

**PENERAPAN MADALINE NEURAL NETWORK  
UNTUK PRAKIRAAN CUACA  
(STUDI KASUS: KOTA PONTIANAK)**

**Steven Joses<sup>1</sup>, Cucu Suhery<sup>2</sup>, Dwi Marisa Midyanti<sup>3</sup>**

<sup>1,2,3</sup>Jurusan Sistem Komputer, Fakultas MIPA Universitas Tanjungpura  
Jalan Prof Dr. H. Hadari Nawawi Pontianak  
Telp./Fax. : (0561) 577963  
e-mail: <sup>1</sup> stevenjoses96@gmail.com, <sup>2</sup>csuhery@siskom.untan.ac.id,  
<sup>3</sup>dwi.marisa@siskom.untan.ac.id

**Abstrak**

*Kondisi cuaca dapat berubah dari waktu ke waktu. Perubahan cuaca di atas permukaan bumi disebabkan oleh kondisi alam yang sedang berlangsung. Faktor-faktor yang mempengaruhi kondisi cuaca dinyatakan dalam bentuk berbagai parameter seperti suhu udara, tekanan udara, kecepatan angin, kelembaban udara, jarak pandang dan fenomena atmosfer lainnya.*

*Penelitian tentang cuaca dengan menggunakan metode MADALINE Neural Network bertujuan untuk melakukan prakiraan cuaca pada waktu yang akan datang. Many ADALINE (MADALINE) merupakan Jaringan Syaraf Tiruan dengan beberapa ADALINE sebagai penyusunnya. MADALINE akan meningkatkan kapabilitas komputasi sehingga dapat merespon dengan baik pola masukan terbaru dengan meminimalkan respon sudah dilatih untuk pola masukan sebelumnya. MADALINE menerapkan prinsip "minimum disturbance" dengan melakukan "trial adaptation" pada setiap ADALINE.*

*Pada penelitian ini diterapkan MADALINE Neural Network dengan algoritma MADALINE Rule II untuk prakiraan cuaca di Kota Pontianak. Jaringan MADALINE terdiri atas 15 unit masukan, 3 unit pada 1 lapisan tersembunyi, dan 3 unit keluaran. Pada proses pelatihan sebanyak 15115 data latih diperoleh nilai toleransi terbaik (perubahan bobot terbesar) sebesar 0,00500639 pada iterasi 65080 dan laju pembelajaran terbaik sebesar 0,005. Akurasi tertinggi sebesar 71,55% dengan 10816 data cuaca diprakirakan dengan tepat. Hasil pengujian menggunakan 6468 data uji diperoleh akurasi sebesar 78,58% dengan 5083 data cuaca diprakirakan dengan tepat.*

**Kata Kunci :** Jaringan Syaraf Tiruan, MADALINE Rule II, prakiraan cuaca

**1. PENDAHULUAN**

Kondisi cuaca adalah keadaan atmosfer di atas permukaan bumi. Kondisi cuaca dapat berubah dari waktu ke waktu. Perubahan cuaca di atas permukaan bumi disebabkan oleh kondisi alam yang sedang berlangsung. Faktor-faktor yang mempengaruhi kondisi cuaca dinyatakan dalam bentuk berbagai parameter seperti suhu udara, tekanan udara, kecepatan angin, kelembaban udara, jarak pandang dan berbagai fenomena atmosfer lainnya.

Penelitian dengan menerapkan ilmu pengetahuan dan teknologi untuk mengetahui kondisi atmosfer pada waktu dan lokasi tertentu berdasarkan parameter-parameter yang mempengaruhi cuaca disebut juga dengan

prakiraan cuaca. Prakiraan cuaca merupakan kegiatan yang masih dilakukan oleh para peneliti cuaca. Banyaknya tuntutan dari berbagai pihak yang menginginkan informasi cuaca, mendorong peneliti cuaca untuk terus melakukan penelitian-penelitian terkait kondisi cuaca dan juga mengembangkan metode-metode untuk menentukan kondisi cuaca sehingga mampu mendapatkan hasil yang baik dan akurat[1].

Penelitian mengenai estimasi kondisi cuaca di Kota Pontianak pernah dilakukan oleh Retnawati dengan menerapkan Jaringan Syaraf Tiruan Hopfield. Pada penelitian tersebut digunakan parameter berupa tekanan udara, suhu udara, arah angin dan kelembaban udara. Hasil dari penelitian tersebut berupa tingkat

ketepatan cuaca pada bulan Mei dengan kategori sangat baik (82%), Juni baik (79%), dan November baik (75%)[2].

Penelitian mengenai prakiraan cuaca juga dilakukan oleh Puspita dengan menggunakan metode Logika Fuzzy. Parameter cuaca pada penelitian tersebut meliputi suhu udara, kecepatan angin dan kelembaban udara. Hasil dari penelitian tersebut memperoleh tingkat keakuratan diatas 60% dengan menggunakan metode Sugeno[3].

*MADALINE Rule II* merupakan bagian dari *MADALINE Neural Network* dengan versi pengembangan lanjutan dari *MADALINE Rule I*. Dalam Penelitian "*MADALINE RULE II: A Training Algorithm for Neural Networks*" yang dilakukan oleh Winter dan Widrow dijelaskan bahwa *MADALINE Rule II* bertujuan untuk menghapus pembatasan atau aturan pada *MADALINE Rule I*. Hal tersebut dapat meningkatkan kekuatan jaringan untuk mewujudkan hubungan yang saling mempengaruhi antara *input-output*[4].

Berdasarkan penelitian-penelitian tersebut, maka dilakukan penelitian prakiraan cuaca dengan menggunakan metode *MADALINE Rule II*. Penggunaan metode *MADALINE Rule II* pada penelitian ini dikarenakan keluaran dari sistem lebih dari satu keluaran.

Penelitian prakiraan cuaca yang dilakukan ini menggunakan lima parameter cuaca yaitu suhu udara, tekanan udara, kelembaban udara, kecepatan angin, dan jarak pandang. Kelima parameter cuaca tersebut akan berpengaruh terhadap jenis cuaca yang diprakirakan yaitu cuaca cerah, cuaca berawan, cuaca berkabut, cuaca hujan, dan cuaca badai petir. Kelima jenis cuaca tersebut merupakan kondisi cuaca yang biasa terjadi di Kota Pontianak.

## 2. LANDASAN TEORI

### 2.1 Cuaca

Cuaca terbentuk dari gabungan unsur pembentuk cuaca seperti suhu udara, kelembaban udara, tekanan udara, kecepatan angin, jarak pandang dan unsur-unsur lainnya. Unsur-unsur pembentuk cuaca memiliki hubungan saling mempengaruhi satu dengan yang lainnya. Unsur-unsur yang mempengaruhi kondisi cuaca antara lain:

1. Suhu udara merupakan keadaan udara yang berlangsung pada waktu dan lokasi tertentu. Suhu udara ditimbulkan oleh pancaran radiasi sinar matahari yang diserap oleh permukaan bumi. Suhu yang

meningkat di permukaan bumi disebabkan karena permukaan bumi menyerap radiasi sinar matahari secara berkelanjutan. Perubahan suhu karena faktor perbedaan ketinggian jauh lebih cepat jika dibandingkan dengan perubahan suhu karena faktor perbedaan letak lintang[3].

