Available online at www.sciencedirect.com

ScienceDirect

Expert Systems with Applications 34 (2008) 833–844

Expert Systems with Applications

www.elsevier.com/locate/eswa

Dynamic EMCUD for knowledge acquisition

Shun-Chieh Lin [a,*], Shian-Shyong Tseng [a,b], Chia-Wen Teng [a]

[a] Department of Computer Science, National Chiao Tung University, Taiwan, ROC
[b] Department of Information Science and Applications, Asia University, Taiwan, ROC

Abstract

Due to the knowledge explosion, the new objects will be evolved in a dynamic environment. Hence, the knowledge can be classified into static knowledge and dynamic knowledge. Although many knowledge acquisition methodologies, based upon the Repertory Grid technique, have been proposed to systematically elicit useful rules from static grid from domain experts, they lack the ability of grid evolution to incrementally acquire the dynamic knowledge of new evolved objects. In this paper, we propose dynamic *EMCUD*, a new Repertory Grid-based knowledge acquisition methodology to elicit the embedded meanings of knowledge (embedded rules bearing on *m* objects and *k* object attributes), to enhance the ability of original *EMCUD* to iteratively integrate new evolved objects and new added attributes into the original Acquisition Table (AT) and original Attribute Ordering Table (AOT). The AOT records the relative importance of each attribute to each object in *EMCUD* to capture the embedded meanings with acceptable certainty factor value by relaxing or ignoring some minor attributes. In order to discover the new evolved objects, a collaborative framework including local knowledge based systems (*KBS*s) and a collaborative *KBS* is proposed to analyze the correlations of inference behaviors of embedded rules between multiple *KBS*s in a dynamic environment. Each *KBS* monitors the frequent inference behaviors of interesting embedded rules to construct a small AT increment to facilitate the acquisition of dynamic knowledge after experts confirming the new evolved objects. Moreover, the significance of knowledge may change after a period of time, a trend of all attributes to each evolved object is used to construct a new AOT increment to help experts automatically adjust the relative importance of each attribute to each object using time series analysis approach. Besides, three cases are considered to assist experts in adjusting the certainty factor values of the dynamic knowledge of the new evolved objects from the collection of inference logs in the collaborative *KBS*. To evaluate the performance of dynamic *EMCUD* in incrementally integrating new knowledge into the knowledge base, a worm detection prototype system is implemented.

Keywords: Knowledge acquisition; Dynamic *EMCUD*; Dynamic knowledge; Trend analysis; Worm detection

1. Introduction

As we know, knowledge based system (*KBS*) is an intelligent computer program that uses knowledge and inference procedures to solve problems that are difficult enough to require significant human expertise for their solutions, such as disease diagnosis, investment prediction, or computer science. Knowledge acquisition (KA) is one of the critical bottlenecks in developing a *KBS* for obtaining the knowledge of a special domain from domain experts.

Repertory Grid, based on Kelly's Personal Construct Theory (Kelly, 1955) which reports how people make sense of the world, could be used as an efficient knowledge acquisition technique in identifying different objects and distinguishing these objects in a domain. Although many KA systems and tools, e.g., *NeoETS* (Boose & Bradshaw, 1986), *AQUINS* (Boose & Bradshaw, 1987), *KITTEN* (Shaw & Gaines, 1987), *EMCUD* (Hwang & Tseng, 1990), *KADS* (Wielinga, Schreiber, & Breuker, 1992), have been proposed to rapidly build prototypes and improve the quality of the elicited knowledge of well-known objects by

[*] Corresponding author. Tel.: +886 3 5731966; fax: +886 3 5721490.
E-mail addresses: jielin@cis.nctu.edu.tw (S.-C. Lin), sstseng@cis.nctu.edu.tw (S.-S. Tseng).

0957-4174/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2006.10.041

domain experts with/without knowledge engineers based upon the Repertory Grid technique in the past twenty years, they are still focusing on acquiring a static gird information to generate static knowledge, which remains the same in the changing environment as time goes on.

