

BAB II

METODE PENELITIAN

2.1 Objek Penelitian

Salah satu penghambat dalam pertanian padi adalah hama dan penyakit yang menyerang pada tanaman padi. Namun pengetahuan para petani padi tentang hama dan penyakit serta pengendaliannya masih kurang. Sementara jumlah pakar tentang hama dan penyakit padi terbatas (Sholikhah, Kurniadi and Riansyah, 2021). Maka dari itu salah satu ilmu dalam bidang teknologi yang dapat menjadi solusi pada masalah tersebut yaitu ilmu Jaringan Syaraf Tiruan.

Pada penelitian ini Jaringan Syaraf Tiruan digunakan untuk mengklasifikasi penyakit pada tanaman padi. Untuk menghasilkan nilai klasifikasi yang bagus dipilihlah metode *Backpropagation* dengan algoritma optimasi *Firefly*. Data tanaman padi di ambil dari kuisioner kepada petani padi berdasarkan gejala-gejala yang terdapat di kota Samarinda. Adapun jenis-jenis penyakit dan gejala pada penyakit tanaman padi dapat dilihat pada Table 2.1.

Table 2.1 Data Penyakit Padi

Penyakit	Gejala
1. Tungro	<ul style="list-style-type: none"> - Daun kuning kemerah – merahan - Daun muda menjadi belang/garis hijau pucat - Bulir bercak cokelat dan beratnya kurang dibanding normal - Kerdil - Jumlah anakan sedikit berkurang - Pertumbuhan akar tidak sempurna
2. Blas	<ul style="list-style-type: none"> - Bercak berbentuk belah ketupat - Bercak cokelat kehitaman pada batang - Batang mudah patah - Malai hampa atau tidak berisi
3. Kresek	<ul style="list-style-type: none"> - Daun berwarna putih kekuningan - Layu - Batang berwarna cokelat - Daun garis memanjang atau oval
4. Bercak daun cokelat	<ul style="list-style-type: none"> - Bercak muda berbentuk bulat kecil - Bercak berwarna cokelat gelap - Bercak tua berukuran lebih besar (0,4 – 1 cm x -,1 – 02 cm) - Bercak berwarna kuning di sekelilingnya - Bulir berwarna cokelat kehitaman
5. Bercak cokelat sempit	<ul style="list-style-type: none"> - Pada daun dan pelepah daun terdapat bercak cokelat yang sempit - Varietas yang tahan bercak berukuran 0,2 – 1 cm x 0,1 cm, berwarna cokelat gelap - Varietas bercak lebih besar & berwarna cokelat terang
6. Bercak Garis	<ul style="list-style-type: none"> - Muncul garis yang kebasah – basahan diantara urat daun

- Garis memanjang dan menjadi cokelat dengan lingkaran kuning di sekelilingnya
 - Berlendir
 - Lendir yang kering berbentuk butiran kecil pada garis luka
-

2.2 Data Penelitian

Dalam penelitian ini, data diperoleh dari kuisisioner yang di ambil oleh beberapa petani di Samarinda. Data yang di ambil berjumlah 1000 data, terdiri dari jawaban dan tanggapan terhadap petani. Data ini berisi 11 penyakit dan gejala yang terdapat pada tanaman padi seperti pada contoh Tabel 2.1.

Untuk mendapatkan data yang akurat, dalam penelitian ini menggunakan Teknik wawancara kepara petani yang ada di Samarinda, Kec. Samarinda Utara, Lempake, Kec. Tenggarong Seberang, Kabupaten Kutai Kartanegara. Untuk mengamati kondisi tanaman padi dilakukan Observasi. Observasi dilakukan agar bisa mendapatkan informasi terkait penyakit yang ada pada tanaman padi melalui gejala-gejala yang ada. Dengan ada nya proses wawancara dengan petani, diharapkan dapat membentuk dasar yang kuat untuk klasifikasi penyakit pada tanaman padi menggunakan metode *Backpropagation* dengan algoritma optimasi *Firefly*.

