

IMPLEMENTASI JARINGAN SYARAF TIRUAN BACKPROPAGATION UNTUK MEMPREDIKSI CUACA (*STUDI KASUS : KOTA BENGKULU*)

Samuel Agus Febru Haryanto¹, Ernawati², Diah Puspitaningrum³

^{1,2,3}Program Studi Teknik Informatika, Fakultas Teknik, Universitas Bengkulu.
Jl. WR. Supratman Kandang Limun Bengkulu 38371A INDONESIA
(telp: 0736-341022; fax: 0736-341022)

¹samuel.agus14@gmail.com,
²w_ier_na@yahoo.com,
³diyahpuspitaningrum@gmail.com

Abstrak. Penelitian ini bertujuan untuk memprediksi cuaca harian di kota Bengkulu bulan januari 2014 menggunakan data suhu, kelembaban, dan curah hujan hasil pengamatan cuaca harian tahun 2008 – 2013 dari Badan Meteorologi Klimatologi dan Geofisika (BMKG) stasiun Meteorologi Fatmawati Bengkulu. Sebagai alat prediksi cuaca digunakan metode Jaringan Syaraf Tiruan algoritma Backpropagation. Eksperimen dilakukan dengan menggunakan: a) iterasi maksimum 100, 500, 1000, 1500, 10000, 15000; b) hidden layer sejumlah 1, 2, 3, 4, 5, 6, 7, 8, 9, 10; c) error maksimum sebesar 0.01 dan 0.001. hasil terbaik diperoleh menggunakan iterasi maksimum 15000, 7 hidden layer, dan error maksimum 0.001 dengan hasil aktual yang memprediksi secara tepat sebesar 32.25% dan yang memiliki jarak prediksi bergeser 1 kelas cuaca sebesar 42%. Adapun kelas cuaca adalah: 1) hujan sangat ringan (0 – 5 mm/hari), 2) hujan ringan (6 – 20 mm/hari), 3) hujan sedang (21 – 50 mm/hari), 4) hujan lebat (51 – 100 mm/hari), 5) hujan sangat lebat (>100 mm/hari).

Kata kunci : Backpropagation, *time series*, prediksi cuaca.

Abstract: *This research aims topredict daily and 15000 of maximum iteration; b) 1, 2, 3, 4, 5, 6, weather Bengkulu City in januari 2014 using data 7, 8, 9, and 10 of hidden layer; c) 0.01 and 0.001 of temperature, humidity and rainfall from daily maximum error. The best result was obtained using weather observations of Badan Meteorologi the maximum iteration 15000, 7 hidden layers, and Klimatologi dan Geofisika (BMKG) Meteorological the maximum error 0.001 with the actual results station Fatmawati Bengkulu in 2008 – 2013. As that accurately predict amounted to 32.25% and weather prediction tool used method which has a first-class range weather predictions backpropagation Neural Network Algorithm. shifted by 42%. The weather class are: 1) very light Experiments using: a) 100,500, 1000, 1500,10000, rain (0-5 mm / day), 2) a light rain (6-20 mm / day),*

3) moderate rainfall (21-50 mm / day), 4) heavy rain (51-100 mm / day), 5) very heavy rainfall (> 100 mm / day).

Keywords: Backpropagation, time series, weather forecasting.

I. PENDAHULUAN

Peramalan adalah perkiraan tentang sesuatu yang akan terjadi pada waktu mendatang yang berdasarkan pada data yang ada pada waktu sekarang dan waktu lampau (data historis) [4]. Penggunaan data *time series* diantaranya ditemukan pada data-data cuaca (meteorologi dan klimatologi), pengukuran biologi, astronomi dan financial. Pada bidang cuaca. Data *time series* diantaranya digunakan untuk merepresentasikan pergerakan nilai-nilai temperatur udara, kelembaban udara, kecepatan angin, dan curah hujan.

Cuaca adalah perilaku yang terjadi pada atmosfer (yang berhubungan dengan Suhu, Tekanan Udara, Angin, Awan, Kelembaban udara, Radiasi, Jarak Pandang/Visibility) dan berdampak langsung pada aktifitas manusia. Pada suatu wilayah, cuaca bisa berubah dalam hitungan menit, jam, hari, dan musim. Jenis-jenis cuaca meliputi hujan, panas, salju, angin, dan badai [6]. Sedangkan pola cuaca dalam jangka panjang dan dalam rentang wilayah yang cukup besar biasa di sebut dengan iklim.

Provinsi Bengkulu di bagian barat berbatasan langsung dengan Samudera Indonesia dengan garis pantai sepanjang ± 525 Km dengan dataran yang relatif sempit. Iklim Kota Bengkulu sangat dipengaruhi oleh Samudera Hindia, jika terjadi tekanan rendah di Samudera Hindia, maka Kota Bengkulu akan mengalami hujan yang lebat, bahkan bisa disertai dengan petir dan badai [2]. Curah hujan

tinggi seperti yang terjadi pada tanggal 21 Agustus 2005 yang lalu mencapai 143 mm, dan berdasarkan catatan stasiun Klimatologi Pulau Baai Bengkulu sepuluh tahun yang lalu, tepatnya tanggal 28 Agustus 1996, juga terjadi curah hujan tinggi, bahkan mencapai 151 mm.

Informasi cuaca yang disajikan dalam bentuk *time series* sangat diperlukan oleh semua orang, di antaranya untuk pertanian. Pertanian sangat bergantung pada kondisi cuaca, oleh sebab itu pemerintah harus memikirkan strategi yang sesuai dalam memperkirakan pola cuaca.

