Diseases warning system for spinach based on genetic-BP network algorithm

Zhang Lingxian[1,2], Gu Dongyue[1], Liu Chundi[1], Zou Chunyu[1], Li Xinxing[1,2*]

(1. *China Agricultural University, Beijing 100083, China*;

2. *Key Laboratory of Agricultural Informationization Standardization, Ministry of Agriculture, Beijing 100083, China*)

Abstract: A diseases warning model of forecasting spinach common diseases was designed, which could provide a low-cost solution to inform users before plants getting disease. Through systematic learning on a lot of spinach diseases knowledge, the environmental parameters of lowest temperature, highest temperature, average temperature, lowest humidity, highest humidity and average humidity are defined as input parameters. Besides, downy mildew index, leaf spot index, black spot index, blight index and anthracnose index are defined as output parameters of BP network. The BP network's number of hidden layer nodes was determined by genetic algorithm. The improved model has better accuracy and faster convergence speed. A complete system has been developed, consisting of user subsystem, data management subsystem, query subsystem and forecasting subsystem. After training, the system can provide index number of each common disease according to the input environmental parameters.

Keywords: genetic-BP network algorithm, spinach, diseases warning system

1 Introduction

Spinaches have a vast market in China and spinaches are grown very widely. With the demand for spinaches consumption increasing, more and more farmers grow spinaches in greenhouses. Diseases of spinaches have become the main factors that obstacle the spinaches productions. Moreover, diseases control activity is an important research problem of agricultural experts.

Nowadays, the major method to control spinach diseases is pesticide spraying while excess pesticide may cause problems of damaging humans' health and ecological balance. Diseases control should base on preventive methods, so it is of great importance to accurately predict diseases, and fairly use pesticide to achieve diseases control effectively. ANN (Artificial Neural Network) is a kind of intelligent algorithm which is suite to describe nonlinear phenomena and resolve complex problems (Worner et al., 2012), so more and more scientists built models to predict crop disease by ANN (Francl et al., 2004).

De-Wolf (1997) classified infect period of wheat tan spot by neural networks in last 1990s (Liu et al., 2008). Morbidity, temperature, relative humidity, precipitation, and leaf wetness duration were defined as input variables. They compared the results of different methods of BP networks, generalized regression neural networks and logistic regression. Results indicated that BP networks could provide more accurate forecast. Liu (2012) used neural networks and principal components analysis to evaluate the severity of rice rhizoctonia. Yong (2016) constructed a dynamic predict model, based on BP neural networks to predict forest disease and insect and rat pest. Shivling (2011) developed an artificial neural network based on system for predicting the risk of powdery mildew in Picorrhiza kurrooa, by putting temperature and wetness duration so as to predict the level of risk at that particular value of the input. Shi (2008) presented a new prediction model by principal components analysis (PCA) and back propagation artificial neural network (BP-ANN)

methods to forecast wheat sharp eyespot. Lankinvega (2008) predicted the fall of barley yellow dwarf virus based on the theory of artificial neural network model, and compared with the results of the multiple regression analysis model. The error of artificial neural network is better than that of multivariate regression analysis. Based on the fusion of RBF neural network and WebGIS, Li (2005) established apple pest disease forecasting system, which can be used to predict diseases and pests, and spread the information network geographic with information system.

Aiming at forecasting different kinds of spinaches diseases, an improved method was chosen which based on the genetic and BP network algorithm and defined environmental parameters as input variables.

2   System design

2.1   Overall design

2.1.1   Work flow

Agricultural plants diseases knowledges were integrated first to determine the primary factors that cause spinaches diseases. Then BP network was chosen to establish a warning model which can be improved by genetic algorithm. The improved algorithm has faster convergence and better accuracy, and it is more suited to