2. Kelembaban udara adalah banyaknya jumlah kandungan uap air yang terkandung dalam udara. Nilai parameter kelembaban udara digunakan untuk menyatakan banyaknya kandungan uap air dalam udara.
3. Kecepatan angin merupakan udara yang bergerak dengan kecepatan tertentu. Secara singkat dapat dijelaskan bahwa angin adalah udara yang bergerak. Menurut Hukum Buys Ballot, angin merupakan massa udara yang bergerak menuju daerah bertekanan minimum. Gerakan massa udara secara horizontal juga dikenal dengan istilah angin.
4. Tekanan udara adalah berat dari massa udara pada suatu tempat dan waktu tertentu. Besarnya tekanan udara di suatu tempat sangat bergantung pada jumlah udara di atasnya. Pada umumnya semakin tinggi suatu ketinggian dari permukaan laut, maka kondisi tekanan udaranya semakin berkurang[5].
5. Jarak pandang (*visibility*) adalah jarak terjauh yang mampu dilihat oleh mata manusia. Jarak pandang merupakan salah satu unsur yang dapat menentukan kondisi cuaca.

Indonesia berada di garis khatulistiwa yang memiliki iklim tropis, tepatnya iklim tropis basah. Hal ini dipengaruhi juga oleh bentuk negara Indonesia yang adalah negara kepulauan yang sebagian besar daratan di Indonesia dikelilingi oleh lautan dan samudera.

### 2.2 Jaringan Syaraf Tiruan

Jaringan syaraf pada manusia merupakan kumpulan dari sel-sel syaraf atau yang disebut dengan neuron. Neuron memiliki tugas dalam mengolah informasi dan bekerja berdasarkan impuls syaraf atau rangsangan. Tiap neuron dihubungkan dengan sel syaraf lainnya. Titik temu antara terminal akson neuron yang satu dengan neuron yang lain disebut dengan sinapsis.

Pada neuron terdiri atas 3 komponen utama antara lain[6]:

1. Dendrit yaitu percabangan yang ada pada badan sel. Fungsi dari dendrit adalah untuk menerima rangsangan.
2. Badan sel yaitu badan inti pada neuron. Fungsi dari badan sel yaitu menerima rangsangan dari dendrit, kemudian meneruskannya ke akson.
3. Akson yaitu suatu bagian pada sel syaraf yang memiliki ukuran yang panjang. Fungsi dari akson yaitu menghantarkan rangsangan dari badan sel menuju ke efektor berupa otot dan kelenjar.

Neuron pada sel syaraf manusia merupakan sistem yang memiliki karakteristik *fault tolerance*. *Fault tolerance* adalah kondisi yang memungkinkan suatu sistem tetap berjalan normal meskipun terdapat kerusakan pada salah satu komponennya.

Jaringan Syaraf Tiruan adalah sebuah model pemrosesan yang diasumsikan menyerupai otak manusia dimana setiap neuron berinteraksi satu sama lain dan memindahkan informasi antar neuron[7]. Jaringan Syaraf Tiruan memiliki kemampuan untuk memilih salah satu pola yang paling mirip dalam ingatannya apabila pola yang diberikan tidak sama seperti salah satu pola dalam pembelajaran (*fault tolerance*)[8].

Jaringan Syaraf Tiruan dibentuk sebagai generalisasi dari model jaringan syaraf pada manusia, dengan asumsi bahwa[6]:

1. Pemrosesan informasi terjadi pada neuron.
2. Informasi dikirimkan antara neuron melalui penghubung-penghubung.
3. Penghubung antar neuron memiliki bobot yang dapat memperkuat ataupun memperlemah sebuah informasi.
4. Untuk menentukan keluaran, setiap neuron menggunakan fungsi aktivasi yang dikenakan pada jumlahan *input* yang diterima. Besarnya keluaran ini selanjutnya dibandingkan dengan suatu batas ambang (*threshold*). Fungsi aktivasi adalah suatu fungsi yang akan mentransformasikan suatu masukan menjadi suatu keluaran tertentu berdasarkan pemberian batas.

#### 2.2.1 Arsitektur Jaringan Syaraf Tiruan

Arsitektur Jaringan Syaraf Tiruan disebut juga sebagai rancangan arus atau jalur komunikasi antar neuron. Arsitektur Jaringan Syaraf Tiruan terbagi menjadi tiga jenis yaitu jaringan dengan lapisan tunggal dan jaringan

dengan banyak lapisan serta jaringan dengan lapisan kompetitif[6].

Jaringan dengan lapisan tunggal hanya memiliki lapisan *input* dan *output* dengan bobot-bobot terhubung. Jaringan tersebut hanya menerima *input* kemudian secara langsung akan mengolahnya menjadi *output* tanpa harus melalui lapisan tersembunyi (*hidden layer*).

Pada jaringan dengan banyak lapisan memiliki lebih dari satu lapisan tersembunyi yang terletak di antara lapisan *input* dan lapisan *output*.

Jaringan dengan lapisan kompetitif umumnya memiliki hubungan neuron yang tidak diperlihatkan pada diagram arsitektur. Model jaringan kompetitif mirip dengan jaringan lapisan tunggal maupun jaringan lapisan ganda. Perbedaannya terletak pada neuron *output* yang memberikan informasi pada unit *input* atau yang dikenal dengan istilah *feedback loop*[6].

Terdapat aturan yang digunakan dalam penentuan lapisan tersembunyi pada Jaringan Syaraf Tiruan. Ukuran *hidden layer* berada di antara ukuran *input layer* dan *output layer* sehingga tidak melebihi ukuran *input layer* maupun ukuran *output layer*[9].

#### 2.2.2 Fungsi Aktivasi pada Jaringan Syaraf Tiruan

Fungsi aktivasi pada Jaringan Syaraf Tiruan adalah suatu operasi matematika yang mentransformasi *input* menjadi *output* tertentu. Fungsi aktivasi merupakan fungsi yang menentukan nilai *output* dari suatu neuron. *MADALINE* menggunakan fungsi aktivasi dengan *threshold*[6].

Fungsi *threshold* (batas ambang)

$$f(x) = \begin{cases} 1, & x \geq a \\ -1, & x < a \end{cases} \quad (1)$$

dimana:

$f(x)$  = keluaran

$x$  = masukan

$a$  = batas ambang

Fungsi *threshold* yang digunakan adalah fungsi *threshold bipolar* yang bernilai -1 atau 1.

#### 2.2.3 ADALINE Network

*Adaptive Linear Neuron* atau biasa disebut *ADALINE* merupakan model linier Jaringan Syaraf Tiruan dengan dua lapisan yang sederhana yang diperkenalkan oleh Widrow dan Hoff[10]. Jaringan *ADALINE* tidak terdapat lapisan tersembunyi sehingga lapisan *input* terhubung langsung pada satu neuron di lapisan

*output*. Perubahan bobot yang terjadi pada *ADALINE* berfungsi untuk mengurangi *MSE* (*Mean Square Error*) antara keluaran dan target pada setiap iterasi[11]. Pada proses pelatihan *ADALINE* menggunakan algoritma *LMS* (*Least Mean Square*), atau yang dikenal dengan *Delta Rule* atau algoritma Widrow-Hoff.