Pada penelitian prediksi cuaca ini dilakukan dengan menggunakan Jaringan Syaraf Tiruan. Secara sederhana, JST adalah sebuah alat pemodelan data statistik non-linier. JST dapat digunakan untuk memodelkan hubungan yang kompleks antara input dan output untuk menemukan pola-pola pada data [3].

Metode yang digunakan untuk menganalisa data *time series* cuaca secara teknis adalah pendekatan dengan konsep data mining. Data mining merupakan analisa data dengan tujuan menyingkap pola yang tersembunyi. Algoritma untuk menyingkap pola yang tersembunyi tersebut diantaranya *k-means*, *fuzzy c-means*, *self organizing maps* dan *subsequence matching* [8].

Maka dari itu penulis ingin membuat penelitian untuk tugas akhir dengan judul, **“Implementasi Jaringan Syaraf Tiruan Backpropagation Untuk Memprediksi Cuaca (Studi Kasus: Kota Bengkulu)”**.

II. LANDASAN TEORI

A. Peramalan

Peramalan adalah seni dan ilmu memprediksi/memperkirakan peristiwa-peristiwa yang akan terjadi dengan menggunakan data historis an memproyeksikannya ke masa depan dengan beberapa bentuk model matematis.. Peramalan/perkiraan atau pengukuran dapat dilakukan secara kualitatif maupun kuantitatif [5]. Dalam prakteknya terdapat berbagai metode peramalan antara lain :

- 1) Deret Waktu (*Time Series*)
- 2) *Exponential Smoothing* (Penghalusan Ekponensial)
- 3) *Moving average* (rata-rata bergerak)
- 4) *Mean Square Error*

B. Jaringan Syaraf Tiruan

Jaringan Saraf Tiruan (JST), adalah jaringan dari sekelompok unit pemroses kecil yang dimodelkan berdasarkan jaringan saraf manusia. [1] Jaringan syaraf tiruan dapat digunakan dalam proses data mining, dengan karakteristik jaringan syaraf tiruan. Terutama informasi tentang aturan yang terkandung dapat diprediksi dan dijelaskan dari data.

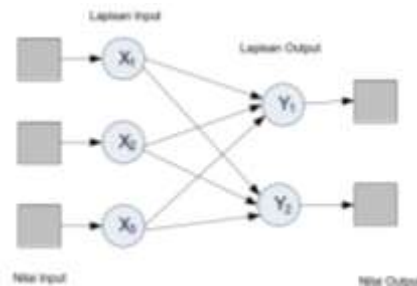
C. Arsitektur Jaringan Syaraf Tiruan

Jaringan syaraf tiruan memiliki beberapa arsitektur jaringan yang sering digunakan dalam berbagai aplikasi [1]. Arsitektur jaringan syaraf tiruan tersebut, antara lain:

1) Jaringan Layer Tunggal (*single layer network*)

Jaringan dengan lapisan tunggal terdiri dari 1 layer *input* dan 1 layer *output*. Setiap *neuron/unit* yang terdapat di dalam lapisan/*layer input* selalu terhubung dengan setiap *neural* yang terdapat pada *layer output*. Contoh algoritma jaringan syaraf tiruan

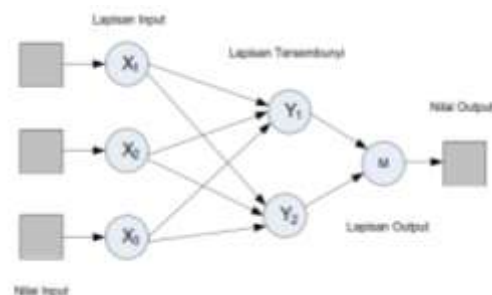
yang menggunakan metode ini yaitu : ADALINE, hofield, Perceptron. Arsitektur jaringan layer tunggal dapat dilihat pada gambar 1 berikut ini:



Gambar 1. Arsitektur Layer Tunggal [1]

2) Jaringan layer jamak (*multi layer network*)

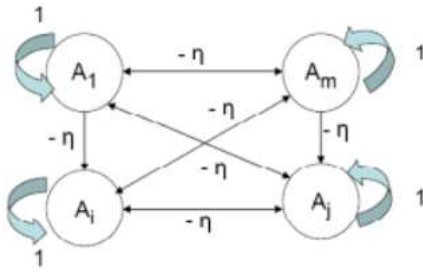
Jaringan dengan lapisan jamak memiliki ciri khas tertentu yaitu memiliki 3 jenis *layer*, yakni *layer input*, *layer output*, layer tersembunyi (*hidden layer*). Contoh algoritma jaringan syaraf tiruan yang menggunakan metode ini yaitu : MADALINE, *backpropagation*, *neocognitron*. Untuk lebih memahami jaringan layer jamak, dapat dilihat pada gambar 2 berikut ini:



Gambar 2. Arsitektur Layer Jamak [1]

3) Jaringan dengan lapisan kompetitif

Pada jaringan ini sekumpulan neuron bersaing untuk mendapatkan hak menjadi aktif. Contoh algoritma yang menggunakan metode ini adalah LVQ. Arsitektur jaringan kompetitif dapat diperhatikan pada gambar 3 di bawah ini:



Gambar 3. Arsitektur Layer Kompetitif [1]

D. Jaringan Syaraf Tiruan Backpropagation

Secara umum proses JST terbagi menjadi 2 bagian yaitu training dan testing. Struktur algoritma Backpropagation (Propagasi balik) secara lengkap adalah :

1) Inisialisasi

Tentukan angka pembelajaran (α). Tentukan pula nilai toleransi *error* yang diinginkan dan set maksimal *epoch* (iterasi) jika ingin membatasi *epoch* yang digunakan. Inisialisasi bobot dan bias sesuai dengan bobot yang dihasilkan pada proses pelatihan.