establish warning system. After data training, warning system can be used to forecast spinach diseases. The complete work flow is shown in Figure 1.
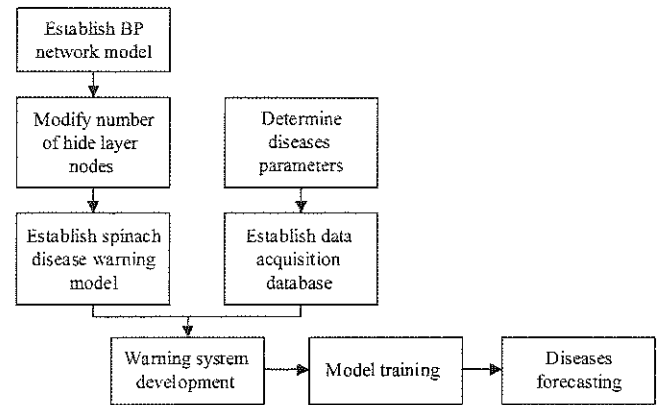

Figure 1   Work flow

2.1.2   System structure and functions

Warning system is composed of user subsystem, data management subsystem, query subsystem and forecasting subsystem. The structure shows in Figure 2.

2.2   Database design

The input and output data are stored in SQL Server 2008 database. And the main two tables were shown in Figure 3 and Figure 4. The data in table sample are used to train network. Table real Time Data is jointly maintained by data acquisition system and diseases warning system.
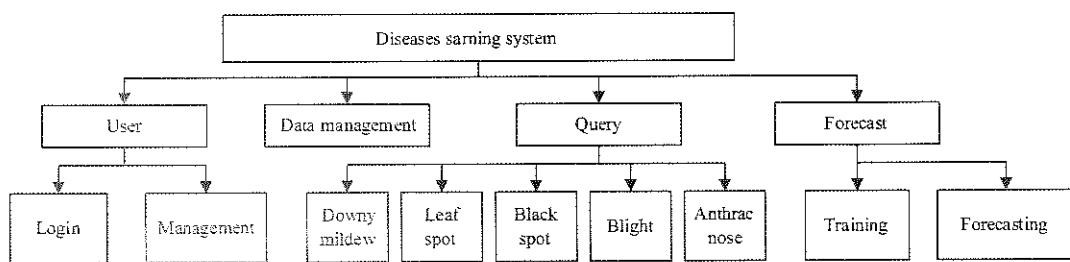

Figure 2   System structure

Figure 3   Example table

Figure 4   Table real Time Data

2.3   Common diseases of spinach

2.3.1   Downy mildew

(1) Symptoms and signs

In the early stages of downy mildew (Yong, 2006), leaf appears small spots without obvious edge, and then they spread out into big amorphous spots. Then the surface of the leaves back appears gray and white layers of mold which become purple gray at last stage. Downy mildew often causes diseased leaves yellow and rot.

(2) Pathogenic regularity

This disease is caused by fungi. Moreover, plants are more likely infected in the environment in which temperature is 10°C and relative humidity is more than 90%. Besides, poor ventilation and excess irrigation both can aggravate state of disease (Yu et al., 2008).

2.3.2   Leaf spot

(1) Symptoms and signs

This disease damages leaves, and the lower leaves infect earlier. In the initial stage, infected parts lost their green color and the edge of these parts change to light brown. Then the spots whose shapes are nearly round spread out and become white and their edges are quite obvious. The diameters are 0.5-3.5 millimeters. Gray and brown mold can be found when air humidity is high.

(2) Pathogenic regularity

This disease is caused by fungi. Moreover, plants are more likely infected in the environment of high temperature and humidity. In addition, the seriously diseased spinaches often can be found in depression and wind scoop land.

2.3.3   Black spot

(1) Symptoms and signs

This disease mainly damaged leaves. Spots appear round or nearly round whose color is light brown to brow. Their diameters are in 2-6 millimeters range. Black molds attach on it.

(2) Pathogenic regularity

Black spot disease is mostly caused by fungi. Spinaches are more likely infected in the high precipitation year and high humidity environment.

2.3.4   Blight

(1) Symptoms and signs

If the spinach were infected blight, their leaves would lose their sheen. Then their mesophyll changes to yellow and root wither. Moreover, infected plans are obviously dwarfed and will appear chlorotic quickly in high temperature.

(2) Pathogenic regularity

The pathogen can attach on infected plant remains to survive summers and winters. And pathogens are disseminated by rain and irrigation water. Spinaches are more likely infected in warm and wet soil.