2.3 Jaringan Saraf Tiruan

Jaringan Saraf Tiruan (*Artificial Neural Network*), adalah sistem pengolahan informasi yang dirancang untuk meniru cara kerja otak manusia dalam menyelesaikan masalah dengan memperbarui bobot koneksi sinopsisnya selama proses pembelajaran (Damanik *et al.*, 2022).

Jaringan Syaraf Tiruan sering kali menggunakan arsitektur jaringan yang digunakan secara luas dalam berbagai aplikasi. Berikut beberapa contoh dari arsitektur tersebut (Indrawan and Pandu Kusuma, 2021).

2.3.1 Jaringan Lapisan Tunggal

Jaringan ini terdiri dari satu lapisan masukan dan satu lapisan keluaran, dimana *neuron* pada lapisan masukan terhubung langsung ke *neuron* pada lapisan keluaran. Prosesnya langsung dari masukan ke keluaran tanpa melalui *hidden layer*.

2.3.2 Jaringan Banyak Lapisan

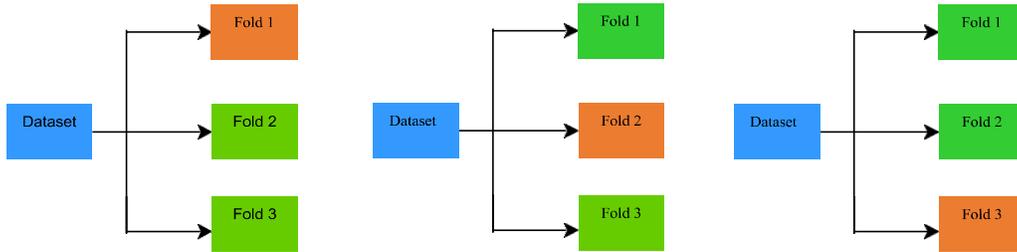
Jaringan dengan banyak lapisan memiliki karakteristik khusus yang terdiri dari tiga jenis lapisan, yaitu lapisan masukan, lapisan keluaran, dan *hidden layer*. Jaringan ini mampu menyelesaikan masalah yang lebih spesifik dibandingkan dengan jaringan yang hanya memiliki satu lapisan. Namun, proses pelatihan pada jenis jaringan ini cenderung lambat, dan menghabiskan waktu yang lebih lama.

2.3.3 Jaringan Lapisan Kompetitif

Pada fase ini, *neuron* dalam jaringan akan terhubung dan bersaing menjadi aktif dengan merespon pola input yang mirip.

2.4 K-Fold Cross Validation

K-Fold Cross Validation adalah metode di mana kumpulan data dibagi menjadi K bagian atau *fold*, dan setiap *fold* secara bergantian digunakan sebagai set pengujian pada berbagai titik dalam proses evaluasi model (Peryanto, Yudhana and Umar, 2020). Setiap kelompok secara bergantian digunakan sebagai data uji, sementara kelompok lainnya berfungsi sebagai data latih. Evaluasi kinerja algoritme dilakukan dengan menggunakan setiap kelompok sebagai data uji secara bergiliran, sementara data yang tersisa digunakan sebagai data latih (Adiba, Nur Risal and Tahir, 2023). Mari menggunakan validasi *2-fold* ($K = 2$). Kumpulan data dibagi menjadi 2 bagian *fold*. Pada iterasi pertama, bagian pertama digunakan untuk menguji model, sementara bagian lainnya digunakan untuk melatih model. Pada iterasi kedua, bagian kedua digunakan untuk data uji, sementara bagian lainnya digunakan untuk data latih. Contoh alur *k-fold* dapat dilihat pada Gambar 2.1.



