward* dengan langkah-langkah sebagai berikut :

0. Inisialisasi bobot (hasil pelatihan);
1. Untuk setiap vektor *input*, kerjakan langkah 2 – 4;
2. Untuk $i=1, \dots, n$: set aktivasi unit *input* X_i .
3. Untuk $j=1, \dots, p$:

$$z_{in_j} = v_{0j} + \sum_{i=1}^n x_i \cdot v_{ij}$$

$$z_j = f(z_{in_j})$$

4. Untuk $k=1, \dots, p$:

$$y_{in_k} = w_{0k} + \sum_{i=1}^p z_i \cdot w_{jk}$$

$$y_k = f(y_{in_k})$$

Denormalisasi *Testing*

Setelah dilakukan proses *training* dan *testing* pola-pola yang dilatih, maka akan diperoleh hasil bahwa pengujian terhadap pola-pola

tersebut apakah telah benar/akurat atau sebaliknya. Untuk menghitung Rata-rata *Error* (RMSE) jaringan, dapat dilakukan dengan rumus (Andrijasa dan Mistianingsih, 2010) :

$$RMSE = \sum_{i=1}^n \frac{(y_i - y_n)^2}{N}$$

Di mana :

y_i = Nilai aktual data (*target*)

y_n = Nilai hasil prediksi (*actual output*)

N = Jumlah data yang diujikan

Sedangkan untuk proses denormalisasi atau mengembalikan kembali nilai hasil prediksi jaringan ke bentuk data semula (sebelum normalisasi) dapat menggunakan rumus sebagai berikut :

$$x_i = y_n(x_{max} - x_{min}) + x_{min}$$

Di mana :

x_i = Nilai X yang akan dilakukan denormalisasi

y_n = Nilai hasil prediksi (*actual output*) yang sesuai dengan x_i

x_{max} = Nilai maksimum pada barisan X

x_{min} = Nilai minimum pada barisan X

6. Prediksi / Peramalan

Teknik peramalan banyak digunakan untuk proses perencanaan dan pengambilan keputusan, suatu ramalan mencoba memperkirakan apa yang akan terjadi dan apa yang akan menjadi kebutuhan. Dalam JST terdapat teknik peramalan yang sering digunakan yaitu *backpropagation*. Teknik ini biasanya digunakan pada jaringan *multi-layer* dengan tujuan meminimalkan *error* pada keluaran yang dihasilkan oleh jaringan (Anike. *et al*, 2012).

Salah satu cara peramalan adalah dengan metode urutan waktu (*time series*) yang menggunakan data *history* (data waktu lampau), misalnya data permintaan, untuk membuat ramalan permintaan diwaktu mendatang. Tujuan dari metode ini adalah untuk mengidentifikasi pola data *history* dan kemudian mengekstrapolasikan pola ini ke masa datang. Metode *NN* ini dilatih dengan seperangkat data untuk bisa mengenal dan mengidentifikasi pola data atau kurva. Proses pelatihan disebut tahap belajar (*Learning Process*), yang merupakan bagian penting dalam metode ini. Pemilihan algoritma dan parameter yang bersesuaian dan penentuan berapa banyak perangkat data yang dibutuhkan sangat penting untuk menentukan akurasi dari peramalan yang dihasilkan (Yanti, 2011).

III. METODE

1. Analisis Kebutuhan

Pada penelitian ini, analisa kebutuhan yang dibutuhkan analisa kebutuhan masukan, kebutuhan proses dan kebutuhan keluaran.

2. Analisis Kebutuhan Masukan

Dalam analisa kebutuhan masukan (*input*) pada penelitian ini dibutuhkan data komoditi pangan yang akan melalui proses pelatihan dan pengujian menggunakan jaringan syaraf tiruan algoritma *backpropagation*, berikut tabel data komoditi pangan yang akan digunakan.

Tabel 1. Komoditi Pangan Provinsi Riau

Tahun	Luas Area Panen (Ha)							Produktivitas (KuHa)						
	Padi	Jagung	Kedelai	Kacang Tanah	Kacang Hijau	Ubi Kayu	Ubi Jalar	Padi	Jagung	Kedelai	Kacang Tanah	Kacang Hijau	Ubi Kayu	Ubi Jalar
F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}
X_1	136.177	15.539	3.994	3.619	2.194	4.410	1.413	31,53	22,35	10,53	9,36	10,53	108,00	78,72
X_2	147.167	18.379	2.266	3.475	1.650	4.718	1.627	33,30	21,99	10,68	9,28	10,54	109,76	78,76
X_3	147.796	21.397	4.319	2.412	1.577	4.625	1.429	33,44	22,41	10,86	9,29	10,70	109,78	79,29
X_4	149.423	25.016	4.906	2.023	958	4.379	1.230	35,57	22,59	10,80	9,99	10,58	155,39	79,15
X_5	156.088	18.044	5.252	2.188	1.140	4.237	1.252	36,83	23,20	11,10	9,17	10,77	179,15	79,61