2.3.5   Anthracnose

(1) Symptoms and signs

Infected leaves appear light yellow spots that spread out into gray and amorphous spots. There are wheeled stripes and black small dots on them.

(2) Pathogenic regularity

Anthracnose disease is mostly caused by fungi. Mycelium can attach on infected plant remains or seeds to survive winters and be disseminated by rain and insects. Spinaches are more likely infected in rainy and high planting density environment.

2.4   Warning model based on genetic algorithm and BP neural network

2.4.1   BP neural network

BP network is a structure based on gradient descent optimization. It is with an algorithm of supervised learning (Yuan, 1999). The learning process of BP network contains the following two phases: In the first phase, the neurons transmit signals forwards. Neutrons in each layer influence only the structures of neurons in the next layer. When the expected output is different from the actual output, namely no expected value is reached, the system will enter the second phase and the neurons transmit signals backwards.

The BP algorithm transmits error signals backwards exactly along the original route and adjusts the weights and thresholds of each layer under the principle of gradient descent of error. Through adjusting the thresholds of neurons in each layer, the modified output will approach the expectations. After being adjusted, the thresholds will be transmitted forwards. Then the expected output will be compared with the actual output.

As described above, the process continues ad infinitum until the output and the expected values reach

an acceptable range or training times exceeds a preset value. The structure of BP neural algorithm is shown in Figure 5.
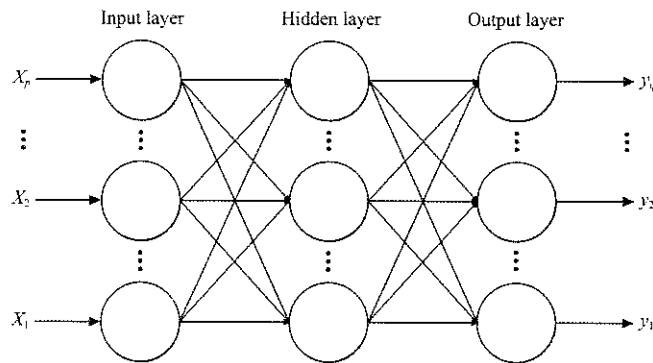

Figure 5   BP network structure

According to the research at this stage, a BP neural network, which contains two hidden layers, can approximate to any continuous function and each arbitrary order derivatives in arbitrary precision. Besides, a neural network, which also possesses multi-layer, can better approximate to the nonlinear function, yet with more computation and it has the possibility to be unstable. For BP neural network, if the settings of the number of hidden layer nodes are reasonable, single hidden layer can achieve the same effect. To reduce the complexity of model calculation, we use the BP neural network with 3 hidden layers here.

A calculate process of BP neural network is below. Let p neurons be in the input layer and h neurons be in the hidden layer nodes, and q neurons be in the output layer. Also let the input value of sample be $(x_1, x_2, ..., x_q)$, the output value of sample be $(S_1, S_2, ..., S_q)$, the output value of hidden layer nodes be $(O_1, O_2, ..., O_h)$, the output value of BP network $(y_1, y_2, ..., y_q)$, the weighting between input layer and hidden layer nodes be, the weighting between hidden layer and output layer nodes be w_{ij} (i=1, 2, ..., p; j=1, 2, ..., h), and the threshold of hidden layer and output layer neurons be v_{jk} (i=1, 2, ..., h; j=1, 2, ..., q), and we choose Sigmoid function as activation function, calculate the output value of hidden layer nodes:

$$H_j = \sum_{i=1}^{p} w_{ij}x_i + \theta_j, j = 1,2,...,h \qquad (1)$$

$$O_j = f(H_j) = \frac{1}{1 + e^{-H_j}} \qquad (2)$$

Similarly, use the output value of hidden layer nodes as the input value of output layers nodes, and figure out the output value of BP neural network according to the following Equations (3) and (4):

$$I_k = \sum_{j=1}^{h} f(H_j)v_{ij} + \phi_k, k = 1,2,...,q \qquad (3)$$

$$y_k = f(I_k) = \frac{1}{1 + e^{-I_k}} \qquad (4)$$

Calculate the actual output y_k according to the samples in sample set one by one, also the error metric E. Then figure out the error metric about the r sample of the network according to the following Equations (5) and (6), and the error metric about the whole sample set of the network:

$$E_p = \frac{1}{2} \sum_{k=1}^{q} (y_{rk} - S_{rk})^2 \qquad (5)$$

$$E = \sum_{p=1}^{m} E_p \qquad (6)$$

Use the error metric of the output layer to adjust the weight matrix of output layer, also to estimate the error of the hidden layer. Then use the error of the hidden layer to estimate the error of the input layer. All the layers' error estimates come out in this way too, and the estimation can be used to modify the weight matrix, and then form the process of letting the error showed in the output port transfer to the input port stepwise through the direction which is opposite to the direction input signal goes. According to δ earning and control way, adjust the weighting. Below are the weighting adjusting formulas. In the formulas, η is the learning factor, t is the training number, w_{kj} and v_{kj} represent the weightings which are modified after the end of this training, and δ is the error variety between input and output.

$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_{ri}O_j \qquad (7)$$

$$v_{jk}(t+1) = v_{jk}(t) + \eta \delta_{rk}x_i \qquad (8)$$

Verify the error between sample output and network output, if it is less than $\varepsilon(\varepsilon<0.001)$, then BP neural network training process is over, otherwise continue to adjust the weighting or threshold, and then calculate the error again until it becomes less than ε ($\varepsilon<0.001$).

To enhance the training accuracy of the neural network, using the method of increase the neuron

numbers is much easier than the method of adding hidden layers. Thus, network design selected the best neuron numbers, on the premise of as less layers as possible. Theoretically, there is no definite rule for best hidden layer nodes. Yet if there are few hidden layer neurons, fault tolerance will be worse, therefore it cannot recognize the sample which is exclude from training sample sets. Too many neurons in hidden layer will not only cause the learning time stay too long, but also lower the learning ability in learning the new sample. The whole network performance becomes bad. Thus, it is very crucial to confirm the hidden layer nodes of BP network (Li et al, 2008).

2.4.2 Genetic algorithm and BP network

After the introduction of the genetic algorithm, genetic-BP neural network takes advantages of strong global search capability of genetic algorithms.

Preliminary study the network nodes of hidden layer, weights and thresholds for global rough precision through genetic algorithms, then position the optimal solution area and determine the optimal number of hidden layer nodes, like weights and thresholds population gathered in the solution space of a few parameters, and then use the BP neural network algorithm to search the smaller fine gradient solution space, finally get the optimal solution.

Genetic algorithm is a probabilistic search algorithm based on the basic idea of simulating the evolution of binary groups and composing the real strings. Specific operation is from all strings by a number of groups consisting of an initial starting of the current population of individuals cyclically selection, crossover and mutation process until the optimal solution can be received. Genetic algorithm flowchart program runs as follows:
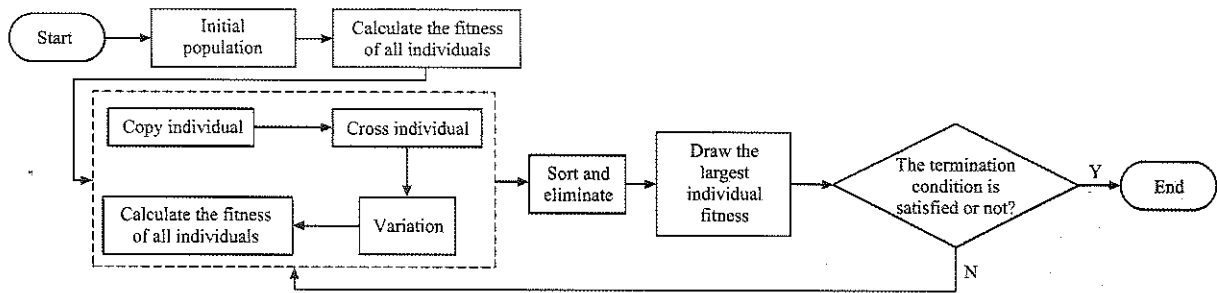

Figure 6　Genetic algorithm flowchart

Genetic algorithm can optimize BP neural network algorithm, including: encoding, decoding, determine the fitness function, genetic manipulating (natural selection, crossover, mutation, etc.). The specific description of this process is as below.

