CHAPTER - V

IMPLEMENTATION OF PROPOSED ALGORITHM
5.1 Introduction

The framework of this expert system is successfully developed in ASP.Net. The C# platform of ASP.Net is used to design the user interface. Microsoft SQL Server is used to design knowledge base. Expert system framework and its components are tested with several cases by the farmers. Cases applied on implemented algorithm are represented. Following is technical specifications used for the development of the framework.

- **Operating System**: Microsoft Windows 7 Professional
- **Development Framework**: Microsoft Dot Net Framework 4.5
- **Development Tool**: ASP.Net (C# Programming Language)
- **Database**: Microsoft SQL Server 2008
- **Processor**: Intel i5 Processor 5th Generation
- **RAM**: 4 GB DDR2

The framework of crop disease diagnosis system consists of an inference engine and learning tool for disease diagnosis. The algorithm implemented in framework works in three cases of disease diagnosis.

- **Case I: Disease Diagnosis Using Exact Matching Rules**
  
  In first case, if exact matching rule found in the knowledge base then system follows that rule and produce output.

- **Case II: Disease Diagnosis Using Matching Threshold**
  
  In this case, if exact matching rule not found in the knowledge base then disease diagnosis is done by matching threshold. This case is further divided into another two sub cases as:
  
  - If only one disease found with rule matching percentage greater than variant threshold value then it is considered as optimized disease.
  
  - If more than one disease found with symptoms matching percentage greater than the variant threshold value then they are used to find optimal diseases using variant inertia weight.
Case III: Disease Diagnosis Using Added Rules

At the time of disease diagnosis by an implemented algorithm using case II, the system creates new rules based on provided symptoms. Next time when farmers input same symptoms for diseases diagnosis, learning tool optimizes diseases based on these added rules.

5.2 Implementation of Framework

Implementation of framework considering above three cases is explained using examples.

5.2.1 Disease Diagnosis Using Exact Matching Rules

Farmer supplies symptoms to the system. Supplied symptoms are used to create a tag. Length of the tag is equal to a total number of symptoms in the knowledge base. The input tag is used to compare existing rules in the knowledge base. If the exact matching rule found in the knowledge base then system follows that rule and displays appropriate disease. Diseases found in this case are based on expert’s knowledge.

Figure 5.1: Selection of Symptoms to Obtain Possible Diseases
Number of Symptoms Selected - 1
Selected Symptom(s) - 1) S2 [Dark brown spots with borders on leaf fruit or stems].

Disease Found Using Exact Matching Rule - Yes

Number of Feasible Diseases - 1
Number of Optimized Diseases - 1
Optimized Disease - 1) D2 [Bacterial Spot].

Number of Intensive Disease(s) - 0

Number of Intensive Disease(s) - 0

Farmers provide symptoms to the system as shown in figure 5.1. Diagnosed diseases are displayed as shown figure 5.2.
5.2.2 Disease Diagnosis Based on Matching Threshold
If exact matching rule not found in the knowledge base then system diagnose diseases based on matching percentages of rules.

5.2.2.1 Disease Diagnosis Using Variant Threshold Value
If only one disease found with disease matching percentage greater than the variant threshold value then it is displayed as optimized disease.

Number of Symptoms Selected - 1
Selected Symptom(s) - 1) S1 [Small white spots on leaves, fruits that turn necrotic].
Disease Found Using Exact Matching Rule - No
Found Using Variant Threshold Value - Yes
Number of Feasible Diseases - 1
Number of Optimized Diseases - 1
Optimized Disease - 1) D2 [Bacterial Canker].
Number of Intensive Disease(s) - 0
Number of Intensive Disease(s) - 0
1) D2 [Bacterial Canker]

T1= Spray: Streptoclyne 6 gm/40 ltr + Copper Hydroxide 1.5 gm/ltr
T2= Spray: Kasu-B 1 ml/ltr + Copper OxyChloride 2 gm/ltr
T3= Spray: Bordo Liquid 2.5 gm/ml/ltr (Copper Sulphate 200 gm + Lime 200 gm in 100 ltr water)
5.2.2.2 Disease Diagnosis Using Variant Inertia Weight

If more than one disease found with disease matching percentage greater than the variant threshold value then they are used as feasible diseases. The feasible diseases are then used to find optimized diseases. Maximum optimization value and minimum optimization value obtained from feasible diseases are used to find final optimal diseases.

All diseases in the knowledge base are initialized with their random position and velocity vectors using maximum optimization value and minimum optimization value. Fitness value of each disease is evaluated. If the fitness value of disease is better than its personal best value then its current fitness value becomes its new personal best value. The best fitness value of all personal best values of disease is selected and it is set as global best value. The personal best value and global best value are used to calculate the velocity of each disease. The inertia weight is also used to calculate the velocity of each disease. To balance exploration and exploitation by updating the previous velocity of diseases, the researcher has used variant inertia weight which gives the best performance as compared with fixed inertia.
weight. The calculated velocity of each disease is used to update the position of disease and the disease best position is calculated.