#### 2.2.4 *MADALINE Network*

*Many ADALINE (MADALINE)* adalah sebuah Jaringan Syaraf Tiruan yang dibentuk dari kombinasi beberapa *ADALINE* yang terhubung satu sama lain. *MADALINE* untuk lebih dari dua unit masukan (atau lebih dari dua unit tersembunyi) dapat dibentuk secara analog[6].

Metode *MADALINE Neural Network* menggunakan prinsip “*Minimum Disturbance*” yang ditentukan oleh Widrow pada tahun 1962. Hasil keluaran dari perhitungan algoritma dimana saat keluaran tidak sesuai target maka akan terjadi perubahan bobot pada *input* yang keluarannya tidak sesuai dengan target. Perubahan bobot dapat mengurangi *error* yang ada pada pelatihan jaringan.

Penggunaan *Delta rule* atau *LMS (Least Mean Square)* pada metode ini bertujuan meminimalkan *MSE* antara fungsi aktivasi pada keluaran dengan target. Persamaan *MSE* ditunjukkan pada persamaan 2[12].

$$MSE = \frac{1}{n} \sum_{i=1}^n (f_i - y_i)^2 \quad (2)$$

dimana:

$n$  = jumlah data

$f_i$  = target ke- $i$

$y_i$  = keluaran ke- $i$

Pada *MADALINE Neural Network* terdapat 3 jenis aturan yang dikembangkan yaitu:

##### a. *MADALINE Rule I*

Pada *MADALINE Rule I*, perubahan bobot hanya terjadi pada lapisan pertama. Pada lapisan kedua, bobot-bobot yang menuju *output* diatur sedemikian rupa agar membentuk fungsi logika pada *output*. Dengan membentuk fungsi “OR” pada sebuah *output* dengan dua neuron pada lapisan tersembunyi, maka ditetapkan nilai  $b_3=v_1=v_2=0.5$ [6].

Algoritma pelatihan *MADALINE Rule I* sebagai berikut[6]:

1. Inisialisasi semua bobot dan bias dengan bilangan acak kecil. Inisialisasi laju pemahaman ( $\alpha$ ) dengan bilangan kecil.
2. Selama perubahan bobot lebih besar dari toleransi yang ditentukan (atau jumlah

epoch belum melebihi batas yang ditentukan), lakukan langkah a – e.

- a. Set aktivasi unit masukan:  $x_i = s_i$  untuk semua  $i$

dimana:

$x_i$  = unit masukan ke- $i$

$s_i$  = vektor masukan ke- $i$

$i = 0, 1, 2, \dots, n$

$n$  = baris atau kolom data

- b. Hitung net masukan untuk setiap unit tersembunyi *ADALINE* ( $z_1, z_2, \dots$ )

$$z_{in\_j} = b_j + \sum_i x_i w_{ji} \quad (3)$$

dimana:

$z_{in\_j}$  = unit tersembunyi ke- $j$

$w_{ji}$  = bobot menuju unit tersembunyi pada baris ke- $j$  dan kolom ke- $i$

$j = 0, 1, 2, \dots, n$

$n$  = baris atau kolom data

- c. Hitung keluaran setiap unit tersembunyi dengan menggunakan fungsi aktivasi *bipolar*:

$$z_j = f(z_{in\_j}) = \begin{cases} 1, & \text{jika } z_{in\_j} \geq 0 \\ -1, & \text{jika } z_{in\_j} < 0 \end{cases} \quad (4)$$

dimana:

$z_j$  = keluaran unit tersembunyi ke- $j$

- d. Tentukan keluaran jaringan

$$y_{in} = b_k + \sum_j z_j v_j \quad (5)$$

$$y = f(y_{in}) = \begin{cases} 1, & \text{jika } y_{in} \geq 0 \\ -1, & \text{jika } y_{in} < 0 \end{cases} \quad (6)$$

dimana:

$y_{in}$  = unit keluaran

$y$  = keluaran jaringan

$b_k$  = bias ke- $k$

$v_j$  = bobot menuju keluaran ke- $j$

$k = 0, 1, 2, \dots, n$

$n$  = baris atau kolom data

- e. Hitung *error* dan tentukan perubahan bobot

Jika  $y = \text{target}$ , maka tidak dilakukan perubahan bobot

Jika  $y \neq \text{target}$ :

Untuk  $t = 1$ , ubah bobot ke unit  $z_j$  yang  $z_{in}$  nya terdekat dengan 0 (misal ke unit  $z_p$ ) sebagai berikut:

$$b_{p\text{baru}} = b_{p\text{lama}} + \alpha(1 - z_{in\_p}) \quad (7)$$

$$w_{pi\text{baru}} = w_{pi\text{lama}} + \alpha(1 - z_{in\_p})x_i \quad (8)$$

dimana:

$b_{p\text{baru}}$  = bias baru ke- $p$

bp lama = bias lama ke-p  
 $\alpha$  = laju pembelajaran  
 Wpi baru = bobot baru pada baris ke-p dan kolom ke-i  
 Wpi lama = bobot lama pada baris ke-p dan kolom ke-i  
 p = 0, 1, 2, ..., n  
 n = baris atau kolom data  
 Untuk t = -1, ubah semua bobot ke unit zk yang zinya positif sebagai berikut:

$$b_k \text{ baru} = b_k \text{ lama} + \alpha(-1 - z_{in,k}) \quad (9)$$

$$w_{ki} \text{ baru} = w_{ki} \text{ lama} + \alpha(-1 - z_{in,k})x_i \quad (10)$$

dimana:

$b_k$  baru = bias baru ke-k  
 $b_k$  lama = bias lama ke-k  
 $W_{ki}$  baru = bobot baru pada baris ke-k dan kolom ke-i  
 $W_{ki}$  lama = bobot lama pada baris ke-k dan kolom ke-i

#### b. *MADALINE Rule II*

*MADALINE Rule II* adalah pengembangan lebih lanjut dari *MADALINE Rule I* dimana tersusun atas beberapa *ADALINE* penyusunnya. *MADALINE Rule II* merupakan algoritma pelatihan yang lebih rumit dari *MADALINE* sebelumnya. Untuk memenuhi konvergensi, pada *MADALINE Rule II* minimal memiliki 3 jumlah neuron pada lapisan tersembunyi (*Hidden Layer*)[13].