BP neural network weights and thresholds by using real-coded for hidden layer nodes with binary numbers, 0 represents the hidden layer node is invalid, and 1 represents effectiveness. According to the Kolmogorov theorem, the maximum empirical formula can be determined from the hidden layer nodes, where L is the maximum number of hidden layer nodes, n is the number of input layer nodes, o is the number of output layer nodes, and a is in the range of integers from 1 to 10.

To replicate the parent population first, thus creating the same part of the parent offspring. Then conduct crossover and mutation operations on the copied parent. Cross plays in the evolution is a very important role,

which is an essential step to optimize the search, while the population is to avoid premature convergence, and mutation is also necessary. Crossover and mutation operations are carried out, the need for real-coded and binary coded should be considered separately.

For binary encodings, only during the operation to produce a cross-cross point immediately before the boundary to the cross-point exchanges chromosome segment. Similarly, generate a random mutation point mutation during the course of the operation. n this variation of a certain mutation rate variation points, change 0 to 1 or change 1 to 0 (Wang et al., 2008).

For real-coded, the cross-based method uses a convex set theory to do crossover operation according to the Equation (9):

$$\begin{cases} y_i^{(1)} = a \times x_i^{(1)} + (1-a) \times x_i^{(2)} \\ y_i^{(2)} = a \times x_i^{(2)} + (1-a) \times x_i^{(2)} \end{cases} \qquad (9)$$

Where in $x_i^{(1)}, x_i^{(2)}, y_i^{(1)}, y_i^{(2)}$ respectively adjacent two adjacent i-parent and offspring chromosomes variable, a is the cross ratio of each chromosome, can take a random number in the range 0-1.

Variability is in Equation (10):

$$\begin{cases} y_i = y_i + \beta \times (a_i - y_i), & 0.5 \le \beta \le P_m \\ y_i = y_i + \beta \times (b_i - y_i), & \beta < 0.5, \beta \le P_m \end{cases} \quad (10)$$

y_i is variable i of variation in chromosomes; β is the random number in 0-1 range, it decides variation in direction; P_m is the mutation rate of the variable i; a_i, b_i respectively are the upper and lower weights and threshold value, the value of +1 and -1.

Fitness of the population through the computer all the chromosomes, chromosome number from the total number of people choose to 1/2 the evolution to the next generation, because the parent has copied before crossover and mutation, so eliminated after 1/2, the population does not vary of change.

After eliminating a new generation of populations by optimizing, returning to the second step, re-replication, crossover and mutation, select options. The cycle of generations knows the best fitness function difference is small or it reaches preset evolution algebra when the program ends. The type of fitness function is below:

$$F = a \times f_1 + b \times f_2 \quad (11)$$

And in which $0 < a, b < 1$ and $a + b = 1$, we select $a = 0.8$, $b = 0.2$, $f_1 = 1/(E+1)$, where E is the BP neural network error metric, $f_2 = L/q$, L is hidden layer node maximum number, q is the current number of hidden layer nodes.

2.4.3 Diagnosis warning model based on genetic algorithm and BP network

In this paper, the lowest temperature, highest temperature, average temperature, lowest humidity, highest humidity and average humidity are defined as input parameters. Besides, downy mildew index, leaf spot index, black spot index, blight index and anthracnose index are defined as output parameters of BP network. However, the experimental data are different in different groups, so it is necessary to normalize the experimental data, which contribute to network training speed. Normalization formulas are as follows:

$$X = \begin{pmatrix} x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16} \\ x_{21}, x_{22}, x_{23}, x_{24}, x_{25}, x_{26} \\ \vdots \\ x_{n1}, x_{n2}, x_{n3}, x_{n4}, x_{n5}, x_{n6} \end{pmatrix} \quad (12)$$