X_6	145.242	14.139	6.425	1.819	938	4.144	1.203	36,89	23,48	11,05	9,30	10,61	191,80	82,39
X_7	144.015	13.284	1.949	1.723	865	3.642	1.137	35,56	23,66	11,35	9,41	10,64	243,21	82,88
X_8	118.518	11.748	3.686	1.325	585	3.863	1.028	36,63	23,88	11,34	9,38	10,58	266,81	82,32

Tahun	Produksi (Ton)							Konsumsi (Ton)						
	Padi	Jagung	Kedelai	Kacang Tanah	Kacang Hijau	Ubi Kayu	Ubi Jalar	Padi	Jagung	Kedelai	Kacang Tanah	Kacang Hijau	Ubi Kayu	Ubi Jalar
F_1	F_{16}	F_{17}	F_{18}	F_{19}	F_{20}	F_{21}	F_{22}	F_{23}	F_{24}	F_{25}	F_{26}	F_{27}	F_{28}	F_{29}
X_1	429,380	34,728	4,205	3,387	2,311	47,586	11,123	546,418	11,175	42,465	4,718	4,966	26,820	9,685
X_2	490,087	40,410	2,419	3,225	1,739	51,784	12,814	552,270	11,294	42,920	4,768	5,019	27,107	9,788
X_3	494,260	47,959	4,689	2,240	1,688	50,772	11,330	577,028	12,194	46,287	5,137	5,396	29,214	10,533
X_4	531,429	56,521	5,298	2,020	1,014	68,046	9,736	557,359	11,916	45,232	5,020	5,273	28,549	10,294
X_5	574,864	41,862	5,830	2,007	1,228	75,904	9,967	576,475	47,115	37,249	6,097	6,097	74,276	16,629
X_6	535,788	33,197	7,100	1,692	995	79,480	9,912	611,763	52,814	56,335	6,455	6,455	78,634	17,604
X_7	512,152	31,433	4,182	1,622	920	88,577	9,424	630,878	58,512	59,701	6,404	7,532	78,921	16,629
X_8	434,151	28,052	2,211	1,243	619	103,070	8,462	655,637	59,411	50,663	6,762	7,225	82,991	18,109

3. Analisis Kebutuhan Proses

Dalam analisa kebutuhan proses pada penelitian ini dibutuhkan perangkat keras dan perangkat lunak melakukan proses pelatihan dan pengujian jaringan syaraf tiruan algoritma *backpropagation*, berikut perangkat keras dan perangkat lunak yang akan digunakan.

- a. Perangkat Keras : Laptop Axioo Pico 10 Inches
Processor : Intel Atom CPU D2500 1,86 GHz
RAM : 2 GB
Harddisk : 320 GB
- b. Perangkat Lunak
Oprating System : Windows 7 Ultimate
Aplikasi Pengujian : Matlab Versi 6.1

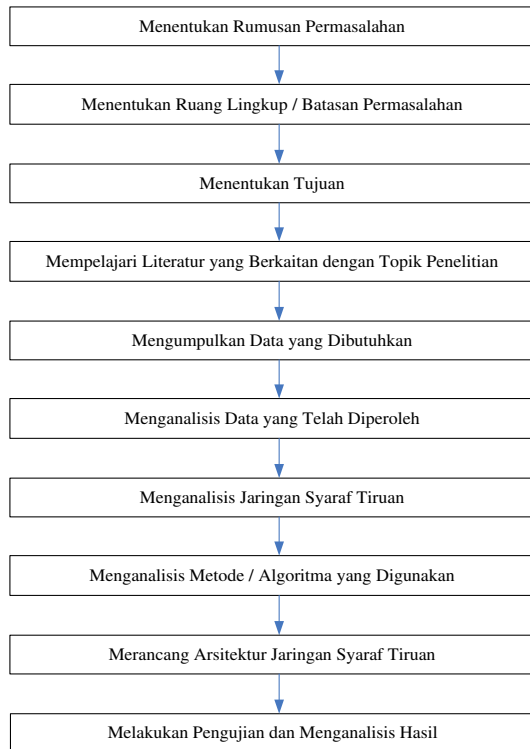
4. Analisis Kebutuhan Keluaran

Dalam analisa kebutuhan keluaran pada penelitian ini dibutuhkan proses denormalisasi data hasil keluaran dari proses pelatihan dan pengujian. Proses denormalisasi ini dimaksudkan agar data-data dapat dibaca dengan mudah kembali. Selain itu juga dibutuhkan beberapa arsitektur jaringan lain sebagai pembanding atas jaringan yang telah

dilatih dan diuji sebelumnya, dan memperoleh arsitektur jaringan terbaik dalam memprediksi ketersediaan komoditi pangan ini.

5. Kerangka Penelitian

Kerangka kerja merupakan langkah-langkah dan metode dalam melakukan penelitian. Suatu penelitian biasanya selalu dimulai dengan suatu perencanaan yang seksama yang mengikuti sejumlah petunjuk yang disusun secara sistematis sehingga hasilnya dapat mewakili kondisi yang sebenarnya dan dapat dipertanggung jawabkan. Tahapan dari penelitian ini dapat dilihat pada gambar 3.