Terdapat perbedaan antara *MADALINE Rule II* dengan *MADALINE* sebelumnya yaitu pada perubahan bobot. Pada *MADALINE Rule I*, perubahan bobot hanya terjadi banyak lapisan pertama sedangkan bobot pada lapisan kedua merupakan kuantitas yang tetap. Pada *MADALINE Rule II*, perubahan bobot terjadi pada semua unit *ADALINE* sesuai dengan "*minimal disturbance principle*". Adapun prosedur atau algoritma *MADALINE Rule II* sebagai berikut[14]:

1. Set sebuah *training vector* pada *input MADALINE* dan menyebarkannya sampai pada *output* unit.
2. Hitung *error* pada lapisan *output*.
3. Untuk semua unit pada lapisan *output*,
  - a. Pilih node yang sebelumnya belum dipilih dengan *analog output* paling mendekati nilai nol (*Node* ini adalah *node* yang bisa membalikkan *output bipolar*-nya dengan sedikit perubahan bobot sesuai dengan *minimum disturbance*).

- b. Ubah bobot pada unit yang dipilih sehingga mengubah *output bipolar* unit.
- c. Propagasikan *input* vektor ke depan dari *input* ke *output* sekali lagi.
- d. Jika hasil perubahan bobot menghasilkan pengurangan jumlah kesalahan (*error*), lakukan perubahan bobot; Jika tidak, kembalikan bobot semula.

4. Ulangi langkah 3 untuk semua lapisan kecuali lapisan *input*.

5. Untuk semua unit pada lapisan *output*,
 

- a. Pilih unit berpasangan yang sebelumnya belum dipilih yang *analog output* paling mendekati nol.

b. Set perubahan bobot pada kedua unit sehingga mengubah *bipolar output* pada kedua unit.

c. Propagasikan *input* vektor ke depan dari *input* ke *output* sekali lagi.

d. Jika hasil perubahan bobot menghasilkan pengurangan jumlah kesalahan (*error*), lakukan perubahan bobot; Jika tidak, kembalikan bobot semula.

6. Ulangi langkah 5 untuk semua lapisan kecuali lapisan *input*.

Jika perlu, urutan pada langkah 5 dan 6 dapat diulang dengan tiga pasang unit, atau empat pasang unit, atau kombinasi yang lebih besar lagi, sampai hasil memuaskan diperoleh.

#### c. *MADALINE Rule III*

*MADALINE Rule III* mirip dengan *MADALINE Rule II*, namun unit individu memiliki fungsi keluaran kontinu. Oleh karena itu, *MADALINE Rule III* tidak menggunakan fungsi *threshold bipolar*. *MADALINE Rule III* menggunakan fungsi *sigmoid* pada setiap *node*-nya.

*MADALINE Neural Network* menggunakan masukan dengan *range* di antara -1 dan 1. Oleh sebab itu, diperlukan normalisasi data sebelum dilakukan pelatihan maupun pengujian. Untuk mentransformasikan data dengan *range* -1 dan 1, maka digunakan metode Min-Max. Persamaan metode Min-Max ditunjukkan pada persamaan 11[15].

$$\text{data baru} = \frac{\text{data lama} - \text{min}}{\text{max} - \text{min}} (\text{max baru} - \text{min baru}) + \text{min baru} \quad (11)$$

dimana:

min = data lama terendah  
max = data lama tertinggi  
min baru = nilai terendah baru  
max baru = nilai tertinggi baru

Dalam algoritma *MADALINE Neural Network* terdapat beberapa cara membatasi perulangan untuk pencarian bobot terbaik. Cara pertama dengan memberi batas iterasi. Pada cara ini, hasil bobot yang diambil adalah hasil bobot terakhir saat iterasi mencapai maksimum. Cara kedua dengan memberi nilai *goal*. Nilai *goal* merupakan nilai batas toleransi yang diijinkan atau perubahan bobot terbesar yang diijinkan. Variasi nilai *goal* yang dapat digunakan adalah antara 1 hingga 0. Semakin banyaknya iterasi, semakin kecil juga nilai toleransi yang diijinkan diperoleh. Semakin kecil nilai toleransi yang diijinkan sama dengan semakin kecil perubahan bobot terbesar yang terjadi. Semakin kecil perubahan bobot terbesar yang terjadi semakin kecil perbedaan keluaran dengan target yang dicapai [16].

Dalam Jaringan Syaraf Tiruan terdapat pengujian terhadap keakuratan sistem. Perhitungan akurasi merupakan perbandingan antara jumlah data yang benar terhadap keseluruhan data. Perhitungan akurasi menggunakan persamaan 12 [1].

$$\text{akurasi} = \frac{\text{jumlah data terdeteksi dengan benar}}{\text{jumlah total data}} \times 100\% \quad (12)$$

### 3. METODE PENELITIAN

Dalam penelitian ini penerapan *MADALINE Neural Network* untuk prakiraan cuaca, terdapat enam metodologi penelitian, yaitu studi literatur, metode pengumpulan data, analisis kebutuhan, perancangan sistem, implementasi, dan pengujian.

Studi literatur bertujuan untuk mempelajari teori-teori yang berhubungan dengan prakiraan cuaca dengan penerapan *MADALINE Neural Network* pada prakiraan cuaca. Studi literatur dilakukan dengan cara membaca buku-buku baik nasional maupun buku-buku berskala internasional yang diterjemahkan dan jurnal penelitian yang terkait serta artikel dari internet. Hasil pelaksanaan studi literatur mengenai metode *MADALINE Neural Network* dan parameter-parameter yang mempengaruhi cuaca serta kelas-kelas cuaca di Kota Pontianak.

Metode pengumpulan data untuk kebutuhan penelitian ini meliputi data-data yang dapat menunjang penelitian yang berkaitan dengan informasi data histori cuaca

untuk prakiraan cuaca. Data histori cuaca merupakan data sekunder yang diperoleh dari *website* Weather Underground. Data pada penelitian ini berupa data parameter-parameter cuaca (suhu udara, tekanan udara, kelembaban udara, kecepatan angin dan jarak pandang), kondisi cuaca berupa jenis cuaca yang terjadi di Kota Pontianak. Terdapat data tambahan pada aplikasi berupa kondisi cuaca beserta parameter-parameter cuaca yang sedang berlangsung dan yang telah berlangsung selama satu hari. Cara pengambilan data dari *website* Weather Underground dengan pengambilan data berupa format JSON.

Analisis kebutuhan adalah suatu proses untuk mendapatkan informasi, mode, spesifikasi tentang perangkat lunak yang diinginkan klien/pengguna. Analisis kebutuhan digunakan dalam pembuatan sebuah aplikasi prakiraan cuaca berbasis *website* dengan penerapan metode *MADALINE neural network*. Kebutuhan dalam penelitian ini meliputi kebutuhan perangkat lunak. Kebutuhan perangkat lunak yang dibutuhkan dalam penelitian ini berupa Sublime Text Editor, XAMPP, Mysql, Balsamiq Mock-up dan *web browser*.

Perangkat keras komputer yang digunakan adalah perangkat keras yang dapat mendukung perangkat lunak yang memiliki kemampuan atau tampilan grafis yang cukup baik. Kebutuhan perangkat keras yang digunakan pada aplikasi *MADALINE Neural Network* untuk prakiraan cuaca. Kebutuhan perangkat keras yang digunakan pada aplikasi *MADALINE Neural Network* untuk prakiraan cuaca adalah Intel® Core™ i7-4720HQ CPU @ 2.60 GHz, penggunaan RAM 8 GB, memori 1000 GB, dan VGA 950M GTX.