Gambar 2.1 Alur K-Fold

2.5 Backpropagation

Backpropagation adalah algoritma pembelajaran dalam Jaringan Saraf Tiruan. *Backpropagation* adalah proses pembelajaran dimana bobot-bobot dalam Jaringan Saraf Tiruan disesuaikan mundur berdasarkan nilai kesalahan yang terjadi selama proses pembelajaran (Ikhsan, Armansyah and Tamba, 2022). Langkah awal yang harus dilakukan adalah mengaktifkan *neuron* dengan menggunakan fungsi aktivasi serta dapat diubah-ubah seperti sigmoid berikut (Putri, 2021).

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

$$f'(x) = \sigma f(x)[1 - f(x)] \quad (2)$$

atau menggunakan tangen sigmoid

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

$$f'(x) = [1 + f(x)][1 - f(x)] \quad (4)$$

Langkah berikutnya adalah melakukan perhitungan feedforward dengan rumus sebagai berikut.

$$z_{in_j} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad (5)$$

Setiap data nilai masukan ($Z_{jj} = 1, \dots, p$) yang dijumlahkan termasuk biasanya. Kemudian, hasil dari data tersembunyi dihitung menggunakan fungsi aktivasi $z_j = f(z_{in_j})$ langkah berikutnya adalah mengirimkan keluaran dengan rumus sebagai berikut.

$$y_{in_k} = w_{0k} + \sum_{j=1}^n z_j w_{jk} \quad (6)$$

Dimana setiap data nilai luaran ($y_{k^f} k = 1, \dots, m$) yang di jumlahkan dengan bobot pada masukan sehabis itu dari data luaran menggunakan aktivasi $y_k = f(y_{in_k})$ yang selanjutnya di kirim ke seluruh luaran. (Zhao *et al.*, 2021)

Untuk melakukan penyesuaian bobot dan bias pada setiap keluaran, menggunakan cara sebagai berikut.

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad (7)$$

Untuk melakukan penyesuaian bobot dan bias pada setiap input, menggunakan cara sebagai berikut.

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (8)$$

Untuk melakukan propagasi error dengan mempertimbangkan nilai vektor selisih yang dievaluasi melalui perulangan mundur $r = r - 1, r - 2, \dots, 1$ adalah

$$\delta_k^i(i) = \left[\sum_{l=1}^{c+1} \delta_{k}^{r+1}(l) W^{r+1}(li) \right] \frac{\partial f^r}{\partial n_k^r(i)} \quad (9)$$

Dalam algoritma proses *Backpropagation of error* terdapat beberapa langkah, yang pertama setiap nilai keluaran yang dihasilkan ($y_{k^f} k = 1, 2, \dots, m$) di bandingkan dengan nilai target (t_k). Kemudian menentukan nilai bobot yang menghubungkan *hidden note*, *output note*, dan bias dengan menghitung nilai koreksi. Langkah selanjutnya adalah menghitung nilai bobot pada jalur yang dilaluinya, kemudian dikalikan dengan nilai *error* dari semua node pada lapisan tersembunyi saat telah mencapai lapisan luar. Nilai dari *hidden note* akan dijumlahkan setelah menerima setiap nilai.

Untuk setiap output, *error* akan dihitung untuk setiap nilai dan dibandingkan dengan target yang diinginkan. Karena untuk melakukan penyesuaian nilai bobot yang menghubungkan antara lapisan tersembunyi dan lapisan luar, nilai *error* tersebut digunakan sebagai dasar. Dalam *Backpropagation*, bentuk fungsi error (MSE) sebagai berikut

$$E = \frac{1}{2} \sum_{k=1}^c (t_k - y_k^{(T)})^2 \quad (10)$$

2.6 Multi-Class

Multi-Class merupakan Kumpulan data atau yang biasa di sebut *dataset* yang terdiri dari atribut, dan kelas. Dataset dengan karakteristik *Multi-Class* merupakan dataset yang memiliki kelas klasifikasi lebih dari 2 dan biasa di sebut *polynomial* (Handayani *et al.*, 2023). Dalam situasi klasifikasi multi kelas, terdapat lebih dari satu *hyperplane* yang dibentuk. Salah satu metode pendekatan yang digunakan adalah *One Versus All*.

