

Research on the Knowledge ES Tool Based on Binary Tree

Qiu Yun*, Fan Jingchao, Zhou Guomin

Agricultural Information Institute (AII) of the Chinese Academy of Agricultural Sciences (CAAS)

Beijing, 86-10-82106266

*Corresponding author, e-mail: qiuyun@caas.cn

Abstract

In the numerous agricultural information technologies, agricultural knowledge system has become an important part of the agricultural intelligent information technology (AIIT). This paper introduces the composition and working principle of the development tool for the binary tree-based knowledge system and uses one of the simplest examples to expound the operation method of the binary tree-based knowledge visualization editor and the development process of the accomplishment of the expert system for the apple disease diagnosis.

Keywords: *binary tree-based knowledge, knowledge system, knowledge representation, knowledge editing, visual editing*

Copyright © 2014 Institute of Advanced Engineering and Science. All rights reserved.

1. Problem Introduction

In summarizing the experiences in classification retrieval or diagnosis, the customary approach for the experts in the classification retrieval or plant disease and insect pest (or failure) diagnosis would be to adopt the classification retrieval key (also called the classification tree) to express them as a variety of specific solution strategies by use of the explicit "standard" mode. Generally speaking, the classification tree would certainly form numerous solutions (i.e. a multi-branched tree shape). But the most important characteristic of all these classification retrieval keys is to be possibly synthesized into one binary tree-based retrieval key (as shown in Figure 1) after the optimized analysis. The system of the binary tree-based retrieval key is usually composed of a series of questions (also called decision nodes). After one problem is answered, the judge would continue to answer the next question until identify the sole destination object at last (also called answer node or leaf node). Such a standardized knowledge can be further compiled into the computer programming language through the knowledge engineers, which would, after repeated debugging, modification and verification, be finally formed into an expert knowledge system that can solve certain specific problems in this field [1].

Therefore, the development process of an expert knowledge system that can be expressed in the binary tree-based mode could be simply described as shown in Figure 2.

Therefore, in such an expert knowledge system as expressed in the binary tree-based mode, we would select the rule-based knowledge representation method. That is to say, each node would be expressed by use of a fact. Because of the difference between the information type of the answer node and that of the decision node, it is required to use the different templates. The template of the answer node is: (node <name> answer <value>), in which <name> is the sole name of the node, <value> expresses the answer as stored on the node, and the word of "answer" is used to express the answer node. The template of the decision node is: (node <name> decision <question> <yes-node> <no-node>), in which <name> is also the sole name of the node, <question> is the question raised on this node, <yes-node> and <no-node> are respectively the nodes to be processed in making the affirmative or negative answer on the question, and the word of "decision" is used to express the decision node [2, 3]. The following is an actual example of the knowledge representation.