Pada perancangan sistem terdapat desain model sistem, desain *database*, desain *flowchart* *Madaline Neural Network*, desain model jaringan, dan desain antarmuka aplikasi.

Setelah sistem telah dirancang, maka dilakukan implementasi sistem berupa aplikasi prakiraan cuaca dengan penerapan metode *MADALINE Neural Network*. Hasil perancangan diimplementasikan berupa model sistem, dan antarmuka aplikasi.

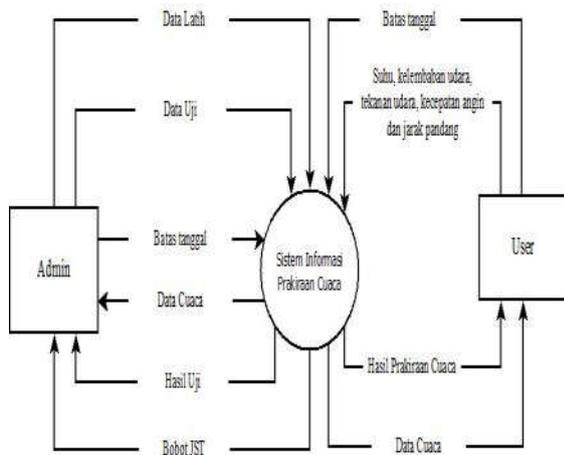
Setelah dilakukan implementasi sistem, selanjutnya adalah pengujian sistem untuk mengetahui akurasi prakiraan cuaca yang dihasilkan oleh sistem. Pengujian sistem dalam penelitian ini menggunakan skenario pengujian prakiraan cuaca dengan variasi nilai toleransi

yang diijinkan dan pengujian prakiraan cuaca dengan variasi laju pembelajaran.

#### 4. PERANCANGAN SISTEM

##### 4.1 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) merupakan aliran dari aplikasi prakiraan cuaca yang dibuat. DFD menggambarkan *input*, *process*, dan *output* yang terjadi dalam suatu sistem. Dibawah ini merupakan desain Data Flow Diagram (DFD) dari aplikasi prakiraan cuaca adalah sebagai berikut:



Gambar 2. DFD Level 0

Pada DFD level 0 seperti pada gambar 2 dapat dilihat bahwa dalam aplikasi prakiraan cuaca terdapat 2 pengguna yaitu admin dan user. Pada aplikasi ini, admin dapat melakukan pengolahan data cuaca, melakukan pelatihan jaringan untuk mendapatkan bobot terbaik, dan melakukan pengujian jaringan untuk mendapatkan keakuratan dalam memprakirakan cuaca. Pengguna atau user dapat memasukkan data-data parameter cuaca untuk memprakiraan cuaca pada hari yang ditentukan. Pengguna atau user juga dapat melihat data cuaca yang tersedia dalam database.

Pada DFD level 1 terdapat 2 pengguna yaitu admin dan user. Seorang admin memiliki hak akses dalam melakukan pengolahan data cuaca dan data admin pada database sistem, sedangkan pengguna hanya dapat melihat data cuaca yang tersedia dalam database. Admin dapat melakukan pelatihan data dan pengujian data untuk mendapatkan hasil bobot dan bias yang terbaik serta hasil keakuratan data. Seorang pengguna dapat melakukan prakiraan

cuaca dengan menggunakan bobot dan bias terbaik serta melihat hasil prakiraan cuaca.

Di dalam proses mengolah data cuaca pada sistem terdapat beberapa proses yang dapat dijabarkan menjadi proses memasukkan data cuaca, proses menghapus data cuaca dan proses melihat data cuaca. Pada ketiga proses ini diperlukan masukan batas tanggal untuk memasukkan data cuaca, menghapus data cuaca, dan melihat data cuaca dengan waktu yang ditentukan.

Di dalam proses pelatihan data pada sistem terdapat proses melatih data, menghapus hasil pelatihan, melihat hasil pelatihan, dan memasukkan hasil pelatihan. Pada proses tersebut dibutuhkan masukan parameter-parameter pelatihan seperti batas tanggal, batas iterasi, laju pembelajaran dan nilai toleransi yang diijinkan untuk melakukan pelatihan data. Proses menghapus hasil pelatihan adalah proses menghapus hasil pelatihan yang telah disimpan sebelumnya. Proses memasukkan hasil pelatihan adalah proses menyimpan bobot, bias, batas latih berupa tanggal, waktu, toleransi, dan jumlah iterasi serta pada proses ini juga menyimpan hasil pelatihan terbaik yang dapat digunakan dalam prakiraan cuaca.

Di dalam proses mengolah data admin pada sistem terdapat beberapa proses yang dapat dijabarkan menjadi proses mengubah data admin, dan proses melihat data admin.

##### 4.2 Perancangan Model Jaringan

Sebelum melakukan implementasi dan pengujian proses prakiraan, terlebih dahulu dilakukan perancangan model jaringan.

###### 1. Input dan Output Jaringan

Tabel 1. Kategori Cuaca

Jenis Cuaca	Y1	Y2	Y3
Cerah	-1	-1	-1
Berkabut	-1	-1	1
Berawan	-1	1	-1
Hujan	1	-1	-1
Badai Petir	1	1	1

Pada prakiraan dengan *time series*, perlu adanya data historis dari variabel yang akan diprakirakan yaitu data parameter cuaca pada hari-hari sebelumnya. Pada penelitian ini model jaringan menggunakan tiga buah *input* yaitu  $H(n)$ ,  $H(n-1)$ ,  $H(n-2)$  dan *output*  $H(n+1)$  untuk prakiraan cuaca. Variabel  $H$  adalah hari. *Input* yang digunakan berjumlah 15 parameter (3 hari x 5 parameter cuaca). *Output* dari jaringan

berupa 5 jenis cuaca yang dimodelkan ke dalam 3 *output bipolar* seperti pada tabel 1.

## 2. Jumlah Lapisan

Jaringan *MADALINE Rule II* umumnya dibangun dengan tiga lapisan, yaitu lapisan *input*, lapisan tersembunyi, dan lapisan *output*. Pada penelitian ini digunakan sebuah lapisan tersembunyi dengan tiga neuron.

## 3. Jumlah Neuron

Jumlah neuron pada lapisan *input* ditentukan berdasarkan jumlah *input* yang digunakan pada jaringan. Pada penelitian ini menggunakan jumlah neuron pada lapisan *input* dan *output* masing-masing adalah 15 dan 3.

## 4. Fungsi Aktivasi

Algoritma yang digunakan adalah algoritma *MADALINE Rule II*, maka dipilih fungsi *threshold bipolar*.

## 5. Data

Data cuaca yang digunakan untuk pelatihan dan pengujian dengan perbandingan data latih dan data uji adalah 70%:30%. Sebelum digunakan dalam algoritma, dilakukan proses normalisasi terlebih dahulu agar hasil perhitungan fungsi aktivasi tidak jatuh dalam daerah saturasi.