However, with the changing environment as time goes on, new objects in many domains are incrementally evolved or developed due to the explosion of knowledge, resulting in the creation of new knowledge. Hence, knowledge can be classified as static knowledge and dynamic knowledge according to the stability of knowledge in a dynamic environment with the times. The static knowledge remains the same in the changing environment as time goes on. The dynamic knowledge, which may be updated or evolved, will be adapted in the changing environment with the times, since the previous knowledge may be degraded or upgraded in the near future. The knowledge evolution we proposed in this paper is the iterative process to acquire evolutional knowledge in a changing environment. Although many Repertory Grid-based KA methodologies can be used to acquire the dynamic knowledge of new evolved objects through rerunning their KA procedures, it is time-consuming and the relevant inference logs are seldom saved in the static grid. If some relevant log information can be analyzed to extract some useful information of discovering new objects, the dynamic knowledge of new evolved objects can be generated by experts.

EMCUD (Embedded Meaning Capturing and Uncertainty Deciding) (Hwang & Tseng, 1990) was proposed to elicit the embedded meanings of knowledge (embedded rules bearing on m objects $(O_1, O_2, \ldots, O_m)$ and k object attributes), which represents the information that domain experts take for granted but are implicit to the people who is not familiar with the application domain, and guide experts to decide the certainty degree of each embedded rule for extending the coverage of the original rules generated by Acquisition Table (AT) based upon Repertory Grid technique. Since the relative importance of each attribute to each object could be represented as Attribute Ordering Table (AOT) in *EMCUD*, some minor attributes can be relaxed or ignored to capture the embedded meanings with acceptable certainty factor (CF).

Assume some objects in O_1 class, which are classified by original rules of O_1, belong to the original object class (OO_1) of O_1; the other objects in O_1 class, which are classified by embedded rules of O_1, belong to the extended object class (EO_1) of O_1. However, some embedded rules may be with marginally acceptable CF values due to the weak suggestions of domain experts. In the age of the knowledge explosion, some objects might be evolved with the times and could be classified by the weak embedded rules of O_1 with weak CF values since some related ambiguous attributes (minor attributes) are ignored to classify these new evolved objects into O_1 class.

Although *EMCUD* can generate embedded rules to classify a new object into an original object class, it still has to be rerun to generate the dynamic knowledge using an updated AT and a new AOT acquired by experts if new objects are evolved. Moreover, the human experts are usually unaware of the occurrence the new evolved objects due to the lack of sufficient relevant information. Therefore, the dynamic *EMCUD* will be proposed in this paper to efficiently generate dynamic knowledge and iteratively integrate an AT increment and an AOT increment into the main AT and the main AOT, respectively. The AT increment records the new evolved objects and related new added attributes and the AOT increment records the evolved trend of all attributes to each new evolved object with the times in a dynamic environment.

In order to analyze useful evidence of the new evolved objects in a dynamic environment, we will propose a collaborative framework (including local *KBS*s and a collaborative *KBS*) to monitor the frequent inference behaviors of weak embedded rules and to trace the evolved behaviors of objects with the times from multiple *KBS*s for assisting experts in efficiently obtaining the dynamic knowledge. Each local *KBS* deploys a New Evolved Object learning (neo-learning) module to monitor the frequent inference behaviors of weak embedded rules to iteratively construct an AT increment. The AT increment could be created to record the relationships between new objects and new attributes after new objects are confirmed by experts without asking too many questions. Moreover, since the evidence of object evolution may appear diversely in unpredictable time, the relevant information can be collected as an Attribute Signal Table (AST) to record the significant importance of each attribute to each object in every time point in a local *KBS*. The AOT increment could be constructed using time series analysis technique to analyze the importance of each attribute to each object recorded in AST with the times to facilitate the acquisition and adaptation of dynamic knowledge without too many interactions with experts in a changing environment.

However, some new evolved objects might be invisible or insignificant under each local *KBS* with neo-learning module, the profile of each *KBS* and the infrequent logs are analyzed in the collaborative *KBS* to collaboratively assist experts in discovering new evolved objects. The infrequent inference logs can be analyzed by neo-learning module and corresponding profiles to discover the interesting knowledge of new evolved objects which is unseen in each *KBS*. In order to acquire a meaningful CF value of each new discovered embedded rule of evolved objects, the CF value of each new embedded rule of evolved objects could be adjusted in the collaborative *KBS* based upon three cases in the CF adjusting function.

A worm detection prototype system using the collaborative framework with the neo-learning module is implemented to evaluate the performance of dynamic *EMCUD*, which incrementally integrates the new evolved knowledge into original knowledge base. Based upon the collaborative framework, the dynamic knowledge of new worms could be elicited to discover the new variant worms generated by the attacking traffic generator in the experimental environment based upon the worm classification embedded rule base, which results in the ability of knowledge evolution.