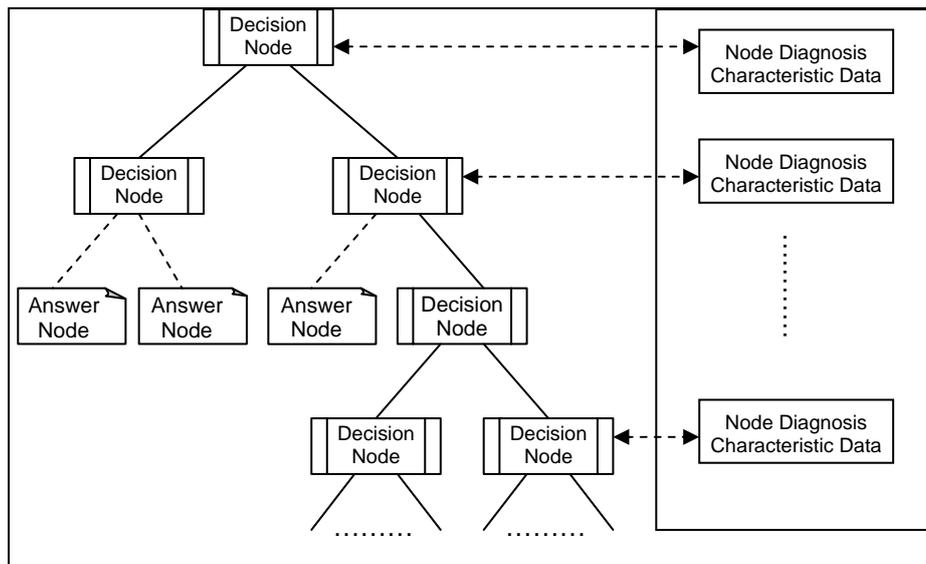


Figure 1. Structure of the Binary Tree-Based Retrieval Key

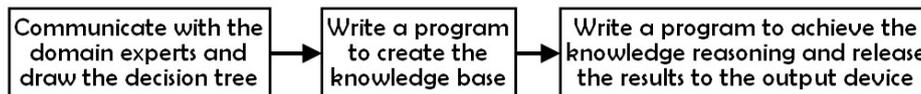


Figure 2. Development Process of the Expert Knowledge System

```
(deftemplate rule
  (multislot if)
  (multislot then))
(defrule propagate-goal ""
  (goal is ?goal)
  (rule (if ?variable $?)
    (then ?goal ?value))
  =>
  (assert (goal is ?variable)))
(defrule goal-satisfied ""
  (declare (saliency 30))
  ?f <- (goal is ?goal)
  (variable ?goal ?value)
  (answer ?text ?goal)
  =>
  (retract ?f)
  (format t "%s%s%n" ?text ?value))
.....
```

It can be seen from the above-mentioned development process that the workload of the program compilation is so huge that must be accomplished by knowledge engineers who have been professionally trained. Such development engineering would bring about three problems: Firstly, the human experts cannot understand such an expert knowledge base that is coded by the software of the knowledge engineers and cannot find and verify such wrong expert knowledge as bought in due to the communication misunderstandings. Secondly, the new addition and modification of the expert knowledge need the knowledge engineers to rewrite the code, in which the efficiency problem is actually secondary but the even bigger problem is likely to bring in the new errors due to the imprudence in rewriting the code. Thirdly, the reasoning machine cannot be reused.

Is it possible to develop such a tool that can “automatically” accomplish the encapsulated programming tasks that must be accomplished by the knowledge engineers? The answer is affirmative. The binary tree-based knowledge visualization editor is just such a software system that can accomplish this task.

2. Composition of the Binary Tree-Based Knowledge Visualization Editor

The binary tree-based knowledge visualization editor is composed of two parts (as shown in Figure 3): (1) knowledge editor, whose role is to edit the knowledge and manage to upload the knowledge onto the WEB server; and (2) knowledge reasoning components and knowledge service web page, whose role is to accept the users’ input, invoke the knowledge from the knowledge base for reasoning and return the reasoning results to the users in the form of web pages.

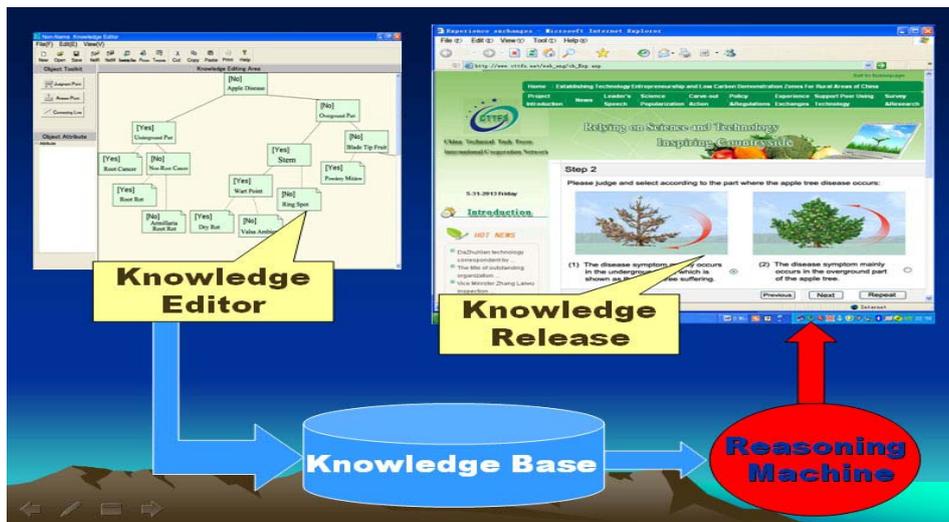


Figure 3. Structural Chart of the Binary Tree-Based Knowledge Visualization Editor