## 6. Nilai Toleransi, Laju Pembelajaran, dan Jumlah Iterasi

Pada penelitian ini dilakukan uji toleransi sehingga dilakukan pemilihan variasi nilai toleransi dari yang terbesar sampai yang terkecil dengan laju pembelajaran awal 0,005. Toleransi dengan akurasi terbaik (toleransi terbaik) dengan iterasi tertentu digunakan untuk uji laju pembelajaran sehingga dilakukan

pemilihan variasi nilai laju pembelajaran dari 0,001 sampai 0,009.

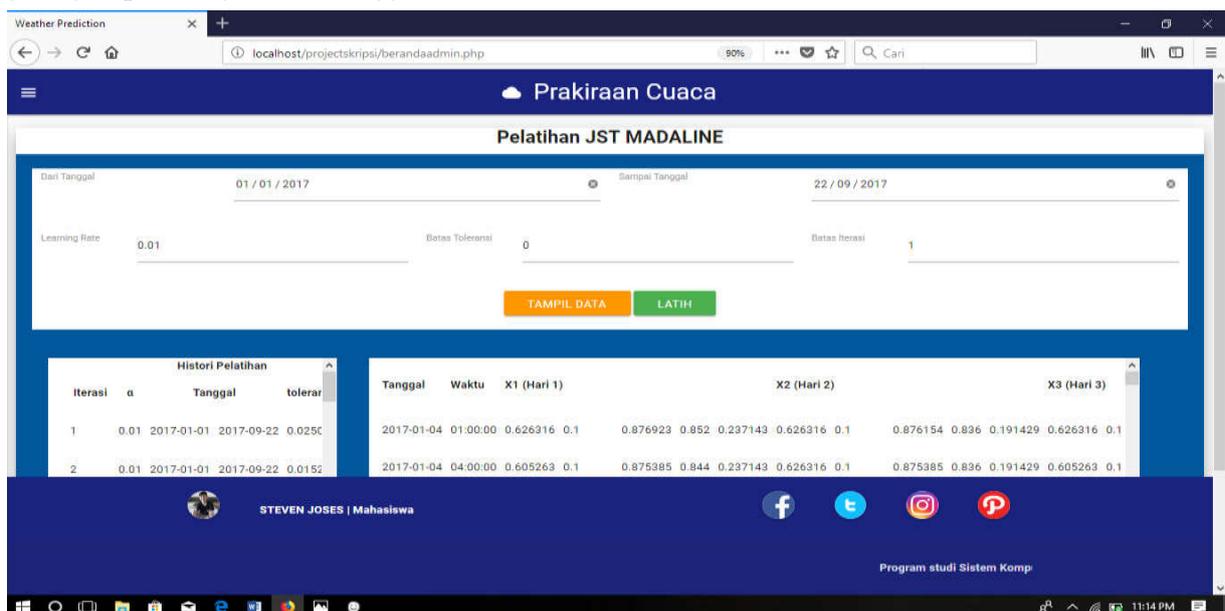
## 5. HASIL DAN PEMBAHASAN

### 5.1 Implementasi Sistem

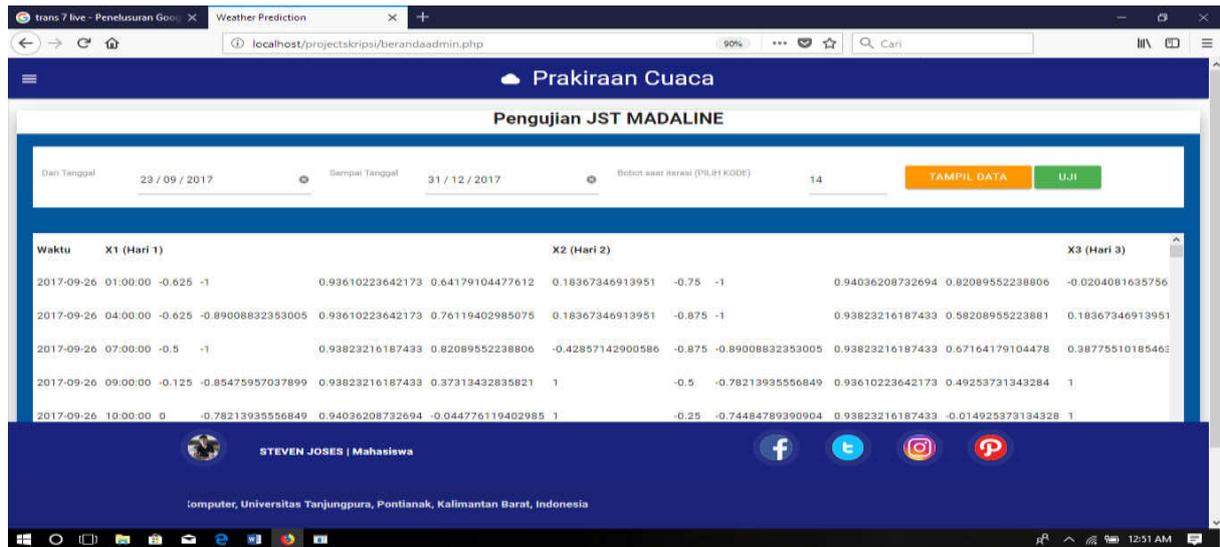
Halaman pelatihan dan pengujian data pada gambar 3 dan 4 dibuat sesuai dengan perancangan sistem. Pada navigasi bar beranda admin terdapat menu pelatihan yang akan diarahkan menuju halaman pelatihan. Halaman pelatihan bertujuan untuk melatih data cuaca dan menghasilkan bobot dan bias yang digunakan untuk pengujian dan prakiraan cuaca. Pada halaman pelatihan terdapat histori pelatihan yang merupakan hasil pelatihan yang sudah dilakukan dan disimpan ke dalam *database*. Pada navigasi bar beranda admin terdapat menu pengujian yang akan diarahkan menuju halaman pengujian. Halaman pengujian bertujuan untuk menguji data cuaca dan menghasil keakuratan data uji.

### 5.2 Data

Pada aplikasi prakiraan cuaca menggunakan data harian cuaca dalam pelatihan jaringan, pengujian jaringan serta proses prakiraan cuaca. Data cuaca berbentuk JSON yang diambil dari *website* Weather Underground. Data cuaca yang diperoleh dari *website* Weather Underground dengan menggunakan *API key* yang diberikan saat melakukan registrasi. *API key* tersebut digunakan pada *format url* yang disediakan Weather Underground dalam melakukan *request* data cuaca. Tampilan awal data cuaca



Gambar 3. Halaman Pelatihan Data



Gambar 4. Halaman Pengujian Data

berupa JSON sehingga dilakukan proses pengubahan data pada JSON ke dalam variabel *array* pada pemrograman *PHP*. Setiap variabel *array* yang berisi parameter-parameter cuaca dan kondisi cuaca selanjutnya disimpan pada *database*. Data yang diambil dari *website* ini berupa parameter-parameter cuaca seperti suhu udara, tekanan udara, jarak pandang, kelembaban udara dan kecepatan angin.