2) Step 2

Setiap *input* (X_i , $i = 1, \dots, n$) menerima sinyal *input* dan menyebarkan sinyal *input* pada seluruh lapisan tersembunyi (*hidden unit*).

3) Step 3

Setiap lapisan tersembunyi (*hidden unit*) (Z_j , $j = 1, \dots, p$) akan menghitung sinyal-sinyal *input* dengan bobot dan biasnya.

$$z_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad (1)$$

Fungsi aktivasi untuk menghitung sinyal *output*nya, yaitu;

$$z_j = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad (2)$$

Kemudian dengan menggunakan fungsi aktivasi yang telah ditentukan diperoleh sinyal *output* dari *hidden unit* tersebut dan mengirimkan sinyal tersebut ke semua unit lapisan di atasnya (lapisan *output*).

4) Step 4

Setiap unit *output* (Y_k , $k = 1, \dots, m$) akan menjumlahkan sinyal-sinyal dari lapisan tersembunyi (*hidden unit*) dengan bobot dan biasnya.

$$y_in_k = w_{0k} + \sum_{j=1}^p z_j w_{jk} \quad (3)$$

fungsi aktivasi untuk menghitung sinyal *output*nya, yaitu:

$$y_k = w_{0k} + \sum_{j=1}^p z_j w_{jk} \quad (4)$$

kemudian dengan menggunakan fungsi aktivasi yang telah ditentukan diperoleh sinyal *output* dari unit *output* tersebut.

$$y_k = f(y_in_k) \quad (5)$$

5) Step 5

Setiap unit *output* y_k menerima pola target t_k untuk menghitung error (δ_k), yaitu:

$$\delta_k = (t_k - y_k) f'(y_net_k) = (t_k - y_k) y_k (1 - y_k) \quad (6)$$

kemudian hitung nilai koreksi bobot yang nantinya digunakan untuk memperbaiki nilai bobot antara lapisan tersembunyi dan lapisan *output* (w_{jk}), yaitu :

$$\delta w_{jk} = \alpha \delta_k z_j \quad (7)$$

hitung juga koreksi bias yang digunakan untuk memperbaiki nilai bias antara lapisan tersembunyi dan lapisan *output* (w_{0k}), yaitu:

$$\delta w_{0k} = \alpha \delta_k \quad (8)$$

6) Step 6

Setiap unit pada lapisan tersembunyi (z_j , $j = 1, 2, \dots, p$) menjumlahkan sinyal-sinyal *input* dari lapisan *output*, yaitu:

$$\delta_in_j = \sum_{k=1}^m \delta_k w_{jk} \quad (9)$$

kalikan nilai ini dengan fungsi aktivasi untuk menghitung error pada lapisan tersembunyi (δ_j), yaitu:

$$\delta_j = \delta_in_j f'(z_in_j) = \delta_in_j z_j (1 - z_j) \quad (10)$$

kemudian hitung koreksi bobot untuk memperbaiki nilai bobot antara lapisan *input* dan lapisan tersembunyi (v_{ij}), yaitu :

$$\delta v_{ij} = \alpha \delta_j x_i \quad (11)$$

kemudian hitung koreksi bias untuk memperbaiki nilai bobot antara lapisan *input* dan lapisan tersembunyi (v_{oj}), yaitu:

$$\delta v_{oj} = \alpha \delta_j \quad (12)$$

7) Step 7

Tiap-tiap unit keluaran (y_k , $k=1,2, \dots, m$) memperbaiki bobot dan bias, yaitu:

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \delta w_{jk}, \quad (k=1,2, \dots, m; j=0,1, \dots, p) \quad (13)$$

tiap-tiap unit tersembunyi memperbaiki bobot dan bias, yaitu:

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \delta v_{ij}, \quad (j=1,2, \dots, p; i=0,1, \dots, n) \quad (14)$$

8) Step 8

Tes kondisi berhenti. Selama kondisi berhenti tidak terpenuhi, lakukan step ke-2 sampai ke-8.

Notasi yang digunakan dalam algoritma pengujian :

X_i = Unit *input* ke-i

Z_j = Hidden unit ke-j

Y_k = Unit *output* ke-k

T_k = pola target ke-k

V_{oj} = Bias untuk hidden unit ke-j

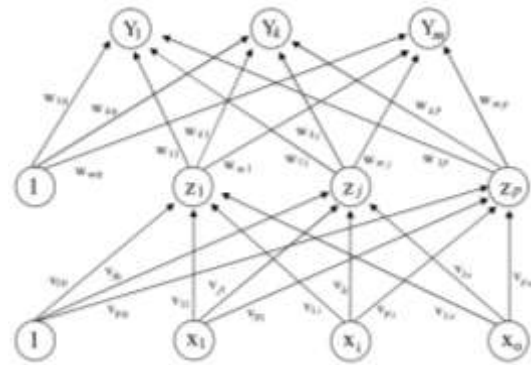
V_{ij} = Bobot antara unit *input* ke-i dengan hidden unit ke-j

W_{ok} = Bias untuk unit *output* ke-k

W_{jk} = Bobot antara hidden unit ke-j dengan unit *output* ke-k

δ_k = error [7]

Untuk lebih mengerti tentang algoritma backpropagation, dapat kita perhatikan arsitektur jaringannya pada gambar 4 di bawah ini:

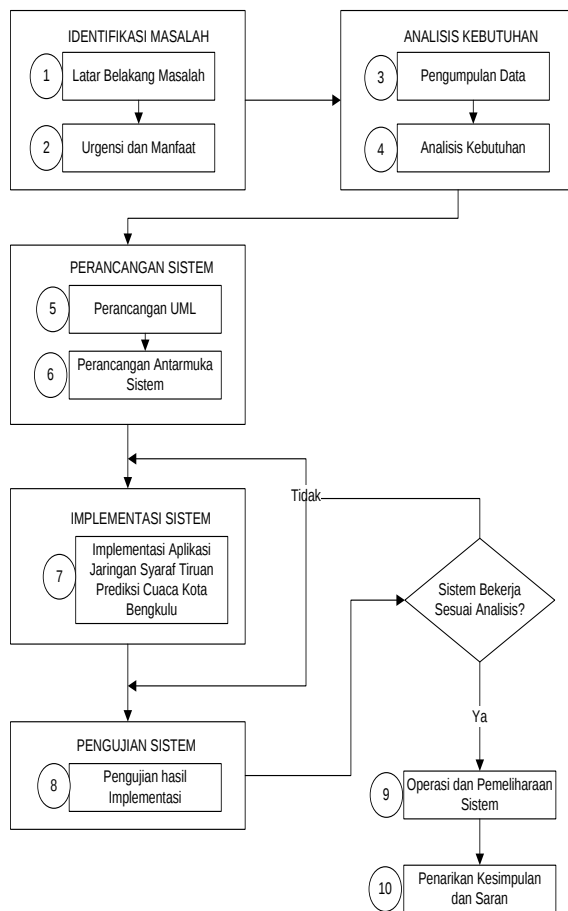


Gambar 4. Arsitektur Jaringan Backpropagation

III. METODOLOGI PENELITIAN

A. Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan pada penelitian ini adalah metode sekuensial linier yang ditunjukkan pada gambar 5 berikut ini :



Gambar 5. Diagram alir penelitian

B. Rancangan Percobaan

Pada penelitian dilakukan pengujian pada konfigurasi Jaringan Syaraf tiruan. Pengujian antara lain :

- Iterasi maksimum, dengan nilai : 100, 500, 1000, 1500, 10000, 15000.
- Hidden layer, dengan : 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
- Error maksimum dengan : 0.01, 0.001.

Sehingga dengan adanya percobaan variasi nilai pada konfigurasi Jaringan Syaraf Tiruan, bisa mendapatkan kesimpulan nilai terbaik yang sangat disarankan untuk melakukan prediksi cuaca harian. Percobaan dilakukan pada data cuaca harian bulan januari 2014. Dengan mengacu kriteria cuaca menurut Badan Meteorologi Klimatologi dan

Geofisika, jenis cuaca di bagi menjadi 5, yang ada pada tabel 1 berikut ini.

Tabel 1. Kategori Cuaca Harian menurut BMKG

No	Intensitas curah hujan (mm/hari)	Jenis hujan
1.	0 – 5	Hujan sangat ringan
2.	6 – 20	Hujan ringan
3.	21 – 50	Hujan sedang
4.	51 – 100	Hujan lebat
5.	> 100	Hujan sangat lebat

Setelah dilakukan percobaan, dicocokkan hasil prediksi tersebut dan akan di kelompokkan sesuai dengan ketepatannya, seperti pada tabel 2 berikut:

Tabel 2. Pengelompokkan Ketepatan hasil prediksi

No	Ketepatan prediksi	Keterangan
1	Aktual	Ketika hasil prediksi sama dan tepat dengan hasil pengamatan BMKG.
2	Beda 1 kelas	Ketika hasil prediksi berbeda 1 kategori/kelas cuaca dengan hasil pengamatan BMKG. Misal, hasil prediksi “Hujan Sangat Ringan” dan data hasil pengamatan BMKG “Tidak Hujan”, begitu juga sebaliknya.
3	Beda 2 kelas	Ketika hasil prediksi berbeda 2 kategori/kelas cuaca dengan hasil pengamatan BMKG. Misal, hasil prediksi “Hujan Sangat Ringan” dan data hasil pengamatan BMKG “Hujan Sedang”, begitu juga sebaliknya.
4	Beda 3 kelas	Ketika hasil prediksi berbeda 3 kategori/kelas cuaca dengan hasil pengamatan BMKG. Misal, hasil prediksi “Tidak Hujan” dan data hasil pengamatan BMKG “Hujan sedang”, begitu juga sebaliknya.
5	Beda 4 kelas	Ketika hasil prediksi berbeda 4 kategori/kelas

		cuaca dengan hasil pengamatan BMKG. Misal, hasil prediksi “Hujan sangat ringan” dan data hasil pengamatan BMKG “Hujan Lebat”, begitu juga sebaliknya.
6	Tidak ada hasil	Ketika sistem tidak mengeluarkan hasil prediksi atau ada kegagalan prediksi pada aplikasi, sehingga hasil prediksi tidak ada (-).

Dan untuk menghitung seberapa besar tingkat akurasi hasil prediksi dengan aplikasi Jaringan Syaraf Tiruan ini, digunakan perhitungan sebagai berikut:

$$\text{Akurasi} = \frac{n}{m} \times 100\%$$

Dimana:

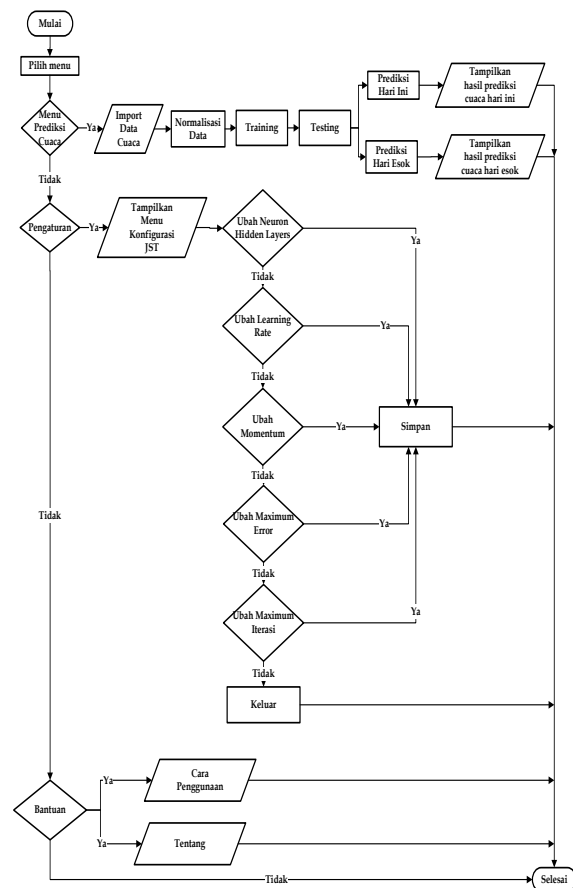
n = jumlah hari prediksi

m = jumlah total semua hari yang diprediksi

IV. ANALISIS DAN PERANCANGAN SISTEM

A. Analisis Sistem

1) Alur kerja Sistem

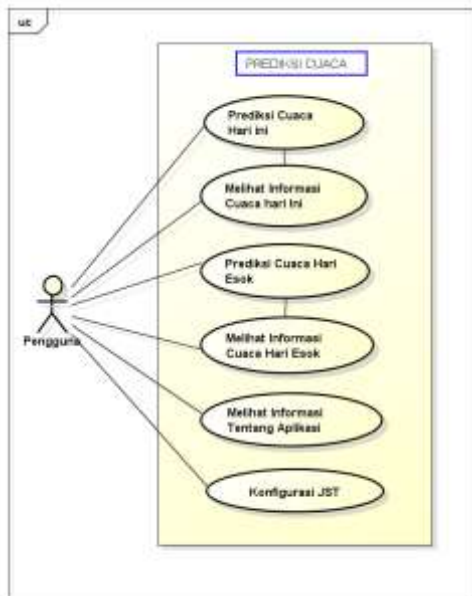


Gambar 5. Alur kerja sistem

B. Perancangan UML (Unified Modelling Language)

1. Use case diagram

Diagram ini digunakan untuk mendeskripsikan tipikal interaksi antara pengguna dengan sebuah sistem melalui sebuah cerita bagaimana sebuah sistem dipakai. *Use Case diagram* ini terdiri dari sebuah aktor dan interaksi yang dilakukannya.

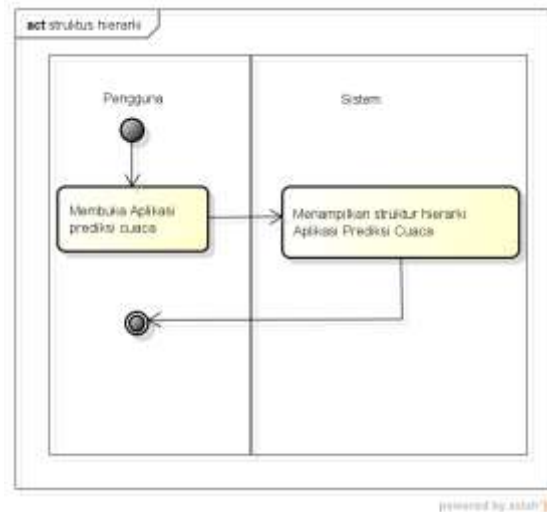


Gambar 6. Use Case Diagram

Pada gambar 6 merupakan gambaran kegiatan yang dilakukan oleh pengguna sistem (*forecaster*). Pengguna sistem dapat memilih menu yang di butuhkan. Setelah itu, pengguna dapat mengisi nilai konfigurasi Jaringan Syaraf Tiruan.

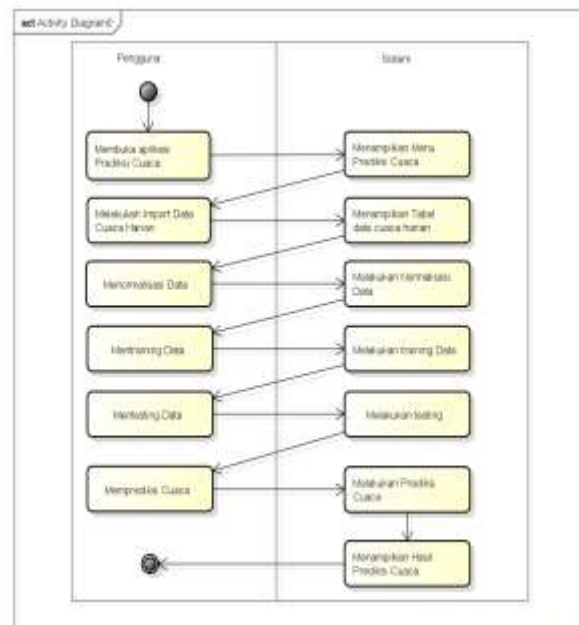
2. Activity Diagram

Activity diagram dapat digunakan untuk menjelaskan bisnis dan alur kerja operasional secara *step-by-step* dari komponen suatu sistem. *Activity diagram* menunjukkan keseluruhan dari aliran kontrol.



Gambar 7. Activity Diagram Struktur Hierarki

Gambar 7 merupakan aktivitas untuk proses dalam menampilkan tampilan struktural hierarki prediksi cuaca. Aktivitas ini dimulai dengan membuka aplikasi prediksi cuaca dan sistem akan menampilkan data prediksi cuaca.