Metode pendekatan *One Versus All* adalah metode yang membandingkan satu kelas dengan semua kelas lainnya, sesuai dengan namanya (Sastypratiwi, Muhardi and Noveanto, 2022). *One Versus One* adalah metode klasifikasi biner dibangun oleh setiap pasangan kelas. Pendekatan sederhana untuk menggabungkan semua hasil dari metode klasifikasi biner pada *One Versus One* yaitu menggunakan majority voting dimana contoh uji akan diklasifikasikan ke dalam kelas yang memiliki *vote* tertinggi (Meidianingsih, Eka and Meganingtyas, no date).

2.7 Firefly

Algoritma kunang-kunang atau *Firefly Algorithm* adalah sebuah algoritma metaheuristik yang terinspirasi dari perilaku berkedip kunang-kunang. Fokus utama perilaku ini adalah menarik kunang-kunang lain. Algoritma ini dikembangkan oleh Dr Xin She Yang pada tahun 2007 (Simamora, Hajar and Fernandes, 2019). Berikut adalah rumus nya

Berdasarkan pada algoritma *Firefly*, fungsi objektif didapatkan dari rumus berikut.

$$\text{maks } f(x), x = (x_1, \dots, x_d)^T \quad (1)$$

Algoritma *Firefly* juga harus menentukan intensitas cahaya dari setiap kunang-kunang.

Untuk menentukan intensitas cahaya di dapatkan dengan rumus berikut.

$$I(x) = f(x) \quad (2)$$

Setelah memperoleh intensitas cahaya, Langkah selanjutnya adalah menentukan jarak antara kunang-kunang. Jarak antara kunang-kunang dihitung menggunakan rumus berikut.

$$r_{1,2} = \sqrt{(m_1^1 - m_2^1)^2 + (m_1^2 - m_2^2)^2} \quad (3)$$

Bukan hanya jarak aja, tetapi algoritma *Firefly* juga butuh nilai daya tarik yang di miliki setiap kunang-kunang. Nilai daya tarik di dapatkan dengan rumus.

$$\beta = \beta_0 e^{-\gamma r^2} \quad (4)$$

Pergerakan kunang-kunang menjadi penentu dalam algoritma *Firefly*. Kunang-kunang dengan intensitas cahaya yang lebih rendah akan bergerak menuju kunang-kunang dengan intensitas cahaya yang lebih tinggi. Perpindahan *Firefly* di dapatkan dengan rumus.

$$m_{i_baru}^k = m_{i_lama}^k + \beta \frac{e^{-\gamma r^2}}{0} (m_{i_lama}^k - m_{i_lama}^k) + a(\text{rand} - \frac{1}{2}) \quad (5)$$

2.8 Alat dan Bahan

Dalam penelitian ini membutuhkan berbagai alat dan bahan untuk mendukung penelitian. Alat dan bahan yang di gunakan sebagai berikut.

a. Alat

Alat penelitian adalah alat perangkat fisik atau bisa juga berbentuk digital yang kegunaannya untuk mengumpulkan data, dan melakukan percobaan yang sesuai. Tabel alat yang di gunakan untuk mengklasifikasikan jenis penyakit pada tanaman padi menggunakan metode *Backpropagation* dengan optimasi *Firefly* dapat dilihat pada Table 2.2.

Table 2.2 Alat Penelitian

Komponen-komponen	Spesifikasi
Laptop	MSI KATANA GF66 11UC
Processor	intel Core i5 11400H Gen 11

RAM	16 GB
SSD	512 GB + 1 TB
<i>Operation System</i>	<i>Windows 11</i>
<i>Text Editor</i>	<i>Google Colab</i>
<i>Software Perhitungan Manual</i>	<i>Microsoft Excel</i>

b. Bahan

Bahan penelitian adalah materi yang digunakan dalam percobaan, dan analisis untuk mendukung tujuan riset. Tabel bahan yang digunakan untuk mengklasifikasikan jenis penyakit pada tanaman padi menggunakan metode *Backpropagation* dengan optimasi *Firefly* dapat dilihat pada Table 2.3.