2. Related work

Several knowledge acquisition (KA) methodologies and related systems are introduced in this section. Then Repertory Grid, one of the popular indirect KA techniques, is also discussed. Finally, the elicitation of embedded meaning and some problems of traditional KA methodologies are discussed.

2.1. Knowledge acquisition methodologies

Since the knowledge in many domains (the experience of domain experts) is continuously growing, many KA methodologies and tools have been proposed to help experts acquire the useful static knowledge with/without knowledge engineers and then to transfer these knowledge into knowledge base or other computerized representation forms. In general, there are three approaches for knowledge acquisition (Crowther & Hartnett, 1996; Hwang & Tseng, 1990; Mcgraw & Harbison-Briggs, 1989):

(1) *Interviewing experts by experienced knowledge engineers:* Knowledge engineers directly retrieve domain knowledge by interviewing with human experts, and transform the knowledge into the computerized format to help experts solve difficult problems in the real world. However, interviewing experts is usually time-consuming if the communication between domain experts and knowledge engineers is insufficient. It is also difficult for experts to be aware of the new evolved objects without any additional related information in the interviewing methodologies.
(2) *Machine learning:* The machine learning approaches can learn the useful static knowledge of well-known objects by collecting many useful cases and instances with/without the involvement of domain experts. However, the quality of the results usually relies on the selected training cases and lacks new cases of evolved objects in the training process.
(3) *Knowledge acquisition systems:* KA systems assist domain experts in generating useful static knowledge of well-known objects with/without the help of knowledge engineers. These systems or tools could reduce the effort of communication between knowledge engineers and domain experts and could reduce the risk and difficulty of selecting the suitable training cases.

In the past decades, many KA systems, e.g., *ETS* (Boose, 1984, 1985), *NeoETS* (Boose & Bradshaw, 1986), *AQUINS* (Boose & Bradshaw, 1987), *KITTEN* (Shaw & Gaines, 1987), *RuleCons* (Davis, 1987), *MOLE* (Eshelman, Ehret, McDermott, & Tan, 1987), *KSSO* (Gaines, 1987), *KRITON* (Diederich, Ruhmann, & May, 1987), *EMCUD* (Hwang & Tseng, 1990; Hwang & Tseng, 1991), *KADS* (Wielinga et al., 1992), *MCRDR* (Kang, 1996), *KAMET* (Cairo, 1998), *MedFrame/CADIAG-IV* (Boegl, 1997; Kolousek, 1997; Leitich et al., 2001), (Pan, Zheng, Zeng, & Hu, 2002) have been developed to rapidly build prototypes and improve the quality of the elicited static knowledge of well-known objects by domain experts. However, most of them cannot be used to construct the dynamic knowledge due to the limitation of the static attribute set of the static grid in a dynamic environment. For acquiring the dynamic knowledge of new evolved objects, the traditional KA approach needs to rerun to generate the dynamic knowledge, and thus cannot keep the useful information of each object during the period of evolution, resulting in the limitations of traditional Repertory Grid-based KA methodologies.

2.2. Repertory Grid methodology and relevant systems

Repertory Grid, based on Kelly's Personal Construct Theory (Kelly, 1955) which reports how people make sense of the world, could be used as an efficient KA technique in identifying different objects and distinguishing these objects in a domain. It is the basis of several computer assisted KA tools, such as *ETS* (Boose, 1984, 1985), *AQUINAS* (Boose & Bradshaw, 1987) and *KSSO* (Gaines, 1987).

A single grid represented as a matrix whose columns have element objects (labels) and whose rows have construct's attributes (labels) can classify a class of objects, or individuals. The value assigned to an element-construct pair need not be Boolean. Grid values have numeric ratings, probabilities, and other characteristics, where each value reflects the degree. Then, the expert is asked to fill the grid with 5-scale ratings, where "1" represents the most relevant attribute to the object; "2" represents that the attribute may relevant to the object; "3" represents "unknown" or "no relevance"; "4" represents that the object may have the opposite characteristic; "5" represents the most relevant opposite characteristic to the object. The whole concept of Repertory Grid technique can be described as following steps:

(1) Elicit all of the element objects, e.g., E_1, E_2, E_3, E_4, E_5 from the expert.
(2) Elicit the construct attributes (and their opposites), e.g., C_1, C_2, C_3, C_4 (C'_1, C'_2, C'_3, C'_4), from the expert. Each time three elements are chosen to ask for a construct to distinguish one element from the other two.