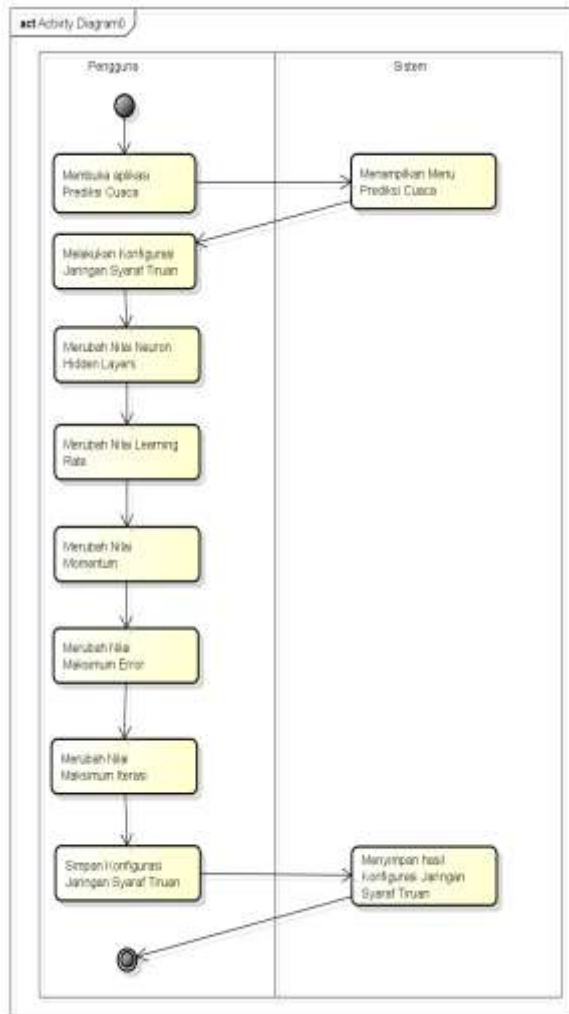


Gambar 8 Activity Diagram Prediksi Cuaca

Gambar 8 merupakan aktivitas untuk proses dalam melakukan prediksi untuk cuaca. Aktivitas ini dimulai dengan membuka aplikasi prediksi cuaca,

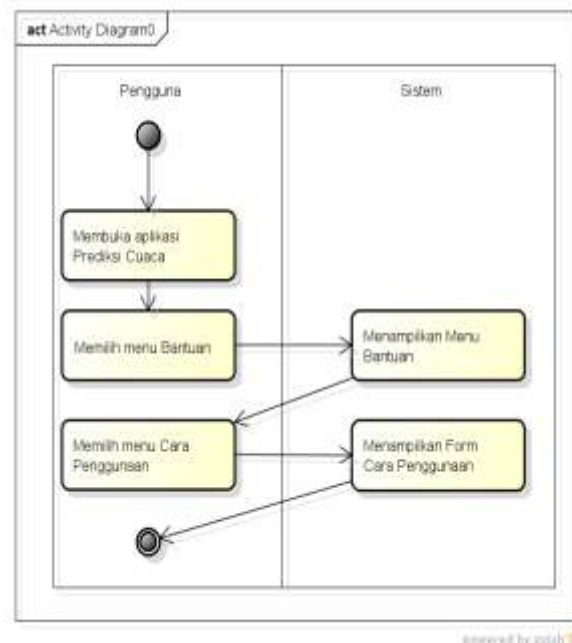
kemudian melakukan import data cuaca, selanjutnya melakukan normalisasi yang akan digunakan untuk training data. Setelah data di Normalisasi, langkah selanjutnya adalah melakukan training data, kemudian testing. Dan proses yang terakhir adalah melakukan prediksi cuaca.

rate, Momentum, Maksimum Error, dan Maksimum Iterasi. Setelah pengguna memilih button simpan, maka konfigurasi tersebut akan disimpan oleh sistem. Dan akan digunakan pada proses training data selanjutnya.



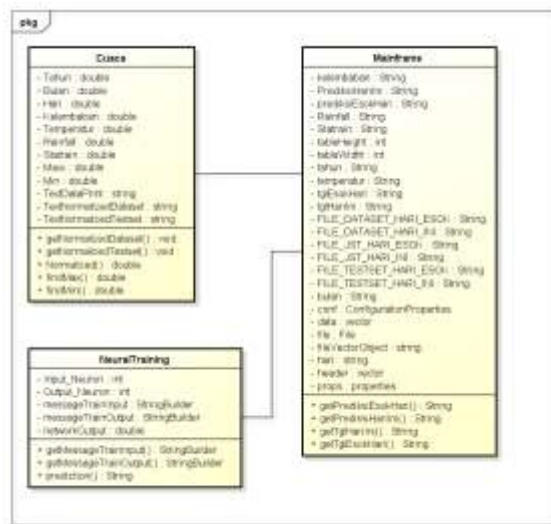
Gambar 9. Konfigurasi JST

Gambar 9 merupakan aktivitas dalam proses melakukan Konfigurasi Jaringan Syaraf Tiruan untuk *training* data. Pengguna dapat merubah konfigurasi Jaringan Syaraf Tiruan sesuai dengan kebutuhan. Dengan merubah *Neuron Hidden Layers*, *Learning*



Gambar 10. Activity Diagram Cara Penggunaan

Gambar 10 menunjukkan *activity diagram* cara penggunaan aplikasi. Setelah pengguna memilih menu cara penggunaan, lalu sistem akan menampilkan *form* cara penggunaan untuk mengetahui informasi singkat mengenai cara penggunaan sistem ini.