Gambar 3. Tahapan Penelitian

IV. HASIL DAN PEMBAHASAN

1. Normalisasi Data

Data-data yang terdapat pada tabel 4.1 hingga 4.5 tidak dapat kita gunakan secara langsung pada proses pelatihan dan pengujian arsitektur Jaringan Syaraf Tiruan yang kita buat, data-data tersebut nilai nya harus kita normalisasikan terlebih dahulu. Pada penelitian ini menggunakan fungsi aktivasi *Sigmoid*

Biner, maka nilai-nilai yang akan kita gunakan pada Jaringan Syaraf Tiruan nantinya harus bernilai antara 0 hingga 1, rumus normalisasi nilai sesuai dengan Rumus 8. Dengan catatan, rujukan nilai untuk menentukan nilai *maximum* dan *minimum* adalah perkolom (masing-masing *F*) dan sesuaikan dengan jenis tabelnya, karena setiap jenis tabel memiliki satuan yang berbeda-beda.

Tabel 2. Normalisasi Data Komoditi Pangan

F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}
X_1	0,4700	0,2857	0,4569	1,0000	1,0000	0,7138	0,6427	0	0,1905	0	0,2317	0	0	0
X_2	0,7625	0,4998	0,0708	0,9372	0,6619	1,0000	1,0000	0,3302	0	0,1829	0,1341	0,0417	0,0111	0,0096
X_3	0,7793	0,7272	0,5295	0,4738	0,6165	0,9136	0,6694	0,3563	0,2222	0,4024	0,1463	0,7083	0,0112	0,1370
X_4	0,8226	1,0000	0,6606	0,3043	0,2318	0,6849	0,3372	0,7537	0,3175	0,3293	1,0000	0,2083	0,2984	0,1034
X_5	1,0000	0,4745	0,7379	0,3762	0,3449	0,5530	0,3740	0,9888	0,6402	0,6951	0	1,0000	0,4480	0,2139
X_6	0,7113	0,1802	1,0000	0,2153	0,2194	0,4665	0,2922	1,0000	0,7884	0,6341	0,1585	0,3333	0,5277	0,8822
X_7	0,6787	0,1158	0	0,1735	0,1740	0	0,1820	0,7519	0,8836	1,0000	0,2927	0,4583	0,8514	1,0000
X_8	0	0	0,3881	0	0	0,2054	0	0,9515	1,0000	0,9878	0,2561	0,2083	1,0000	0,8654

F_{16}	F_{17}	F_{18}	F_{19}	F_{20}	F_{21}	F_{22}	F_{23}	F_{24}	F_{25}	F_{26}	F_{27}	F_{28}	F_{29}
0	0,2345	0,4079	1,0000	0,0013	0	0,6114	0	0	0,2323	0	0	0	0
0,4173	0,4341	0,0425	0,9244	0,0007	0,0757	1,0000	0,0536	0,0025	0,2526	0,0245	0,0207	0,0051	0,0122
0,4460	0,6993	0,5069	0,4650	0,0007	0,0574	0,6590	0,2803	0,0211	0,4025	0,2050	0,1676	0,0426	0,1007
0,7014	1,0000	0,6314	0,3624	0	0,3688	0,2927	0,1002	0,0154	0,3556	0,1477	0,1196	0,0308	0,0723
1,0000	0,4851	0,7402	0,3563	0,0002	0,5104	0,3458	0,2752	0,7451	0	0,6747	0,4408	0,8448	0,8243
0,7314	0,1807	1,0000	0,2094	1,0000	0,5748	0,3332	0,5983	0,8632	0,8501	0,8498	0,5803	0,9224	0,9401
0,5689	0,1188	0,4031	0,1768	0,9245	0,7388	0,2210	0,7733	0,9814	1,0000	0,8249	1,0000	0,9275	0,8243
0,0328	0	0	0	0,6217	1,0000	0	1,0000	1,0000	0,5975	1,0000	0,8804	1,0000	1,0000

2. Arsitektur Jaringan Syaraf Tiruan Yang Akan Dibangun

Pada penelitian ini, akan dirancang suatu arsitektur Jaringan Syaraf Tiruan sebagai berikut :

1. Arsitektur Jaringan Syaraf Tiruan yang dirancang adalah model dengan lapisan banyak.
2. Fungsi aktivasi.

Karakteristik yang harus dimiliki oleh suatu fungsi aktivasi yaitu kontinu, differensiabel dan monoton menurun. Salah satu fungsi yang sering digunakan adalah fungsi *sigmoid biner* yang memiliki *range* (0-1). *Node* pada *output layer* memiliki nilai antara 0-1.