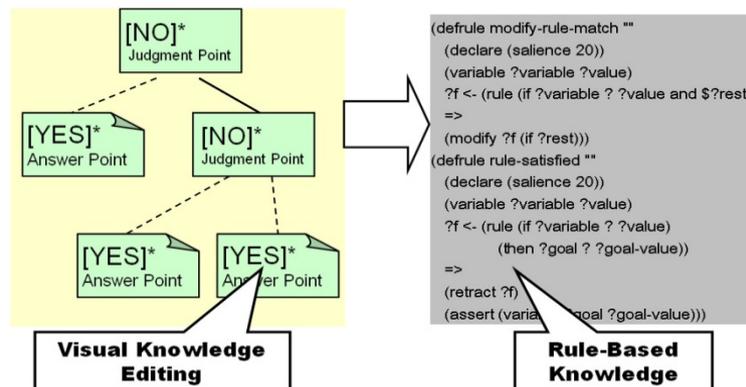


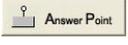
Figure 4 Working Principle of the Binary Tree-Based Knowledge Visualization Editor

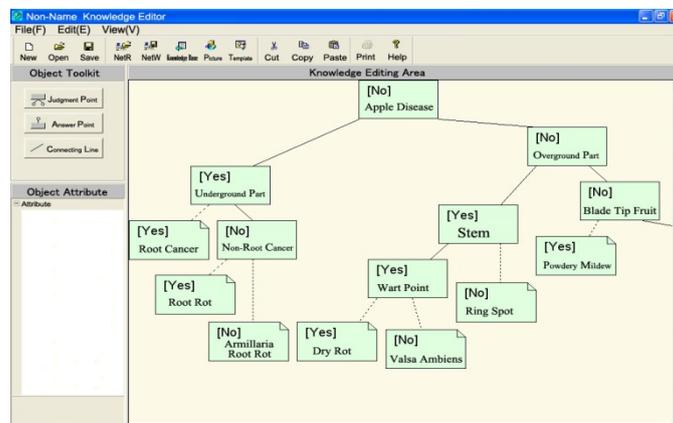
The binary tree-based knowledge visualization editor consists of three main functions: (1) visual knowledge editing function, which can make it convenient to edit the knowledge of the books or experts’ minds; (2) easy-to-use knowledge release function, through which the users can, with no need to write programs, use such knowledge reasoning components and corresponding ASP script as provided by this tool to release the knowledge onto the Internet;

and (3) flexible web page customization function, which can make it possible to customize the corresponding web interface according to the need and develop the personalized knowledge service system.

3.Working Principle of the Binary Tree-Based Knowledge Visualization Editor

The “Binary Tree-Based Knowledge Visualization Editor” adopts the visual editing method, in which the users need only to represent the knowledge by use of some icons and they need only to edit the icons to accomplish the editing of the knowledge. The “Binary Tree-Based Knowledge Visualization Editor” can automatically “translate” the icon-represented knowledge into the rule representation-based knowledge (Figure 4).

In the knowledge editor (Figure 5), according to the characteristics of the binary tree-based knowledge, we have designed three object tools: “Judgment Point” icon , “Answer Point” icon  and “Connecting Line” icon . By means of clicking these three object tools, you can draw the structural chart of the binary tree-based retrieval key in the knowledge editing area. The diagnosis characteristic data of each node shall be edited and accomplished through the node interface editor (Figure 6).