Gambar 13. Diagram kelas Aplikasi Prediksi

V. HASIL DAN PEMBAHASAN

A. Hasil Pengujian Aplikasi Jaringan Syaraf Tiruan

1) Pengujian Hidden Layer

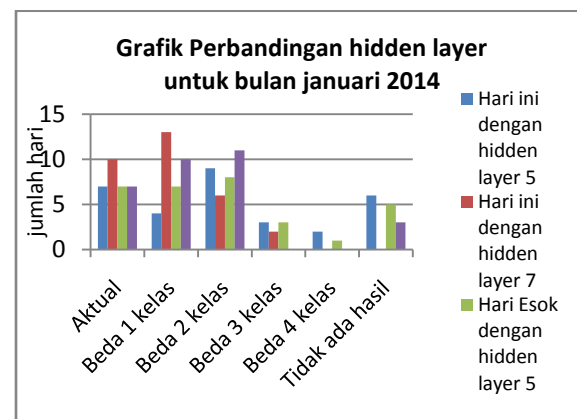
Pada tahap ini dilakukan pengujian pada *Hidden Layer* yang akan digunakan untuk mendapatkan nilai *Hidden Layer* yang paling baik.

Tabel 3. Pengujian Hidden Layer

No	Neuron hidden layers	Prediksi hari ini (1 Januari 2014)	Prediksi esok hari (2 Januari 2014)
1	1	-	-
2	2	-	-
3	3	Hujan sedang	Hujan lebat
4	4	-	Hujan lebat
5	5	Hujan ringan	-
6	6	Hujan lebat	Hujan sedang
7	7	Hujan ringan	-
8	8	Tidak hujan	Tidak hujan
9	9	Tidak hujan	Tidak hujan
10	10	-	-

Pengujian hidden Layer pada tabel 3 di atas adalah untuk mengetahui nilai *Hidden Layer* yang paling baik untuk digunakan pada proses prediksi. Dari pengujian *hidden layer* di atas dengan

menggunakan *Learning Rate* = 0.2, *Momentum* = 0.7, Maksimum error = 0.001, Maksimum Iterasi = 1000. Dan didapatkan hasil prediksi mendekati tepat adalah pada *hidden layer* 5 dan 7. Setelah itu, di ujikan kembali antara *Hidden Layer* 5 dan 7 untuk mencari *Hidden Layer* yang lebih baik, dengan memprediksi cuaca bulan januari 2014. Dengan menggunakan *Learning rate* 0.2, *Momentum* 0.7, Maksimum eror 0.001, dan Maksimum Iterasi 15000. Pada gambar 14 berikut ini merupakan hasil perbandingan hidden layer 5 dan 7 :

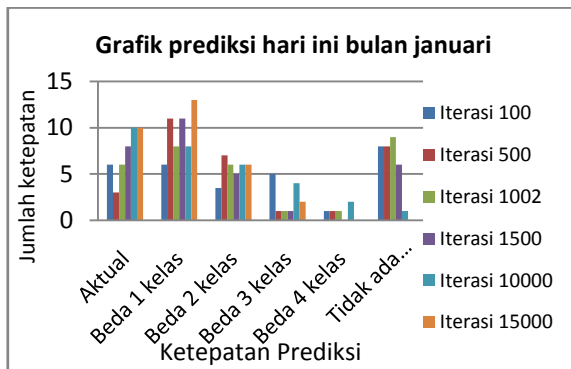
Gambar 14. Perbandingan *hidden layer*

2) Pengujian Iterasi Maksimum

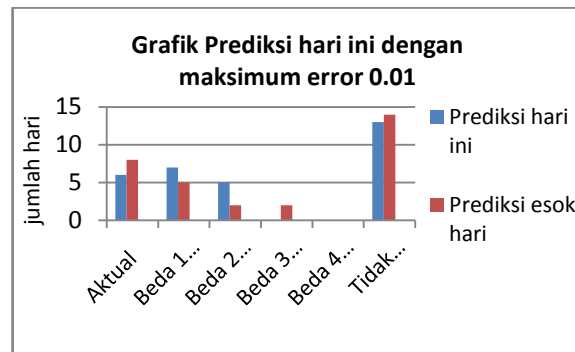
Setelah pengujian untuk penggunaan iterasi maksimum maka di dapatkan hasil pada tabel berikut ini:

Tabel 4. Pengujian Iterasi maksimum untuk prediksi cuaca hari ini

No	Hasil prediksi	Iterasi					
		100	500	1000	1500	10000	15000
1.	Aktual	6	3	6	8	10	10
2.	Beda 1 kelas	6	11	8	11	8	13
3.	Beda 2 kelas	3	7	6	5	6	6
4.	Beda 3 kelas	5	1	1	1	4	2
5.	Beda 4 kelas	1	1	1	0	2	0
6.	Tidak ada hasil	8	8	9	6	1	0



Gambar 15. Hasil prediksi hari ini



Gambar 17. Hasil Pengujian Iterasi Maksimum 0.01

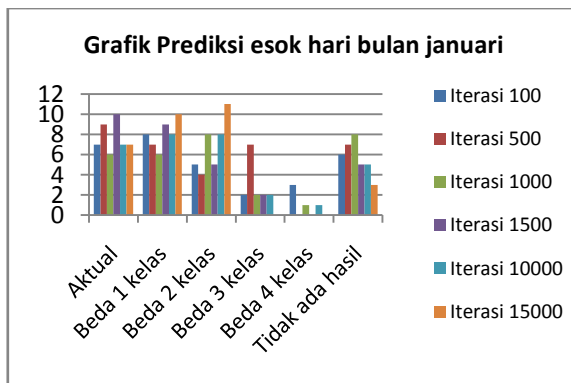
Tabel 5. Pengujian Iterasi maksimum untuk Prediksi Esok Hari