3. *Input-an* parameter jaringan *node* pada penelitian ini adalah:

Jumlah *node input layer* = 7 *node*, yang telah diinisialisasikan sebagai

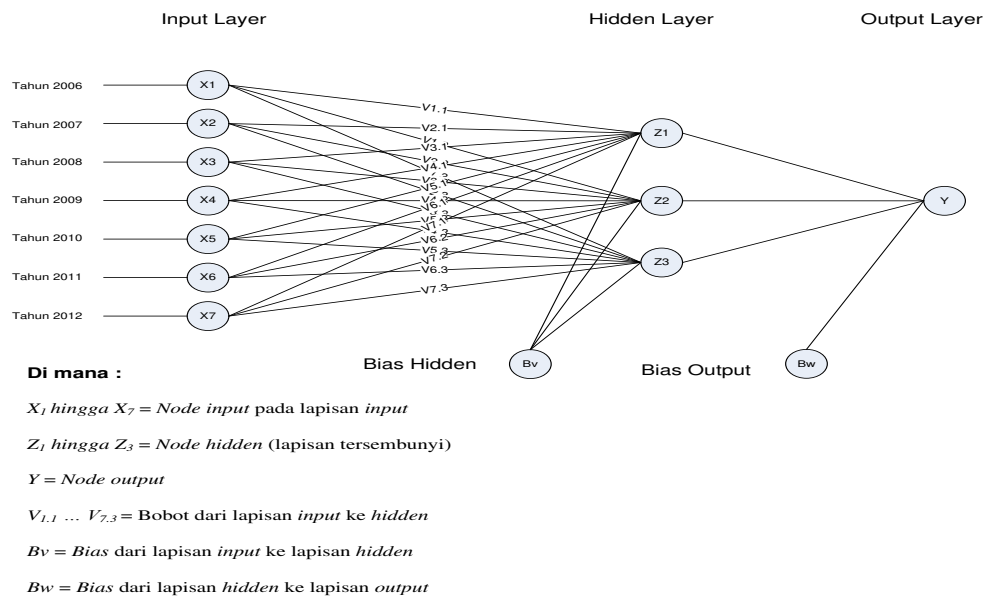
X_1 (Tahun 2006), X_2 (Tahun 2007), X_3 (Tahun 2008), X_4 (Tahun 2009), X_5 (Tahun 2010), X_6 (Tahun 2011), X_7 (Tahun 2012).

Data yang telah dinormalisasi pada Tabel 2 diatas akan ditransformasikan (normalisasikan) kembali dengan melakukan perubahan kolom untuk data *node input layer* (X_1 hingga X_7) dan X_8 sebagai data target.

Tabel 3. Normalisasi Data *Input*

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	$X_8(t)$
F_1	0	0,1429	0,2857	0,4286	0,5714	0,7143	0,8571	1
F_2	0,4700	0,7625	0,7793	0,8226	1	0,7113	0,6787	0
F_3	0,2857	0,4998	0,7272	1	0,4745	0,1802	0,1158	0
F_4	0,4569	0,0708	0,5295	0,6606	0,7379	1	0	0,3881
F_5	1	0,9372	0,4738	0,3043	0,3762	0,2153	0,1735	0
F_6	1	0,6619	0,6165	0,2318	0,3449	0,2194	0,1740	0
F_7	0,7138	1	0,9136	0,6849	0,5530	0,4665	0	0,2054
F_8	0,6427	1	0,6694	0,3372	0,3740	0,2922	0,1820	0
F_9	0	0,3302	0,3563	0,7537	0,9888	1	0,7519	0,9515
F_{10}	0,1905	0	0,2222	0,3175	0,6402	0,7884	0,8836	1
F_{11}	0	0,1829	0,4024	0,3293	0,6951	0,6341	1	0,9878
F_{12}	0,2317	0,1341	0,1463	1	0	0,1585	0,2927	0,2561
F_{13}	0	0,0417	0,7083	0,2083	1	0,3333	0,4583	0,2083
F_{14}	0	0,0111	0,0112	0,2984	0,4480	0,5277	0,8514	1
F_{15}	0	0,0096	0,1370	0,1034	0,2139	0,8822	1	0,8654
F_{16}	0	0,4173	0,4460	0,7014	1	0,7314	0,5689	0,0328
F_{17}	0,2345	0,4341	0,6993	1	0,4851	0,1807	0,1188	0
F_{18}	0,4079	0,0425	0,5069	0,6314	0,7402	1	0,4031	0
F_{19}	1	0,9244	0,4650	0,3624	0,3563	0,2094	0,1768	0
F_{20}	0,0013	0,0007	0,0007	0	0,0002	1	0,9245	0,6217
F_{21}	0	0,0757	0,0574	0,3688	0,5104	0,5748	0,7388	1
F_{22}	0,6114	1	0,6590	0,2927	0,3458	0,3332	0,2210	0
F_{23}	0	0,0536	0,2803	0,1002	0,2752	0,5983	0,7733	1
F_{24}	0	0,0025	0,0211	0,0154	0,7451	0,8632	0,9814	1
F_{25}	0,2323	0,2526	0,4025	0,3556	0	0,8501	1	0,5975
F_{26}	0	0,0245	0,2050	0,1477	0,6747	0,8498	0,8249	1
F_{27}	0	0,0207	0,1676	0,1196	0,4408	0,5803	1	0,8804
F_{28}	0	0,0051	0,0426	0,0308	0,8448	0,9224	0,9275	1
F_{29}	0	0,0122	0,1007	0,0723	0,8243	0,9401	0,8243	1

4. Arsitektur Jaringan Syaraf Tiruan menggunakan Algoritma *Backpropagation* yang dibangun pada penelitian ini dapat dilihat pada Gambar 4.