Table 1
The illustrative example of a Repertory Grid with ratings

Element/construct	E_1	E_2	E_3	E_4	E_5	
C_1	5	1	5	1	1	C'_1
C_2	4	4	4	1	4	C'_2
C_3	1	4	5	1	4	C'_3
C_4	1	4	4	5	5	C'_4

(3) Rate all of the [element, construct] entries of the grid with value range from 1 to 5. An illustrative example is given in Table 1.

As Repertory Grid technique has been widely used by researchers, some extensions have been made to enrich its representative ability for covering more knowledge, the value assigned to an element-construct pair may be Boolean, numeric ratings, probabilities, etc. For example, Dixit and Pindyck (1994), and Hwang (1995) extended the Repertory Grid technique to the fuzzy table, in which constructs were fuzzy attributes that could rate by means of fuzzy linguistic terms from a finite set. Castro-Schez, Jennings, Luo, and Shadbolt (2004) developed a technique using a fuzzy Repertory Grid for acquiring the finite set of attributes or variables that the expert uses in a classification problem to characterize and discriminate a set of elements. Moreover, several models have been proposed for handling uncertainties in expert systems through generating more meaningful rules from the Repertory Grid oriented approaches. *EMYCIN* certainty factor (CF) model was first used to decide the degree of the belief of a rule for uncertain reasoning (Shortliffe & Buchanan, 1975; Adams, 1985). Embedded Meaning Capturing and Uncertainty Deciding (*EMCUD*) KA system was proposed to extract rules with embedded meaning from hierarchical girds by defining the impacts of the constructs to each element (Hwang & Tseng, 1990) and was successive applied in a medical diagnostic system of acute exanthema in Taiwan (Hwang & Tseng, 1991). WebGrid, Calgary's web-based knowledge modeling and inference tool, is based on Repertory Grid elicitation and analysis (Shaw & Gaines, 1996).

Boicu (2001) described a practical approach, methodology and tool, for the development of static knowledge bases and agents by subject matter experts, with limited assistance from knowledge engineers, the idea of constructing the dynamic knowledge bases systematically is launched. However, the dynamic environment limits all their efficiencies for the KA methodology. The real world knowledge in a dynamic environment is often considered evolutional, because the knowledge can be changed or evolved with the times due to the advent of information century. The speed of knowledge changing is too fast to accumulate manually the information by experts which results in limiting the ability of the ritualistic batch process analysis.

Although these methodologies are proposed to extend the ability of uncertain reasoning to classify the well-known objects, they still have the limitation of acquiring static knowledge from the static grid. It is also difficult for experts to notice the occurrence of new evolved objects, which is evolved in the dynamic environment as time goes on. Therefore, an incremental KA system based upon Repertory Grid technique is hence proposed in this paper to integrate dynamic knowledge of new objects into original knowledge base through the observations of the interested inference logs.

2.3. Elicitation of embedded meanings

The embedded meanings referred here represents the information that domain experts take for granted but are implicit to the people who are not familiar with the application domain. The lack of embedded meaning will probably make an expert system fail to infer some cases being trivial to experts. *SEEK* (Politakis & Weiss, 1984) and *SEEK2* (Ginsberg, Weiss, & Politakis, 1988) have been proposed to obtain embedded meanings by some efficient refinement processes. However, the major problem of *SEEK* and *SEEK2* is the case database being assumed to be available because it is difficult to collect sufficient cases in some applications. Moreover, it would be also time-consuming and boring for experts to offer a conclusion for each case in the database before starting the refinement procedure.

EMCUD, a Repertory Grid-based KA method, is hence proposed to elicit the embedded meanings of knowledge (embedded rules bearing on m objects and k object attributes) from the existing hierarchical grids given by experts (Hwang & Tseng, 1990), which represents the information that domain experts take for granted but are implicit to whom are not familiar with the application domain. Additionally, it will also guide experts to decide the certainty degree of each rule with embedded meaning for extending the coverage of generated original rules. Besides using Acquisition Table (AT) to generate the rule of each object, the Attribute Ordering Table (AOT), which is used to record the relative importance of each attribute to each object, is employed to capture the embedded meanings of the resulting grids. The values in each AOT entry, a pair of attribute and object, may be labeled "X", "D" or an integer number. "X" means no relationship existing between the attribute and the object. "D" means that the attribute dominates the object, i.e., if the attribute is not equal to the entry value, it is impossible for the object to be implied. Integer numbers are used to represent for the relative important degree of the attribute to the object instead of dominating the corresponding object. If the attribute does not equal the attribute-value, it is still for the object to be implied. The larger integer number implies the attribute being more important to the object.