Finally, the diseases whose optimization value is greater than best position are displayed as optimized diseases to the farmers. The optimized diseases in this case are displayed as diseases found by optimization.

**Number of Symptoms Selected** - 3

**Selected Symptom(s)**

- 1) S43 [Aphids found on leaves].
- 2) S48 [Immature and adult whiteflies colonize the underside of leaves causes stunting and defoliation].
- 3) S56 [Fruit appear dull and wrinkled, and tend to ripen prematurely]

**Disease Found Using Exact Matching Rule** - No

**Disease Found Using Variant Inertia Weight** - Yes

**Number of Feasible Diseases** - 12

**Number of Optimized Diseases** - 3

**Optimized Disease**

- 1) D23 [Aphids]
- 2) D27 [Whiteflies]
- 3) D34 [Beet Curly Top].

**Number of Intensive Disease(s)** - 1
Implementation of Proposed Algorithm

Intensive Disease(s) - 1) D23 [Aphids].

Number of Similar Treatment Disease(s) - 2

Similar Treatment Disease(s) - 1) D23 [Aphids]

2) D27 [Whiteflies].

Treatment(s)
1) D23 [Aphids]

T1= Spray: Confidor 0.5 ml/ltr + Neem Oil (10000 ppm) 1ml/ltr

T2= Spray: Astaf 1 gm/ltr

T3= Spray: Rogor 2 ml/ltr

2) D23 [Whiteflies]

T1= pray: Record 0.5 gm/ltr + Roger 2 ml/ltr

T2= Spray: Confidor 0.5 ml/ltr + Neem Oil (10000 ppm) 1ml/ltr

T3= pray: Triozophos 1 ml/ltr

Here, Treatment T1 of D23 (Aphids) is Similar to T2 of D27 (Whiteflies).

Information about intensive diseases is shown at the top and it is highlighted with red background color. It is helpful to alert the farmers to take necessary and immediate action on it.

Information about same treatments diseases is displayed with same background colors. It is helpful for the farmers to identify diseases with the same treatment. It reduces the cost of disease management.
5.2.3 Learning Tool

At the time of disease diagnosis by an implemented algorithm, the system creates new rule based on provided symptoms. Newly generated rule gets added in the knowledge base. Next time when farmers provide same symptoms for disease diagnosis, these added rules used for diagnosis. Following information of diagnosis process is captured automatically.

- Disease ID
- Input Tag
- Disease Intensity
- Disease Matching Percentage
- Treatment1
- Treatment2
- Treatment3

This information is automatically stored in the knowledge base. Thus it creates new rule.
Consider Example of 5.2.2.2

<table>
<thead>
<tr>
<th>Number of Symptoms Selected</th>
<th>- 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selected Symptom(s)</td>
<td>1) S43 [Aphids found on leaves].</td>
</tr>
<tr>
<td></td>
<td>2) S48 [Immature and adult whiteflies colonize the underside of leaves causes stunting and defoliation].</td>
</tr>
<tr>
<td></td>
<td>3) S56 [Fruit appear dull and wrinkled, and tend to ripen prematurely]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Disease Found Using Exact Matching Rule</th>
<th>- No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disease Found Using Matching Threshold</td>
<td>- No</td>
</tr>
<tr>
<td>Disease Found by Added Rules</td>
<td>Yes</td>
</tr>
<tr>
<td>Number of Feasible Diseases</td>
<td>- 12</td>
</tr>
<tr>
<td>Number of Optimized Diseases</td>
<td>- 3</td>
</tr>
<tr>
<td>Optimized Disease</td>
<td>1) D23 [Aphids]</td>
</tr>
<tr>
<td></td>
<td>2) D27 [Whiteflies]</td>
</tr>
<tr>
<td></td>
<td>3) D34 [Beet Curly Top].</td>
</tr>
<tr>
<td>Number of Intensive Disease(s)</td>
<td>- 1</td>
</tr>
<tr>
<td>Intensive Disease(s)</td>
<td>1) D23 [Aphids].</td>
</tr>
<tr>
<td>Number of Similar Treatment Disease(s)</td>
<td>- 2</td>
</tr>
</tbody>
</table>
Similar Treatment Disease(s)

- D23 [Aphids]
- D27 [Whiteflies].