$$Y = \begin{pmatrix} y_{11}, y_{12}, y_{13}, y_{14}, y_{15} \\ y_{21}, y_{22}, y_{23}, y_{24}, y_{25} \\ \vdots \\ y_{n1}, y_{n2}, y_{n3}, y_{n4}, y_{n5} \end{pmatrix} \quad (13)$$

$$\begin{cases} x_{\min j} = \min(x_{1j}, x_{2j}, \cdots, x_{nj}), \\ x_{\max j} = \max(x_{1j}, x_{2j}, \cdots, x_{nj}), j = 1, 2, \cdots, 6; \\ y_{\min k} = \min(y_{1k}, x_{2k}, \cdots, y_{nk}), \\ y_{\max k} = \min(x_{1k}, x_{2k}, \cdots, y_{nk}), k = 1, 2, \cdots, 5; \\ x_{inij} = \dfrac{x_{ij} - x_{\min j}}{x_{\max j} - x_{\min j}} \\ y_{outj} = \dfrac{y_{ik} - y_{\min k}}{y_{\max k} - y_{\min k}}, i = 1, 2, \cdots, n \end{cases} \quad (14)$$

The normalized data can be used to train network according to the aforementioned method. The trained genetic-BP network can finally warn spinach diseases and forecast the disease type.

3   System implementation

This system is developed with C# and Matlab combined programming techniques. Mixed programming techniques are implemented by calling matlab dynamic-link libraries in C#.
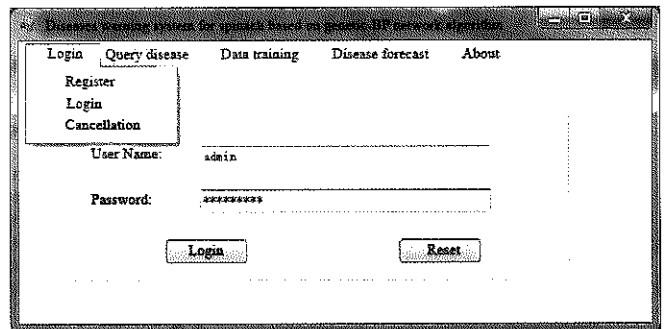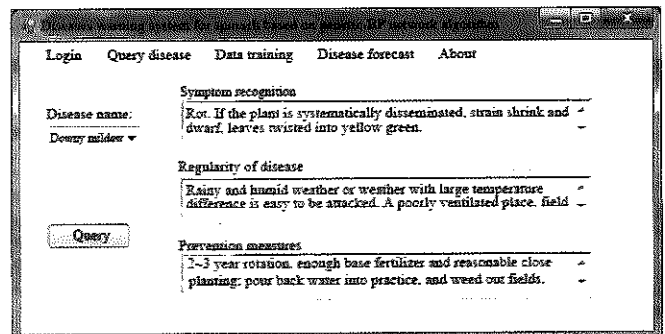
Figure 7   Login interface
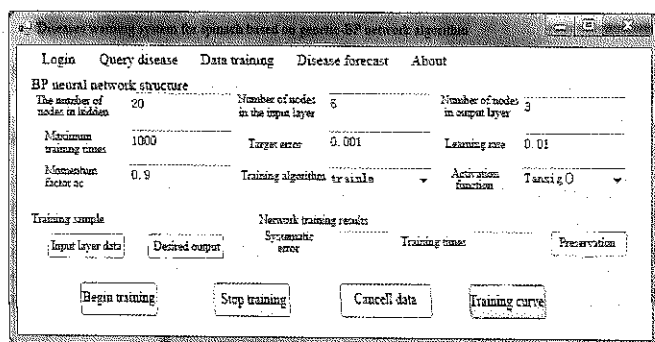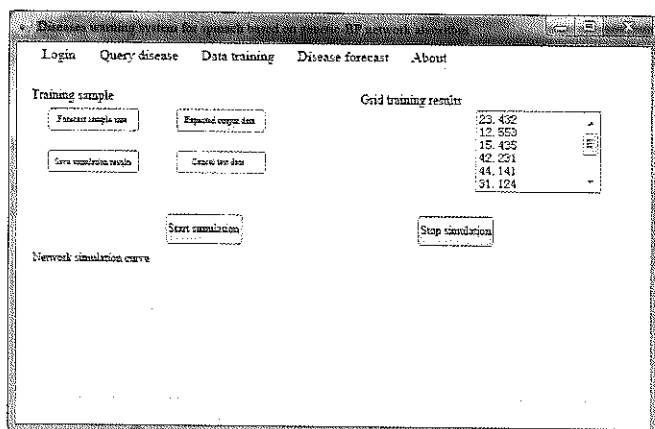
Figure 8   Query interface