Gambar 4. Arsitektur Jaringan Syaraf Tiruan yang Akan Dibangun

3. Implementasi dan Pengujian

Implementasi dan pengujian dari jaringan syaraf tiruan dengan algoritma *backpropagation* dimulai dengan menjelaskan perangkat lunak pengujian yaitu Matlab, pemrograman jaringan syaraf tiruan, langkah pengolahan data dengan matlab dan pelatihan jaringan syaraf tiruan.

4. Implementasi Matlab

Untuk mengolah data *training* dan *testing* dengan menggunakan algoritma *backpropagation* di Matlab menggunakan tahapan-tahapan perintah sebagai berikut :

1. Membuat jaringan *backpropagation* dengan perintah :
 $net = newff(minmax(p),[3,1],\{'logsig','logsig'\},'traingdx');$
2. Melihat bobot dari *input layer* ke *hidden layer* dengan perintah :
 $net.IW\{1,1\}$
3. Melihat bobot dari *hidden layer* ke *output layer* dengan perintah :

$net.LW\{2,1\}$

4. Melihat bias dari input ke *hidden layer* dengan perintah ;
 $net.b\{1\}$
5. Melihat bias dari *hidden layer* ke *output layer* dengan perintah :
 $net.b\{2\}$
6. Melihat *output* yang dihasilkan jaringan, *error* dan *performance* dengan perintah :
 $[y,Pf,Af,e,perf]$
7. Melakukan training terhadap jaringan syaraf tiruan dengan perintah :
 $net = train(net,p,t)$

5. Pengujian

Dari beberapa pengujian arsitektur model Jaringan Syaraf Tiruan yang telah dilakukan dengan memasukkan parameter-parameter berbeda untuk *learning rate*, *goal* dan jumlah data, secara keseluruhan dapat dilihat pada Tabel 4 berikut :

Tabel 4. Pengujian Arsitektur Model Jaringan Syaraf Tiruan

No	Arsitektur	Jumlah Data	Parameter			Epoch Ditemukan	Output		RMSE	% Akurasi	Perf.
			LR	Maks. Epochs	Goal / MSE		Target	Yn			
1	Pola 7-3-1	10	0.01	100.000	0.01	206	0.8571	0.9269	0.0078036727	99.97	0.1565
	Pola 7-3-1	20	0.02	100.000	0.02	164	0.8571	0.9269	0.0097798960	99.97	0.2348
	Pola 7-3-1	29	0.03	100.000	0.03	133	0.8571	0.9269	0.0103021197	99.96	0.1799
2	Pola 7-7-1	10	0.01	100.000	0.01	245	0.8571	0.5966	0.0046332582	99.98	0.2060
	Pola 7-7-1	20	0.02	100.000	0.02	128	0.8571	0.5966	0.0047208536	99.98	0.1675
	Pola 7-7-1	29	0.03	100.000	0.03	101	0.8571	0.5966	0.0054006166	99.98	0.2013
3	Pola 7-14-1	10	0.01	100.000	0.01	274	0.8571	0.8758	0.0037859020	99.99	0.2335
	Pola 7-14-1	20	0.02	100.000	0.02	157	0.8571	0.8758	0.0043314866	99.99	0.2776
	Pola 7-14-1	29	0.03	100.000	0.03	110	0.8571	0.8758	0.0033438208	99.99	0.2185
4	Pola 7-25-1	10	0.01	100.000	0.01	125	0.8571	0.0538	0.0086997268	99.97	0.2474
	Pola 7-25-1	20	0.02	100.000	0.02	131	0.8571	0.0538	0.0103544067	99.96	0.2853
	Pola 7-25-1	29	0.03	100.000	0.03	90	0.8571	0.0538	0.0132718381	99.95	0.4008
5	Pola 7-29-1	10	0.01	100.000	0.01	159	0.8571	0.9314	0.0128796192	99.96	0.5183
	Pola 7-29-1	20	0.02	100.000	0.02	112	0.8571	0.9314	0.0101370875	99.97	0.4536
	Pola 7-29-1	29	0.03	100.000	0.03	83	0.8571	0.9314	0.0086763061	99.97	0.3545