H;p.i

Figure 5. Interface of the Binary Tree-Based Knowledge Visualization Editor

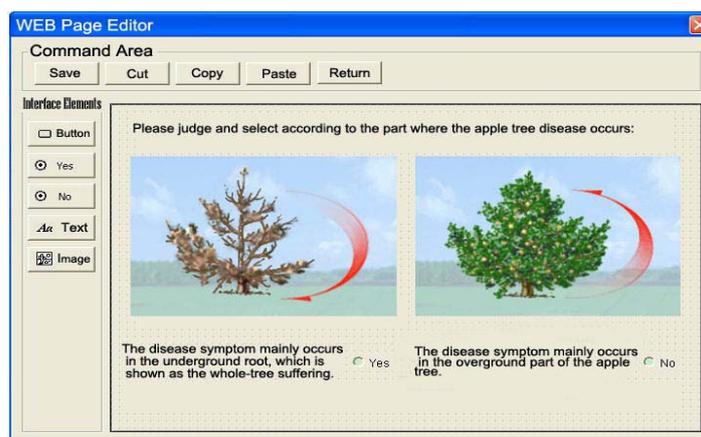


Figure 6. Interface of the Node Interface Editor

The structural chart of the binary tree-based retrieval key as drawn by the knowledge editor shall be stored in the KSF-format files (Figure 7), in which we take the form of a database

to store such knowledge data as represented by each diagnosis node [4]. Table 1 and Table 2 are the instances to describe the knowledge storage.

Paper:1000,800,0;

TD_Class:208,0,280,40, Apple disease,2,0,Arial,14680031,,no,6static|25|16|571|56| Please judge and select according to the part where the apple tree disease occurs:

||lbl_text|0|, static|30|75|303|163| The disease symptom mainly occurs in the underground root, which is shown as the whole-tree suffering. ||lbl_text|0|, static|27|188|300|294| The disease symptom mainly occurs in the overground part of the apple tree. ;

TD_Class:56,88,128;128, Underground root, 26,0, Arial,14680031,2, yes,6, static|25|16|571|56| If it is known that the disease mainly occurs in the underground root, which is shown as the whole-tree suffering, please select:

||lbl_text|0|, static|30|75|303|163| Cancer occurs in the root neck or in the root.

||lbl_text|0|, static|27|188|300|294| There is none of the above-mentioned symptom. ;

TD_note:0,144,72,184, Root cancer,49,0, Arial,14680031,26, yes,3, static|20|29|536|72| According to the symptom information you have provided, the system believes that the occurring disease should be:

||lbl_text|0|, static|34|133|273|184| Root cancer:

TD_line:72,128,56,144,,82,0,,,26,49,8,4,16, Arial,16777215,,,16,56,;;

TD_line:208,40,112,88,,93,0,,,2,26,8,4,0,Arial,16777215,,,0,56,;;

TD_Class:360,48,432,88, Overground part,127,0, Arial,14680031,2, no,6, static|25|16|571|56| If it is known that the disease mainly occurs in the overground part, please select:

||lbl_text|0|, static|30|75|303|163| The disease symptom mainly occurs in the stem of the fruit tree.

||lbl_text|0|, static|27|188|300|294| The disease symptom mainly occurs in the shoots, blades or fruits. ;

Figure 7. KSF Document Instance

Table 1. Structure of the Interface Library

Name	Type	Length	Description
qid	Text	50	Serial number of the nodes
qtitle	Text	50	Description of the problems in the knowledge base
mtitle	Text	50	Title of the interface
mtype	Text	50	Type of the interface
mContent	Remarks		Description files in the interface
mMedia	Text	50	List of the multi-media files in the interface
mMemo	Remarks		Notes of each interface

Table 2. Records in the Interface Library

Field	Value
qid	001
qtitle	Q275 ? (yes no)
mtitle	Please judge and select according to the part where the apple tree disease occurs:
mtype	1
mContent	<input type="rsdio" name=" The disease mainly occurs in the underground root, which is shown as the whole-tree suffering." left="100" top="180" right="200" bottom="200"> <input type="rsdio" name=" The disease symptom mainly occurs in the overground part of the apple tree." left="220" top="180" right="320" bottom="200">
mMedia	Anew.jpg Bnew.jpg
mMemo	In case of coincidence, please select "Yes" or otherwise select "No", and then press [Next Step].