Matching Threshold:

<table>
<thead>
<tr>
<th>Disease</th>
<th>Matching Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>D23 [Aphids]</td>
<td>95.52%</td>
</tr>
<tr>
<td>D27 [Whiteflies]</td>
<td>97.01%</td>
</tr>
<tr>
<td>D34 [Beet Curly Top]</td>
<td>95.52%</td>
</tr>
</tbody>
</table>

The system is not always depending on the expert for new rules. The system creates new rules based on the applied cases. If similar cases are given for disease diagnosis then system recalls these added rules to provide solutions with same diseases as well as same methods of treatment.

Figure 5.5: Disease Diagnosis Using Added Rules
5.3 User Interface

The framework of this crop disease diagnosis expert system is web-based and it is developed in ASP.Net (C#). The system provided user-friendly interface which facilitates in input as

A) Farmer can provide symptoms to the system by selecting it from displayed list.
B) Farmer can search symptoms by known attributes.
C) Farmer can forecast symptoms using current weather conditions such as temperature, humidity, and their crop stage.

A) Selection of Symptoms from Displayed List

Here a list of all available symptoms displayed. Farmer selects symptoms from that list to search possible diseases. Photos of symptoms are displayed along with textual information so that farmers can identify exact symptoms. Selection of symptoms is easy because checkbox control is provided for each symptom.

![Figure 5.6: User Interface I- Selection of Symptoms](image)
B) Searching of Symptoms Using Known Attributes of Symptoms

Here linear searching facility is provided. A farmer can enter known attributes (e.g. leaf color, root color, fruit infections etc.) of the symptom to search matching symptoms from the knowledge base. This linear search displays a list of matching symptoms.

![Figure 5.7: User Interface II- Searching of Symptoms Using Known Attributes](image)

C) Forecasting Using Weather States and Crop Stage

Here weather conditions of next four days including today are displayed. Weather conditions include; Temperature, Humidity, Rainfall etc are displayed separately for Kadgaon, Khanapur, Palus and Walwa tehsil areas (Weather data source: https://www.accuweather.com).

Farmers provide these weather conditions along with crop stage to predict possible symptoms. This forecasting displays a list of the possible symptom(s) based on the supplied conditions.
Figure 5.8: User Interface III- Forecasting Using Current Weather States

[Weather Data Source: https://www.accuweather.com]

D) Intensive Diseases
When farmers get the information of diagnosed diseases, this user interface provides facility to highlight intensive diseases from the list of optimized diseases. Information about intensive diseases is shown at the top and it is highlighted with red background color. It is helpful to alert the farmers to take necessary and immediate action on it.

E) Same Treatment Diseases
Information about same treatments diseases is displayed with same background colors. It is helpful for the farmers to identify diseases with the same treatment. It reduces the cost of disease management.
The implementation of the framework of crop disease diagnosis expert system is done effectively using latest technology which provides user-friendly interfaces and generates optimized results. The working of various cases applied on the framework is explained. The process of disease diagnosis using exact matching rules, variant threshold value, variant inertia weight, and added rules is represented.

<table>
<thead>
<tr>
<th>Observed Disease</th>
<th>Seen Symptoms</th>
<th>Disease Picture</th>
<th>Treatment #1</th>
<th>Treatment #2</th>
<th>Treatment #3</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aphids</td>
<td>Search other Symptoms of this Disease</td>
<td>Spray: Confidor 0.5 ml/ltr + Neem Oil (10000 ppm) 1ml/ltr</td>
<td>Spray: Astaf 1 gm/ltr</td>
<td>Spray: Rogor 2 ml/ltr</td>
<td>Disease Matched with 95.52% Probability</td>
<td></td>
</tr>
<tr>
<td>Whiteflies</td>
<td>Search other Symptoms of this Disease</td>
<td>Spray: Record 0.5 gm/ltr + Rogor 2 ml/ltr</td>
<td>Spray: Confidor 0.5 ml/ltr + Neem Oil (10000 ppm) 1ml/ltr</td>
<td>Spray: Triazophos 1 ml/ltr</td>
<td>Disease Matched with 97.01% Probability</td>
<td></td>
</tr>
<tr>
<td>Beet Curly Top</td>
<td>Search other Symptoms of this Disease</td>
<td>Transplant virus-free seedlings</td>
<td>Remove and Destroy affected plants</td>
<td>Spray of Insecticide like Rogor, Karate, Nuvan, Ekalux, Record regularly with 5 days</td>
<td>Disease Matched with 95.52% Probability</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 5.9: Intensive and Similar Treatment Diseases**

### 5.4 Summary
The implementation of the framework of crop disease diagnosis expert system is done effectively using latest technology which provides user-friendly interfaces and generates optimized results. The working of various cases applied on the framework is explained. The process of disease diagnosis using exact matching rules, variant threshold value, variant inertia weight, and added rules is represented.