Figure 9   Train interface

Figure 10   Simulation interface

4   Conclusion

This paper introduces a design of diseases forecasting model based on genetic-BP network algorithm. By way of sample data training, the model can forecast different kinds of diseases quickly and accurately. Basing on this model, the warning system provides a possible solution to ensure spinaches production in a low–costs way. Moreover, this method can be used in other agricultural plants diseases warning.

Acknowledgments

[References]

[1]   Dalla, M. A., R. D. Magarey, and S. Orlandini. 2005. Modeling leaf wetness duration and downy mildew Simulation on grapevine in Italy. *Agricultural and Forest Meteorology*, 132(1-2): 84–95.

[2]   Francl, L. J. 2004. Squeezing the turnip with artificial neural nets. *Phytopathology*, 94(9): 1007–12.

[3]   De-Wolf, E. D., and L. J. Francl. 1997. Neural networks that distinguish infection periods of wheat tan spot in an outdoor environment. *Phytopathology*, 87(1): 83–87.

[4]   Lankinvega, G., S. P. Worner, and T. Daj. 2008. An ensemble model for predicting Rhopalosiphumpadi abundance. *Entomologia Experimentalis Et Applicata*, 129(3): 308–315.

[5]   Li, L., D. Li, Z. Zhou, Y. Zhang, and Z. Fu. 2008. Forecast of disease and pest in apple orchards based on fusion of radial basis function neural network and WebGIS. *Transactions of the Chinese Society for Agricultural Machinery*, 39(3): 116–117. (In Chinese with English abstract)

[6]   Li, P. C., and S. Y. Li. 2008. Learning algorithm and application of quantum BP neural networks based on universal quantum gates. *Journal of Systems Engineering and Electronics*, 19(1): 167–174.

[7]   Liu, Z. Y., J. F. Huang, and R. X. Tao. 2008. Estimating rice brown spot disease severity based on principal component analysis and radial basis function neural network. *Spectroscopic and Spectral Analysis*, 28(9): 2156–2160.

[8]   Michael, J., W. Susan, and P. Worner. 2012. Using artificial neural networks to predict the distribution of bacterial crop diseases from biotic and abiotic factors. *Computational Ecology & Software*, 2(1).

[9]   Shi, M. W. 2011. Application of BP neural network and PCA in prediction of plant diseases control. *Advanced Materials Research*, 219(220): 742–745.

[10]   Shivling, V. D., C. Ghanshyam, R. Kumar, S. Kumar, and R. Sharma. 2016. Prediction model for predicting powdery mildew using ANN for medicinal plant - Picrorhiza kurrooa. *Journal of the Institution of Engineers*, 1–5.

[11]   Wang, X., M. Zhang, J. Zhu, and S. Geng. 2008. Spectral prediction of Phytophthora infestans infection on tomatoes using artificial neural network (ANN). *International Journal of Remote Sensing*, 29(6): 1693–1706.

[12]   Wang, Y., W. Gang, and S. Feng. 2012. The dynamic model prediction study of the forest disease, insect pest and rat based on BP neural networks. *Journal of Agricultural Science (1916-9752)*, 4(3).

[13]   Yang, X. Q. 2006. The prevention and occurrence of spinach downy mildew. *Northern Horticulture*, 01: 82.

[14]   Yu, S., K. Zhu, and F. Diao. 2008. A dynamic all parameters adaptive BP neural networks model and its application on oil reservoir prediction. *Applied Mathematics and Computation*, 95(1): 66–75.

[15]   Yuan, Z. R. 1999. Artificial neural network and applications. Beijing: *Tsinghua University Press*, 68–80.