Untuk menentukan arsitektur model jaringan syaraf tiruan terbaik yang akan digunakan dalam memprediksi, pada penelitian ini mengambil kriteria bahwa arsitektur terbaik adalah yang memiliki nilai rata-rata *error* (RMSE) terkecil, dengan penemuan *goal* pada *epochs* tercepat (terkecil nilainya) dan memiliki unjuk kerja untuk tingkat kesalahan (*performance*) terkecil. Kandidat terbaik untuk kriteria diatas adalah arsitektur model 7-14-1 skenario ketiga dan arsitektur model 7-29-1 skenario ketiga. Namun dengan memperbandingkan kedua arsitektur ini dan melihat nilai yang terkecil lebih banyak dimiliki suatu arsitektur untuk tiap kriterianya, maka dipilihlah arsitektur model 7-14-1 sebagai yang terbaik dengan *epochs* 110, RMSE 0,0033438208 dan *performance* 0,2185. Solusi

pengujian yang dilakukan pada penelitian ini akan digunakan untuk memprediksi jumlah ketersediaan komoditi pangan Provinsi Riau.

6. Denormalisasi *Testing* Prediksi Ketersediaan Komoditi Pangan

Proses denormalisasi adalah suatu proses untuk mengembalikan lagi data yang telah dinormalisasi kedalam bentuk data semula menggunakan nilai hasil pelatihan dan pengujian arsitektur Jaringan Syaraf Tiruan yang paling baik (Pola 7-14-1 di atas). Pada penelitian ini, denormalisasi bertujuan agar mengetahui nilai keluaran ketersediaan komoditi pangan. Agar lebih jelasnya dapat dilihat pada Tabel 5 hingga Tabel 8 berikut ini :

Tabel 5. Hasil Prediksi Luas Area Panen Provinsi Riau Untuk Tahun 2014

Prediksi	Tahun	Luas Area Panen (Ha)						
		Padi	Jagung	Kedelai	Kacang Tanah	Kacang Hijau	Ubi Kayu	Ubi Jalar
	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8
	X_1	136.177	15.539	3.994	3.619	2.194	4.410	1.413
	X_2	147.167	18.379	2.266	3.475	1.650	4.718	1.627
	X_3	147.796	21.397	4.319	2.412	1.577	4.625	1.429
	X_4	149.423	25.016	4.906	2.023	958	4.379	1.230
	X_5	156.088	18.044	5.252	2.188	1.140	4.237	1.252
	X_6	145.242	14.139	6.425	1.819	938	4.144	1.203
	X_7	144.015	13.284	1.949	1.723	865	3.642	1.137
	X_8	118.518	11.748	3.686	1.325	585	3.863	1.028
Yn :	0,8758	0,9478	0,7995	0,3294	0,4098	0,3945	0,2543	0,6601
Min :	2006	118.518	11.748	1.949	1.325	585	3.642	1.028
Max:	2013	156.088	25.016	6.425	3.619	2.194	4.718	1.627
Hasil Prediksi		154.126 hektar	22.355 hektar	3.423,394 hektar	2.265,081 hektar	1.219,751 hektar	3.915,627 hektar	1.423,4 hektar
Kesimpulan Perbandingan Untuk Tahun Sebelumnya (2013)	2014	Naik	Naik	Turun	Naik	Naik	Naik	Naik

Tabel 6. Hasil Prediksi Produktivitas Komoditi Pangan Provinsi Riau Untuk Tahun 2014

Prediksi	Tahun	Produktivitas (KuHa)						
		Padi	Jagung	Kedelai	Kacang Tanah	Kacang Hijau	Ubi Kayu	Ubi Jalar
	F_1	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}
	X_1	31,53	22,35	10,53	9,36	10,53	108,00	78,72
	X_2	33,30	21,99	10,68	9,28	10,54	109,76	78,76
	X_3	33,44	22,41	10,86	9,29	10,70	109,78	79,29
	X_4	35,57	22,59	10,80	9,99	10,58	155,39	79,15
	X_5	36,83	23,20	11,10	9,17	10,77	179,15	79,61
	X_6	36,89	23,48	11,05	9,30	10,61	191,80	82,39
	X_7	35,56	23,66	11,35	9,41	10,64	243,21	82,88
	X_8	36,63	23,88	11,34	9,38	10,58	266,81	82,32
Yn :	0,8758	0,978	0,8733	0,8747	0,8107	0,8724	0,8131	0,8034
Min :	2006	31,53	21,99	10,53	9,17	10,53	108	78,72
Max:	2013	36,89	23,88	11,35	9,99	10,77	266,81	82,88
Hasil Prediksi		36,772 08 ku/ha	23,640 54 ku/ha	11,24725 ku/ha	9,834774 ku/ha	10,73938 ku/ha	237,1284 ku/ha	82,062 14 ku/ha
Kesimpulan Perbandingan Untuk Tahun Sebelumnya (2013)	2014	Naik	Turun	Turun	Naik	Naik	Turun	Turun

Tabel 7. Hasil Prediksi Jumlah Produksi Komoditi Pangan Provinsi Riau Untuk Tahun 2014