Data cuaca sebanyak 21583 data dengan pembagian 15115 data sebagai data latih dan 6468 data sebagai data uji. Data untuk pelatihan dimulai dari tanggal 1 Januari 2014 sampai dengan 2 Mei 2017, sedangkan data untuk pengujian dimulai dari tanggal 3 Mei 2017 sampai dengan tanggal 30 April 2018.

Dalam metode *MADALINE Rule II* diperlukan normalisasi data ke dalam rentang -1 dan 1. Oleh karena itu, diperlukan proses normalisasi data cuaca. Proses normalisasi digunakan metode Min-Max. Metode Min-Max dapat mengubah data ke dalam rentang data yang diinginkan. Setelah data sudah dinormalisasi, maka dibuat model prakiraan cuaca yaitu hari ke-1, hari ke-2 dan hari ke-3 sebagai data *input*. hari ke-4 sebagai data *output*.

### 5.3 Pengujian

Pengujian sistem dilakukan dengan menguji nilai toleransi, nilai laju pembelajaran dan keakuratan cuaca. Pengujian dengan variasi nilai toleransi digunakan untuk mendapatkan nilai toleransi terbaik berdasarkan tingkat ketepatan prakiraan cuaca. Pelatihan data latih menggunakan laju pembelajaran awal sebesar 0,005 dengan maksimum iterasi adalah 100000

untuk mencari variasi nilai toleransi. Hasil pelatihan dengan variasi nilai toleransi ditunjukkan pada tabel 2.

Tabel 2. Pelatihan data dengan variasi nilai toleransi

Iterasi	Nilai toleransi yang diperoleh
1	0.00511903
416	0.00502004
2506	0.00501859
2950	0.00500821
65080	0.00500639

Nilai toleransi terbaik telah diperoleh berdasarkan tabel 2 adalah 0.00500639. Selanjutnya dilakukan perhitungan akurasi data latih menggunakan setiap nilai toleransi untuk memperoleh nilai toleransi yang terbaik dengan akurasi yang tinggi. Hasil akurasi data latih dengan variasi nilai toleransi ditunjukkan pada tabel 3.

Tabel 3. Hasil akurasi data latih dengan variasi nilai toleransi

Iterasi	Nilai toleransi	Benar	Salah	Akurasi
1	0.00511903	7563	7552	50.03%
416	0.00502004	10766	4349	71.22%
2506	0.00501859	10813	4302	71.53%
2950	0.00500821	10813	4302	71.53%
<b>65080</b>	<b>0.00500639</b>	<b>10816</b>	<b>4299</b>	<b>71.55%</b>

Berdasarkan tabel 3 dapat dilihat bahwa pada uji akurasi data latih diperoleh nilai toleransi dengan akurasi tertinggi adalah 0,00500639 dengan jumlah iterasi yaitu 65080. Data latih yang digunakan sebanyak 15115 data

cuaca dan terdapat 4299 data cuaca yang masih belum diprakirakan dengan baik.

Pengujian selanjutnya adalah pengujian dengan variasi nilai laju pembelajaran digunakan untuk mendapatkan nilai laju pembelajaran terbaik berdasarkan tingkat ketepatan prakiraan cuaca. Pelatihan data latih menggunakan nilai toleransi terbaik dengan iterasi terbaik yang telah diperoleh dari pengujian sebelumnya yaitu nilai toleransi yang diperoleh sebesar 0,00500639 dan jumlah iterasi yaitu 65080. Hasil akurasi data latih dengan variasi nilai laju pembelajaran ditunjukkan pada tabel 4.

Tabel 4. Hasil akurasi data latih dengan variasi laju pembelajaran

Nilai laju pembelajaran	Benar	Salah	Akurasi
0.001	9306	5809	61.56%
0.002	7582	7533	50.16%
0.003	7299	7816	48.28%
0.004	4696	10419	31.06%
<b>0.005</b>	<b>10816</b>	<b>4299</b>	<b>71.55%</b>
0.006	10346	4769	68.44%
0.007	9732	5383	64.38%
0.008	9225	5890	61.03%
0.009	9207	5908	60.91%

Pada pelatihan data dengan variasi nilai laju pembelajaran menggunakan dua *stop* kondisi. Jika perubahan bobot masih lebih besar dari batas toleransi yang diijinkan tetapi maksimum iterasi telah dicapai maka *stop* kondisi menggunakan maksimum iterasi. Jika perubahan bobot lebih kecil dari batas toleransi yang diijinkan dan maksimum iterasi belum dicapai maka *stop* kondisi menggunakan batas toleransi yang diijinkan.

Berdasarkan tabel 4 dapat dilihat bahwa pada uji akurasi data latih diperoleh nilai laju pembelajaran terbaik adalah 0,005 dengan data latih sebanyak 15115 data. Terdapat 10816 data yang tepat dan 4299 data yang masih salah dengan tingkat akurasi 71,55%.

Dengan menggunakan parameter data latih, dilakukan pengujian terhadap 6468 data uji. Perhitungan akurasi data uji menggunakan persamaan 12.

$$\text{Akurasi} = \frac{5083}{6468} \times 100\% = 78.58\%$$

Hasil akurasi dengan 6468 data uji adalah 78,58% dengan 5083 data uji yang bernilai benar.

#### 5.4 Pembahasan

Jaringan Syaraf Tiruan umumnya digunakan untuk pengelompokkan dan pemisahan data yang proses kerjanya menyerupai jaringan syaraf biologi. Jaringan Syaraf Tiruan mampu mengenali data yang sudah diketahui sebelumnya. Sesuai dengan sistem kerjanya, Jaringan Syaraf Tiruan terdiri atas beberapa lapisan yaitu lapisan *input*, lapisan tersembunyi dan lapisan *output*. Setiap lapisan terdapat beberapa neuron yang terhubung dengan neuron terdekat pada lapisan di atasnya. Setiap neuron terhubung dengan neuron lain melalui nilai bobot. Setiap neuron pada semua lapisan (kecuali lapisan *input*) memiliki nilai bias. Pada Jaringan Syaraf Tiruan terdapat proses pelatihan dan pengujian data.

*MADALINE Rule II* adalah salah satu algoritma dari *MADALINE Neural Network*. *MADALINE Neural Network* merupakan salah satu metode yang ada pada Jaringan Syaraf Tiruan. Penerapan *MADALINE Rule II* pada penelitian ini dilakukan dengan menggunakan neuron *input* sebanyak 15 unit, neuron pada lapisan tersembunyi sebanyak 3 unit, dan neuron pada lapisan *output* sebanyak 3 unit.

Pada proses pelatihan membutuhkan waktu eksekusi yang lama. Hal ini disebabkan karena pada algoritma *MADALINE Rule II* melakukan proses "*trial adaptation*" yaitu proses adaptasi setiap neuron dengan melakukan perubahan keluaran pada neuron tersebut untuk mengecek *error* jaringan sehingga dapat melakukan perubahan bobot pada neuron tersebut. Proses ini dilakukan satu per satu pada neuron dari lapisan tersembunyi sampai pada neuron di lapisan keluaran.