When the "Binary Tree-Based Knowledge Visualization Editor" automatically "translates" the icon-represented knowledge into the rule representation-based knowledge, in order to improve the universality and extensibility of the inference system, we adopt the following conversion algorithm [5, 6].

Algorithm H:

H0: Let the numbers of each node on the binary tree be 1, 2, ..., N, in which the root node is given to be 1 and Queue L to be null.

H1: Place the root node into Queue L.

H2: Take out the first node from the queue and define it as R.

H3: Find out the two child nodes of Node R and number them as N1 and N2.

H4: For N1, the generative rule is made to be "if R then N1" and for N2, the generative rule is made to be "if R then N2".

H5: Find the child nodes of N1 and N2 and store them behind Queue L.

H6: Repeat H2 ~ H5 until Queue L becomes null.

By use of the above-mentioned conversion algorithm, the "Binary Tree-Based Knowledge Visualization Editor" can automatically convert the binary classification tree-based knowledge into the rule-based knowledge. By use of the rule-based knowledge, the reasoning machine can process the reasoning for the plant disease and insect pest diagnosis.

In the "Binary Tree-Based Knowledge Visualization Editor", we have also designed and realized such a reasoning machine as running in the HTTP server environment [7]. Since the HTTP server is stateless, each access to the server is independent and contextless. Therefore, we have realized such a stateless reasoning technology as based on the HTTP server by use of the step-by-step approach method.

If we let the system questioning the users be Q and the user answer be A, the question-and-answer would form a QA pair, which can be expressed as "<q> question </q> <a> answer " in the XML language.

The reasoning process is as shown in Figure 8. When the users start interacting with the XML server, the QA pair queue should be null, which is equivalent to no information provided by the users to the system. The XML server activates the expert system component to start the reasoning and run to such points as needed to interact with the users; and when the reasoning is ended, Question Q1 would be returned in the form of a message. If the users answer Question Q1, Answer A1 would be formed, when the content of the QA pair queue should be "<q> Q1</q> <a>A1". The users interact with the XML server by use of the queue of "<q> Q1</q> <a>A1", which is equivalent to some incomplete information provided by the users to the system. The XML server activates the expert system component to start the reasoning and run to such points as needed to interact with the users, in which if the answer to the question is in the QA pair queue, the answer would be directly extracted and the reasoning would continue, but if otherwise, the new Question Q2 would be generated; and when the reasoning is ended, Question Q2 would be returned in the form of a message. If the users answer Question Q2, Answer A2 would be formed, when the content of the QA pair queue should be "<q> Q1</q> <a>A1<q> Q2</q> <a>A2". The rest may be deduced by analogy, in which the information provided by the users to the system would be gradually increased and the system's reasoning path would become increasingly longer until the reasoning result is produced[8]. In the above-mentioned reasoning process, each interaction should be independent, so as to improve the running efficiency and ability of the server.

Interaction in Step n	Interaction in Step n+1
<pre><?xml version="1.0"> <request> <commandtype> rule </commandtype> <kdb> </kdb> <qa> <row value=1> <q>Q275?(yes no)</q> <a>yes </row> </qa> </request></pre>	<pre><?xml version="1.0"> <request> <commandtype> rule </commandtype> <kdb> </kdb> <qa> <row value=1> <q>Q275?(yes no)</q> <a>yes <q>Q27559?(yes no)</q> <a>no </row> </qa> </request></pre>

Figure 8. QA Queue Variation Diagram in the Reasoning Process

The “Binary Tree-Based Knowledge Visualization Editor” has realized the acquisition and reasoning of the binary tree-based knowledge in the Internet environment by use of the binary tree-based knowledge representation technology and the WEB-based stateless reasoning technology. In the HTTP bar of the Internet explorer, the users can directly use the “ks_run.asp?kdb= knowledge base name &title= system name” to run the plant disease and insect pest diagnosis system.