Prediksi	Tahun	Produksi (Ton)						
		Padi	Jagung	Kedelai	Kacang Tanah	Kacang Hijau	Ubi Kayu	Ubi Jalar
	F_1	F_{16}	F_{17}	F_{18}	F_{19}	F_{20}	F_{21}	F_{22}
	X_1	429,380	34,728	4,205	3,387	2,311	47,586	11,123
	X_2	490,087	40,410	2,419	3,225	1,739	51,784	12,814
	X_3	494,260	47,959	4,689	2,240	1,688	50,772	11,330
	X_4	531,429	56,521	5,298	2,020	1,014	68,046	9,736
	X_5	574,864	41,862	5,830	2,007	1,228	75,904	9,967
	X_6	535,788	33,197	7,100	1,692	995	79,480	9,912
	X_7	512,152	31,433	4,182	1,622	920	88,577	9,424
	X_8	434,151	28,052	2,211	1,243	619	103,070	8,462
Yn :	0,8758	0,9421	0,8183	0,865	0,3948	0,7534	0,8319	0,6738
Min :	2006	429,38	28,052	2,211	1,243	1,014	47,586	8,462
Max:	2013	574,864	56,521	7,1	3,387	995	103,07	12,814
Hasil Prediksi		566,440 5 ton	51,348 18 ton	6,439985 ton	2,089451 ton	749,8831 ton	93,74314 ton	11,394 38 ton
Kesimpulan Perbandingan Untuk Tahun Sebelumnya (2013)	2014	Naik	Naik	Naik	Naik	Naik	Turun	Naik

Tabel 8. Hasil Prediksi Jumlah Konsumsi Komoditi Pangan Provinsi Riau Untuk Tahun 2014

Prediksi	Tahun	Konsumsi (Ton)						
		Padi	Jagung	Kedelai	Kacang Tanah	Kacang Hijau	Ubi Kayu	Ubi Jalar
	F_1	F_{23}	F_{24}	F_{25}	F_{26}	F_{27}	F_{28}	F_{29}
	X_1	546,418	11,175	42,465	4,718	4,966	26,820	9,685
	X_2	552,270	11,294	42,920	4,768	5,019	27,107	9,788
	X_3	577,028	12,194	46,287	5,137	5,396	29,214	10,533
	X_4	557,359	11,916	45,232	5,020	5,273	28,549	10,294
	X_5	576,475	47,115	37,249	6,097	6,097	74,276	16,629
	X_6	611,763	52,814	56,335	6,455	6,455	78,634	17,604
	X_7	630,878	58,512	59,701	6,404	7,532	78,921	16,629
	X_8	655,637	59,411	50,663	6,762	7,225	82,991	18,109
Yn :	0,8758	0,7247	0,824	0,9285	0,8194	0,782	0,8522	0,8557
Min :	2006	546,418	11,175	37,249	4,718	4,966	26,820	9,685
Max:	2013	655,637	59,411	59,701	6,762	7,532	82,991	18,109
Hasil Prediksi		625,569 ton	50,922 ton	58,096 ton	6,393 ton	6,973 ton	74,689 ton	16,893 ton
Kesimpulan Perbandingan Untuk Tahun Sebelumnya (2013)	2014	Turun	Turun	Naik	Turun	Turun	Turun	Turun

V. KESIMPULAN

Setelah melalui tahap pengujian untuk arsitektur jaringan syaraf tiruan menggunakan algoritma *backpropagation* dalam memprediksi ketersediaan komoditi pangan

Provinsi Riau, maka didapatkan kesimpulan bahwa :

1. Algoritma *backpropagation* dapat melakukan proses prediksi, akan tetapi baik atau tidaknya nilai yang dihasilkan sangat dipengaruhi oleh penentuan

- parameter seperti besarnya *learning rate* dan jumlah *neuron* pada *hidden layer*.
2. Terdapat faktor-faktor yang mempengaruhi tingkat kebenaran prediksi pada jaringan syaraf tiruan *backpropagation* yaitu *learning rate*, *target error*, jumlah data pembelajaran dan nilai bobot yang diberikan secara acak yang pada tiap-tiap *neuron*.
 3. Penurunan *learning rate* akan membuat proses pembelajaran semakin lambat.
 4. Berdasarkan hasil pelatihan dan pengujian jaringan syaraf tiruan, model arsitektur yang memiliki nilai *RMSE* paling kecil adalah arsitektur 7-14-1 dengan nilai *error RMSE* 0,0033438208, persentase akurasi sebesar 99,99 % dan performa 0,2185. Solusi pengujian yang dilakukan pada penelitian ini bertujuan untuk memberikan prediksi jumlah ketersediaan komoditi pangan Provinsi Riau, sehingga dapat memberikan solusi bagi Dinas Tanaman Pangan dan Hortikultura Provinsi Riau dalam mengambil kebijakan untuk menyediakan komodi pangan untuk satu tahun mendatang.