Table 2.3 Bahan Penelitian

Bahan-bahan	Diperoleh dari
Data gejala penyakit pada tanaman padi	Di ambil dari pertanyaan dari petani
Bahasa Pemrograman	<i>Python</i>

2.9 Evaluasi

Evaluasi merupakan langkah untuk memastikan bahwa model dapat menggeneralisasi pola yang telah dipelajari dari data latihan ke data baru dengan tepat (Nurhikam *et al.*, 2024). Kemampuan klasifikasi di ukur dari nilai akurasi, *recall*, presisi, *f1-score*.

2.9.1 Akurasi

Akurasi digunakan untuk mengetahui seberapa baik sistem dapat mengklasifikasi data dengan benar (Nasution, Limbong and Ramadhan, 2020) (Sari and Suryono, 2024).

$$\text{Akurasi} = \frac{TP + TN}{(\text{Total numbers of sampel})}$$

2.9.2 Presisi

Presisi digunakan untuk mengetahui seberapa akurat suatu model dapat mengidentifikasi suatu sentimen (Sari and Suryono, 2024).

$$\text{Presisi} = \frac{TP}{TP + FP}$$

2.9.3 Recall

Recall digunakan untuk mengetahui seberapa baik model menemukan dan mengenali suatu sentimen (Sari and Suryono, 2024).

$$\text{Recall} = \frac{TP}{TP + FN}$$

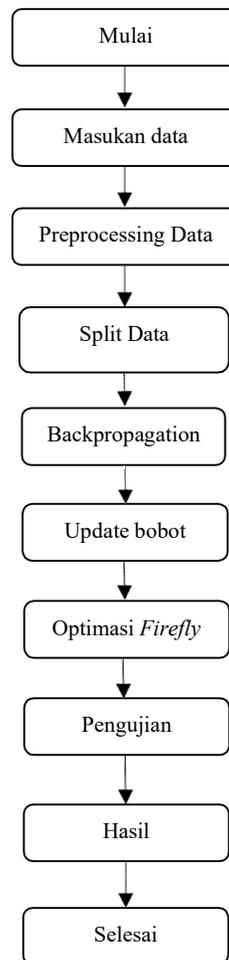
2.9.4 F1-score

F1-score digunakan untuk mengetahui perbandingan rata-rata presisi dan *recall* (Athallah, Azhar and Munarko, 2020).

$$F1 - score = 2 \times \frac{\text{Recall} \times \text{Presisi}}{\text{Recall} + \text{Presisi}}$$

2.10 Prosedur Penelitian

Sesuai dengan tujuan penelitian ini yaitu adalah mendeteksi penyakit pada tanaman padi. Adapun prosedur dalam membangunnya :



Gambar 2.2 Alur Prosedur Penelitian

Keterangan Diagram alir penelitian :

1. Di mulai dengan memasukkan data.
2. Preprocessing data. Beberapa tahapan di lakukan dalam preprocessing di antara nya pembersihan data, transformasi data, normalisasi, dan lain-lain.
3. Data *Splitting*. Membagi data menjadi data latih dan data uji.
4. *K-Fold Cross Validation*. Adalah metode data dibagi menjadi “K” nilai fold.
5. Backpropagation. Melakukan pengujian algoritma Backpropagation untuk melihat hasil kinerja yang di peroleh.
6. Update bobot. Setelah pengujian maka dilakukan pembaruan bobot untuk meningkatkan akurasi dan mengurangi kesalahan model.
7. Optimasi. Langkah model Jaringan Syaraf Tiruan yang telah di latih oleh metode *Backpropagation* lalu dioptimasi dengan algoritma *Firefly*.
8. Evaluasi. Proses untuk memastikan bahwa model dapat menggeneralisasi pola yang telah dipelajari dari data latihan ke data baru dengan tepat. Hasil pengujian yang akan di dapatkan adalah nilai akurasi, *recall*, presisi, *f1-score*