4. Development Example of the Expert System for the Apple Disease Diagnosis

The development process of the expert system for the apple disease diagnosis is as shown in Figure 9.

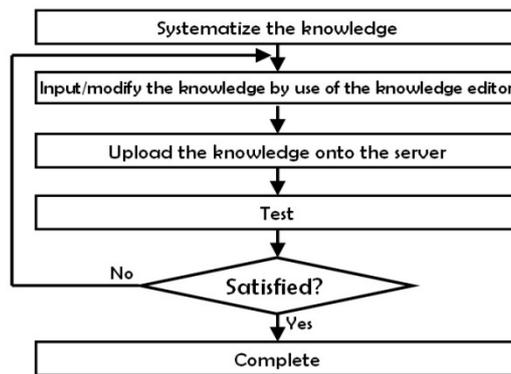


Figure 9. Development Process of the Expert Knowledge System for the Apple Disease Diagnosis

According to the above-mentioned process, we should first systematize the knowledge. Let us assume that the knowledge of the “Expert System for the Apple Disease Diagnosis” would be represented in Figure 10.

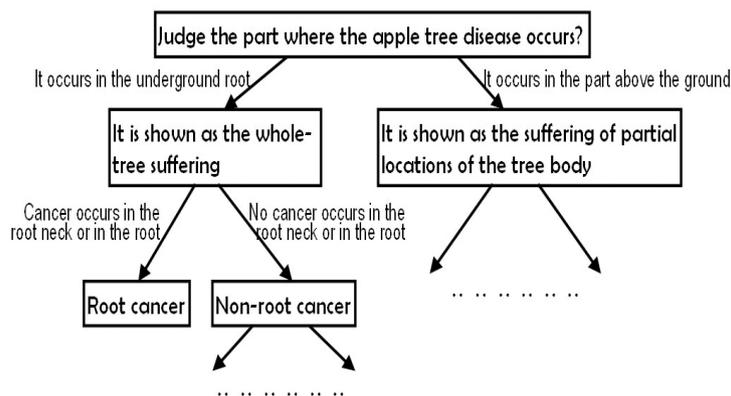
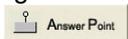


Figure 10. Binary Tree Diagram for the Apple Disease Diagnosis

According to the binary tree diagram for the apple disease diagnosis that has been systematized, we can use the “Knowledge Editor” (Figure 5) to draw the disease diagnosis diagram in the knowledge editing area by means of clicking the “Judgment Point” icon , “Answer Point” icon  and “Connecting Line” icon  in the object toolkit (on the top left of the window).

The icons in Figure 10 are not very similar to those in Figure 5. In Figure 10, it is needed to provide the users with the judgment information, that is to say, the users are allowed to select the part where the disease occurs. Whereas (in Figure 5) it should click the node of Judgment Point and press the right mouse button to open the Interface of the Node Editor (Figure 6) to edit and input these judgment information for the users to select.

After editing and completing all the nodes and node interface information, please click the “NETW” icon  to save the knowledge into the server and name the knowledge base as “pgph”. When it is successful to save the files and save the knowledge base interface, it is also necessary to generate the rule-based knowledge. Only by doing so, can the reasoning machine be used. Therefore, it is needed to click to generate the “Knowledge Base” icon  to generate the rule-based knowledge. If it is also successful to generate the knowledge base, the knowledge base of the “Expert System for the Apple Disease Diagnosis” has so far established and already uploaded onto the server.

Please open the browser and input the following address in the address bar: http://www.cart.net.cn/expert/am_ESMainRight.asp?kdb = pgph&title = apple disease diagnosis....., when the web page of the knowledge service system would be shown as in Figure 8. Click “Next” to start the reasoning.