No	Hasil prediksi	Iterasi					
		100	500	1000	1500	10000	15000
1.	Aktual	7	9	6	10	7	7
2.	Beda 1 kelas	8	7	6	9	8	10
3.	Beda 2 kelas	5	4	8	5	8	10
4.	Beda 3 kelas	2	7	2	2	2	0
5.	Beda 4 kelas	3	0	1	0	1	0
6.	Tidak ada hasil	6	7	8	5	5	3

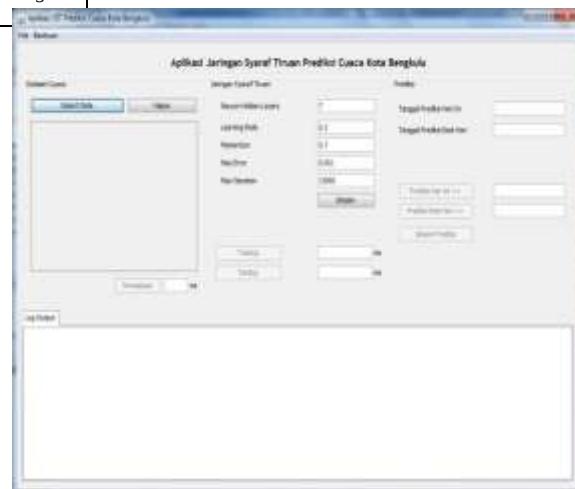
B. Hasil Implementasi Sistem

1. Aplikasi JST Prediksi Cuaca

Hasil implementasi aplikasi JST Prediksi Cuaca Kota Bengkulu dengan menggunakan Bahasa pemrograman *Java Development Kit (JDK) 7 update 55* pada *Netbeans IDE 7.4*, dapat kita perhatikan pada gambar 18 di bawah ini:



Gambar 16. Hasil prediksi esok hari



Gambar 18. Tampilan halaman utama aplikasi

3) Pengujian Maksimum Error

Pengujian untuk maksimum error 0.001 dapat kita lihat pada tabel dan grafik di atas. Dan untuk pengujian maksimum error 0.01 dapat kita lihat pada gambar berikut ini:

VI. KESIMPULAN

Dengan melakukan pengujian prediksi cuaca pada bulan januari 2014, maka didapatkan kesimpulan sebagai berikut :

1. Penelitian untuk memprediksi cuaca harian dengan Jaringan Syaraf Tiruan untuk kota Bengkulu sudah memberikan hasil yang cukup baik.

2. Percobaan untuk *hidden layer*, didapatkan *hidden layer* terbaik pada *hidden layer* 7, nilai prediksi aktual sebesar 32.2%, yaitu sebanyak 10 hari cuaca. Dan yang mengalami pergeseran 1 kelas cuaca sebanyak 13 hari atau 42%.
 3. Pada percobaan iterasi maksimum di dapatkan, bahwa semakin besar iterasi maksimum maka semakin baik hasil prediksi yang di dapatkan. Dengan hasil prediksi pada iterasi maksimum 15000 yaitu 10 hari dengan hasil aktual atau dengan akurasi sebesar 32.25%, 13 hari dengan pergeseran 1 kelas cuaca dengan akurasi sebesar 42%.
 4. Pada percobaan maksimum error 0.01 dan 0.001, di dapatkan maksimum error 0.001 memberikan hasil yang lebih baik. Untuk maksimum error 0.01 ada 6 hari yang memberikan hasil aktual atau akurasi sebesar 19.36%, 7 hari mengalami pergeseran 1 kelas atau akurasi sebesar 22.6%. Sedangkan pada maksimum error 0.001, hasil nilai prediksi aktual sebesar 32.2%, yaitu sebanyak 10 hari cuaca. Dan yang mengalami pergeseran 1 kelas cuaca sebanyak 13 hari atau 42%.
- Least Square*. Fakultas Peternakan, Universitas Diponegoro: 2009.
- [6] Kodoatie, Robert J., Roestam Sjarief. 2010. *Tata Ruang Air*. Penerbit Andi : Jogjakarta.
- [7] Setiawan, wahyudi. 2008. *Prediksi Harga Saham Menggunakan Jaringan Syaraf Tiruan Multilayer Feedforward Network Dengan Algoritma Backpropagation*. Konferensi Nasional Sistem dan Informatika: Bali.
- [8] Wong, L.P., Low, M.Y.H., Chong, S.C. 2008. *A Bee Colony Algorithm for Traveling Salesman Problem*. IEEE Computer Society.

REFERENSI

- [1] Agustin, Maria. 2012. *Penggunaan Jaringan Syaraf Tiruan Backpropagation Untuk Seleksi Penerimaan Mahasiswa Baru Pada Jurusan Teknik Computer Di Politeknik Negeri Sriwijaya*. Program Pascasarjana, Magister Sistem Informasi, Universitas Diponegoro : Semarang.
- [2] Herliana, Arif Ismul Hadi, Suwarsono. 2010. *Analisis Karakteristik Intensitas Curah Hujan Di Kota Bengkulu*. Jurusan Fisika, Fakultas MIPA, Universitas Bengkulu : Bengkulu.
- [3] Hermawan, Arief. 2006. *Jaringan Syaraf Tiruan*. Yogyakarta: Andi
- [4] Makridakis, S, dkk. 1991. *“Metode dan Aplikasi Peramalan”*. Edisi kedua. Jakarta: Erlangga.
- [5] Nasoetion, M. H. 2009. *Peramalan Populasi Ayam Buras di Jawa Tengah dengan Menggunakan Model Trend*