Proses pelatihan menggunakan 15115 data cuaca harian. Dilakukan proses *trial and error* untuk nilai perubahan bobot terbesar (nilai toleransi yang diijinkan) dengan laju pembelajaran awal 0,005. Hasil pelatihan yang terbaik maka akan digunakan untuk pengujian data. Dari hasil pelatihan didapatkan nilai toleransi yang semakin kecil dan iterasi yang semakin banyak dengan beragam tingkat akurasi yang diperoleh. Pada nilai toleransi 0,00500639 telah diperoleh tingkat akurasi yang tertinggi dibandingkan nilai toleransi lainnya yaitu 71,55%. Dengan semakin kecilnya nilai toleransi yang diperoleh maka semakin tinggi tingkat akurasi yang didapatkan.

Nilai toleransi terbaik selanjutnya digunakan untuk uji laju pembelajaran. Dari

hasil pelatihan diperoleh laju pembelajaran terbaik yaitu 0,005 dengan tingkat akurasi data latih sebesar 71,55%. Oleh karena laju pembelajaran 0,005 mencapai akurasi tertinggi dibandingkan nilai laju pembelajaran lainnya maka digunakan untuk proses pengujian data. Hasil akurasi data uji yang diperoleh sebesar 78,58%

## 6. KESIMPULAN DAN SARAN

### 6.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Pada data pelatihan, nilai toleransi yang diperoleh dengan *MADALINE Rule II* yaitu 0,00500639. Nilai toleransi tersebut diperoleh dari pelatihan dengan pemberian laju pembelajaran awal  $\alpha = 0,005$  dengan 15 neuron pada lapisan *input*, 3 neuron pada lapisan tersembunyi, dan 3 neuron pada lapisan *output* dan berhenti pada iterasi ke-65080. Semakin kecilnya nilai toleransi, maka semakin tinggi akurasi yang diperoleh. Nilai toleransi yang besar juga dapat menyebabkan terjadinya *error* yang besar. Hal ini disebabkan karena perbedaan keluaran jaringan dan target yang besar dapat menyebabkan sistem tidak dapat mengenali target dengan tepat.
2. Proses pelatihan data dengan variasi laju pembelajaran digunakan untuk memperoleh laju pembelajaran terbaik. Laju pembelajaran yang terbaik dari proses pelatihan *MADALINE Rule II* yaitu 0,005. Laju pembelajaran tersebut diperoleh dari pelatihan dengan menggunakan nilai toleransi terbaik serta maksimum iterasi terbaik yang diperoleh saat pengujian nilai toleransi. Semakin besar laju pembelajaran, maka jaringan tidak dapat dilatih dikarenakan terjadinya perubahan bobot besar yang menyebabkan besarnya selisih antara target dan keluaran. Namun apabila semakin kecilnya laju pembelajaran, maka diperlukan waktu yang lama dalam pelatihannya dikarenakan terjadinya perubahan bobot kecil yang menyebabkan perubahan kecil pada selisih antara target dan keluaran.
3. Hasil keluaran jaringan menggunakan 15115 data latih dengan nilai toleransi terbaik dan laju pembelajaran terbaik menghasilkan akurasi sebesar 71,55% dengan 10816 data cuaca yang dapat

diprakirakan dengan baik. Hasil keluaran jaringan dengan menggunakan 6468 data uji menghasilkan akurasi sebesar 78,58% dengan 5083 data cuaca yang dapat diprakirakan dengan baik.

### 6.2 Saran

Berdasarkan hasil yang diperoleh dalam penelitian ini, maka penulis memberikan beberapa saran yang perlu menjadi bahan pertimbangan untuk penelitian selanjutnya yaitu:

1. Untuk memperoleh hasil lebih baik, pada metode *MADALINE Rule II* dapat dilakukan perbaikan arsitektur seperti menambah jumlah neuron pada lapisan tersembunyi.
2. Untuk memperoleh hasil lebih baik, sebaiknya dilakukan penambahan faktor-faktor lain yang mempengaruhi kondisi cuaca.

## DAFTAR PUSTAKA

- [1] C. Dewi, "Prediksi Cuaca Pada Data Time Series Menggunakan Adaptive Neuro Fuzzy Inference System (ANFIS)," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, pp. 18-24, 2014.
- [2] Retnawati, "Estimasi Keadaan Cuaca di Kota Pontianak Menggunakan Aplikasi Jaringan Syaraf Tiruan (JST) Algoritma Hopfield," *POSITRON*, pp. 43-46, 2013.
- [3] E. S. Puspita, "Perancangan Sistem Peramalan Cuaca Berbasis Logika Fuzzy," *Jurnal Media Infotama*, pp. 1-10, 2016.
- [4] R. Winter and B. Widrow, "MADALINE RULE II: A Training Algorithm for Neural Networks," in *IEEE International Conference on Neural Networks*, San Diego, 1988.
- [5] A. Fadholi, "Study Pengaruh Suhu Dan Tekanan Udara Terhadap Operasi Penerbangan Di Bandara H.A.S. Hananjoeddin Buluh Tumbang Belitung Periode 1980-2010," *Jurnal Penelitian Fisika dan Aplikasinya (JPFA)*, pp. 1-10, 2013.
- [6] J. Siang, *Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan Matlab*, Yogyakarta: Penerbit ANDI, 2005.

- [7] R. Saranya, "Performance analysis of ADALINE and MADALINE Network in classification of Thyroid Disease," *International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)* , pp. 121-123, 2016.
- [8] Z. Azmi dan H. Freizello, "Implementasi Hopfield untuk Pengenalan Karakter Hiragana," *JISICOM*, pp. 1-8, 2017.
- [9] A. Blum, *Neural Networks in C++*, New York: John Wiley & Sons Inc, 1992.
- [10] B. Widrow and M. A. Lehr, "Artificial Neural Network of Perceptrons Madaline and Backpropagation Family," *Neurobionics*, pp. 133-205, 1993.
- [11] K. Mehrotra, C. Mohan and S. Ranka, *Elements of Artificial Neural Networks*, London: The MIT Press, 1996.
- [12] I. P. Sutawinaya, "Perbandingan Metode Jaringan Saraf Tiruan Pada Peramalan Curah Hujan," *JURNAL LOGIC*, pp. 92-97, 2017.
- [13] D. Graupe, *Principles of Artificial Neural Networks*, 5 Toh Tuck Link: World Scientific Publishing Company, 2013.
- [14] J. A. Freeman and D. M. Skapura, *Neural Networks Algorithms, Applications, and Programming Techniques*, Redwood City: Addison-Wesley Publishing Company, Inc, 1991.
- [15] Fitri, "Aplikasi Jaringan Syaraf Tiruan Untuk Penentuan Status Gizi Balita Dan Rekomendasi Menu Makanan Yang Dibutuhkan," *Jurnal EECCIS*, pp. 119-124, 2013.
- [16] Y. A. Priambudi, "Sistem Klasifikasi Rasa Kopi Berbasis Electronic Tongue Menggunakan Madaline Neural Network," *IJEIS*, pp. 201-210, 2014.