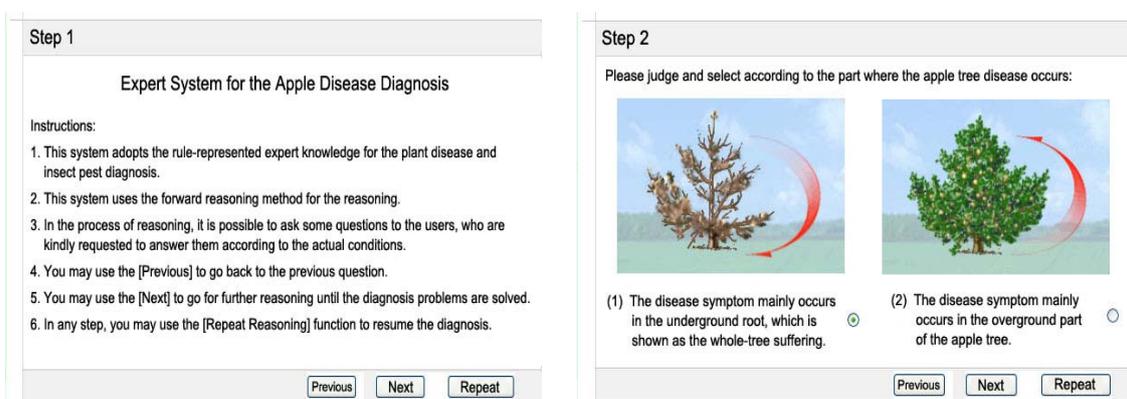


Figure 11. Expert System Interface for the Apple Disease Diagnosis

So far, the development of an “Expert System for the Apple Disease Diagnosis” has been accomplished.

5. Conclusion

By use of the “Binary Tree-Based Knowledge Visualization Editor”, it would be very convenient and efficient to accomplish the development of the expert system with various such specific solution strategies as expressed in the classification retrieval key or called classification tree.

It is shown after the preliminary application that this editor is of beautiful interface and convenient use and it can be conveniently used by the plant protection experts or failure diagnosis experts or classification retrieval experts after simple training to build the networked expert knowledge system for the classification retrieval or plant disease and insect pest (or failure) diagnosis, so as to greatly simplify the development cycle of the expert system for the classification retrieval or plant disease and insect pest (or failure) diagnosis.

Acknowledgements

Funding for this research is one of the National Hi-Tech Research and Development (863) Program’s Projects, titled “Model-Based Digital Management Platform for the Orchard and Canola Crop Production” (2013AA102405).

References

- [1] Wang Wenjie Ye Shiwei. Principle and Application of Artificial Intelligence, Beijing: Posts & Telecom Press. 2004; (3): 201-204.
- [2] Yin Chaoqing, Yin Hao. Artificial Intelligence Principle and Expert System, Beijing: China Water & Power Press. 2002; 34(2): 11-14.
- [3] Liu, Chengxiang, Cheng Li. Application of the Failure Tree in the Expert System. *Mechanical Science and Technology*. 2003; 22(5): 67-69.
- [4] Song Bin, Fang Gefeng, Liu Yi. Generation Technology of the Vector Graphic-Based Expert Knowledge Base. *Electronic Measurement Technology*. 2012; (9): 38-42.
- [5] Wang Zihao, Mo Yimin. Optimization Design of the Failure Tree Based on the Expert System for the Locomotive Fault Diagnosis. *Diesel Locomotive*. 2008; (4):1-4.
- [6] Shen Aihua, Tang Qiyi, Cheng Jiaan. Research and Application of the Forward and/or Backward Reasoning Based on the Binary Classification Retrieval Table. *Journal of Zhejiang University (Edition of Agriculture and Life Sciences)*. 2006; 32(6): 15-18.
- [7] Wu Haining. Realization of the WWW Online Consultation by Use of ASP Technology. *Medical Intelligence Service*. 2003; (4): 34-41.
- [8] Liu Yingtao. Acquisition of the Binary Tree-Based Knowledge and Research on the Reasoning Technology. *High Technology Letters*. 2007; (6): 23-29.