DAFTAR PUSTAKA

- Abd Alkader, Shahlla. A., Al-Allaf, Omaima. N. A. (2011). "*Backpropagation Neural Network Algorithm for Forecasting Soil Temperatures Considering Many Aspects: A Comparison of Different Approaches*". The 5th International Conference on Information Technology.
- Andrijasa, M. F., dan Mistianingsih. (2010). "*Penerapan Jaringan Syaraf Tiruan Untuk Memprediksi Jumlah Pengangguran di Provinsi Kalimantan Timur Dengan Menggunakan Algoritma Pembelajaran Backpropagation*". Jurnal Informatika Mulawarman. Vol. 5. No. 1.
- Anike, Marleni., Suyoto, dan Ernawati. (2012). "*Pengembangan Jaringan Syaraf Tiruan dalam Memprediksi Jumlah Dokter Keluarga Menggunakan Backpropagation (Studi Kasus : Regional X Cabang Palu)*". Seminar Nasional Teknologi Informasi dan Komunikasi (SENTIKA).
- Anwar, Badrul. (2011). "*Penerapan Algoritma Jaringan Syaraf Tiruan Backpropagation dalam Memprediksi Tingkat Suku Bunga Bank*". Jurnal SAINTIKOM. Vol. 10. No. 2.
- Balaji, S. Arun. dan Baskaran, K. (2013). "*Design and Development of Artificial Neural Networking (ANN) System Using Sigmoid Activation Function to Predict Annual Rice Production in Tamilnadu*". International Journal of Computer Science, Engineering and Information Technology (IJCEIT). Vol. 3. No.1.
- Dahria, Muhammad. (2008). "*Kecerdasan Buatan (Artificial Intelligence)*". Jurnal SAINTIKOM. Vol. 5. No. 2.
- Desiani, Anita dan Arhami, Muhammad. (2006). "Konsep Kecerdasan Buatan". Yogyakarta : Penerbit Andi. 161-183.
- Devi, C. J., Reddy, B. S. P., Kumar, K.V., Reddy, B. M., Nayak, N.R. (2012). "*ANN Approach for Weather Prediction using Back Propagation*". International Journal of Engineering Trends and Technology. Vol. 3. Issue 1.
- Hermawan, Arif. (2006). "*Jaringan Syaraf Tiruan : Teori dan Aplikasi*". Yogyakarta : Penerbit Andi. 49-56.
- Jumarwanto, Arif. (2009). "*Aplikasi Jaringan Syaraf Tiruan Backpropagation untuk Memprediksi Penyakit THT di Rumah Sakit Mardi Rahayu Kudus*". Jurnal Teknik Elektro. Vol. 1. No. 1.
- Kusumadewi, Sri. (2010). "*Neuro-Fuzzy : Integrasi Sistem Fuzzy dan Jaringan*

- Syaraf". Yogyakarta : Graha Ilmu. 69-146.
- Kusumadewi, Sri. (2003). "*Artificial Intelligence : Teknik dan Aplikasinya*". Yogyakarta : Penerbit Andi. 203-274.
- Nasution, Helfi. (2012). "*Implementasi Logika Fuzzy pada Sistem Kecerdasan Buatan*". Jurnal ELKHA. Vol .4. No 2.
- Prahesti, Inggit. (2013). "*Implementasi Jaringan Syaraf Tiruan Algoritma Backpropagation untuk Memprediksi Curah Hujan di Yogyakarta*". Naskah Publikasi STMIK-AMIKOM Yogyakarta.
- Puspitaningrum, Diyah. (2006). "Pengantar Jaringan Syaraf Tiruan". Yogyakarta : Penerbit Andi. 125-138.
- Rebello, S., Maheshwari, U., Safreena, Souza, R. V. D. (2011). "*Backpropagation Neural Network Method for Predicting Lac Gene Structures in Streptococcus Pyogenes M Group A Streptococcus Strains*". International Journal for Biotechnology and Molecular Biology Research. Vol. 2 (4), PP. 61-72.
- Sangadji, BM. Iriansyah. (2009). "*Prediksi Perilaku Pola Pengunjung Terhadap Transaksi pada Toko Buku Gramedia Menggunakan Jaringan Syaraf Tiruan Metode Backpropagation*". Jurnal Informatika. Vol. 5. No. 2. 135-150.
- Siang, J.J. (2009). "Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab". Yogyakarta : Penerbit Andi. 97-123.
- Suyanto. (2007). "*Artificial Intelligence*". Bandung : Penerbit Informatika. 163-205.
- Sya'diyah, Zumrotus. (2011). "*Peramalan Jumlah Kendaraan Bermotor di DKI Jakarta dengan Jaringan Backpropagation*". Fakultas Keguruan dan Ilmu Pendidikan. Universitas Darussalam. Ambon.
- Vamsidhar, Enireddy., Varma, K. V. S. R. P., Rao, P. Sankara., Satapati, R. (2010). "*Prediction of Rainfall Using Backpropagation Neural Network Model*". International Journal on Computer Science and Engineering (IJCSSE). Vol. 02. No. 04. 1119-1121.
- Yanti, Novi. (2011). "*Penerapan Metode Neural Network dengan Struktur Backpropagation untuk Prediksi Stok Obat di Apotek*". Seminar Nasional Aplikasi Teknologi Informasi.