Using AOT, the original rules generate some rules with embedded meaning, and the CF value of each rule, which is between -1 and 1, could be determined to indicate the degree of supporting the inference result. The higher CF value implies the more reliable result than smaller CF value. The *EMCUD* algorithm is listed as follows.

```
Algorithm 1: EMCUD algorithm

Input: The hierarchical grids.
Output: The guiding rules with embedded meaning.
Step 1: Build the corresponding AOT with each grid of
the hierarchical multiple grids.
Step 2: Generate the possible rules with embedded
meaning.
```

Step 3: Select the accepted rules with embedded meaning through the interaction with experts.

Step 4: Generate automatically the CF of each rule with embedded meaning.

All rules generated by *EMCUD* can be categorized into two classes: original and embedded rules with acceptable CF value, and discarded rules with unacceptable CF value, according to the confidence degree of domain experts. Embedded rules can be generated by ignoring the minor (non-dominate) attributes recorded in AOT.

Each embedded rule is assigned a certainty sequence (CS), the sum of each AOT values of the ignored attributes, and the CF calculated by formula (1) which is between 0 and 1 can represent the degree of certainty for each embedded rule. Each of them is assigned a CF between 0 and 1 while the value approaches to 1 means more important; otherwise, the value approaches to 0 means less important:

$$\mathrm{CF}(R_{i,j}) = \mathrm{UB}(R_i) - \left[\frac{\mathrm{CS}(R_{i,j})}{\mathrm{MAX}(\mathrm{CS}_{i,j})} \times (\mathrm{UB}(R_i) - \mathrm{LB}(R_i))\right] \tag{1}$$

where $R_{i,j}$ and $\mathrm{CS}(R_{i,j})$ are the *j*th embedded rule of the object O_i and the CS value, respectively. The $\mathrm{MAX}(\mathrm{CS}_{i,j})$ is the maximum CS value of the embedded rules generated from object O_i.

To decide the CF of each embedded rule $R_{i,j}$, the upper ($\mathrm{UB}(R_i)$) and the lower ($\mathrm{LB}(R_i)$) bounds CF values of the object O_i have been firstly defined for accepted embedded rules. Then CF values of each rule can be automatically determine by the mapping function, formula (1). Thus, the useful embedded rules with corresponding CF values could be used to cover more uncertainty cases.

However, the Repertory Grid-oriented method to construct AT is somehow strenuous for an expert and even more strenuous to solve the adaptive problem in a dynamic environment; therefore, we propose an incremental KA method based upon *EMCUD* to cope with the problems above by further enhancing the original *EMCUD* method to become a dynamic *EMCUD* method, which can integrate dynamic knowledge into original knowledge base.

Since embedded rules with weak acceptable CF values (the CF values below a user defined threshold) usually mean domain experts might lack the strong confidence, objects matching weak embedded rules may be the candidates of new evolved objects. For example, the object satisfying the conditions (attribute-value pairs) of the embedded rules with CF = 0.5 means the expert might suggest that it would be marginally classified into the object class and the minor attributes of the embedded rule might be not clearly defined. Therefore, the fired frequencies of this kind of weak embedded rules could be used to discover the occurrence of new evolved objects.

With the changing environment, the adaptation of the acquired rules is required to cope with the new evolved objects. Although *EMCUD* successfully solves the problems of the conventional Repertory Grid including knowledge representation and embedded meaning for covering more similar objects in extended object class, it still exists several problems such as hard to explain the rules with lower CF value, difficulty in deciding the attribute ordering, and lacks the ability of grid evolution for singling these new objects out due to the knowledge explosion in a changing environment with the times. Therefore, enhancing the adaptation ability of embedded rules becomes increasingly important to achieve the ability of grid evolution in *KBS*.

3. The incremental knowledge acquisition methodology – dynamic *EMCUD*

Generating rules in *EMCUD* would be cost inefficient if the size of Acquisition Table (AT) and Attribute Ordering Table (AOT) are too large. After collecting sufficient information of new evolved objects, *EMCUD* has to manually regenerate the original and embedded rules to classify these new objects with the large main AT. Therefore, the concept of dynamic *EMCUD* shown in Fig. 1 is proposed to help experts incrementally generate the dynamic knowledge based upon a new evolved object learning (neo-learning) module to construct a small AT increment and an AOT increment for enhancing the explanation power of the original embedded knowledge base.

The AT increment, which can be generated by monitoring the frequency of the weak embedded rules, is used to record the new evolved objects and the attributes which are updated or added to generate the dynamic knowledge. The AOT increment is used to help experts generate the adaptive relative importance of each attribute to each object in AT increment as time goes on by tracing the importance changing trends of all attributes in a time interval in the trend evolution analysis. Through integrating the AT increment and the AOT increment into the main AT and the main AOT using grid merging in dynamic *EMCUD*, it can generate the knowledge of new evolved objects with the grid evolution ability.

3.1. The new evolved object knowledge acquisition

As we know, the *KBS* is proposed to help experts solve the difficult problems in a specific domain based upon the

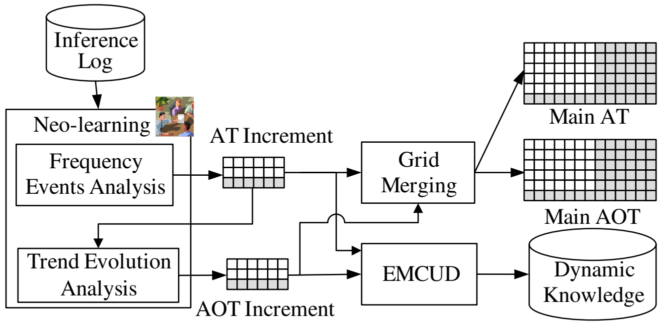

Fig. 1. The concept of dynamic *EMCUD*.

pre-constructed static knowledge base. However, the new objects will be developed or discovered as times goes on and might result in the inefficiency of *KBS*. Based upon the embedded rules generated by *EMCUD*, some new evolved objects may be classified into well-known object class by the weak embedded rule with weak CF which is not strongly suggested by experts. Through monitoring the frequency of these weak embedded rules, the candidates of new evolutional objects might be discovered to notice the experts. Therefore, the characteristics of these candidates of new objects could be extracted from these collected inference logs. The evidence of the new objects can be confirmed by experts and some attributes could be modified and added when the dynamic knowledge is needed to be singled out. Moreover, the relationships between these inference logs might be represented as the significance of each attribute to each new object. Hence, analyzing the evolving trends of all attribute should be useful in capturing the realistic significance of the attribute to the object.

The neo-learning module shown in Fig. 1 can help experts analyze the interesting inference logs of weak embedded rules to learn the evidence of new evolved objects using the frequent evolution analysis to notice experts the occurrence of the new objects. Based upon the confirmed new objects, the relationships of all attributes of each object are analyzed to set the significance of the attribute with the times using trend evolution analysis to help experts decide the CF values of the embedded rules of new objects, which can be generated using *EMCUD* according to the discovered objects stored in an AT increment and an AOT increment. Finally, the AT increment and the AOT increment will be integrated with the main AT and the main AOT, respectively.

3.2. Frequency events analysis

EMCUD lacks the ability of grid evolution for singling the new evolved objects out of well-known objects since experts may be unaware of the occurrence of the new evolved objects without sufficient information. Hence, we propose a frequency events analysis method to monitor the frequent behaviors of interesting inference logs of the weak embedded rules with the lower CF values for helping experts notice the occurrence of the new objects.

The novelty of the frequency events analysis shown in Fig. 2 is to collect the inference logs of weak embedded rules from each *KBS* to learn the candidates of new evolved objects for experts to make a confirmation. The minor attribute-value pairs between inference logs of weak embedded rules are useful to help experts discover new knowledge and determine whether new object is evolved based upon fired frequency. For each object, if its inference logs of weak embedded rules are frequent, the frequent minor attribute-value pairs could be treated as candidates of new evolved objects. Furthermore, new attributes or attribute-values of the new object could be defined and used to generate a small AT increment. Hence, these candidates will be used to help experts single the new objects out of the extended object class using the new object acquisition module based upon the AT increment.

Therefore, if the new objects are confirmed by experts, the related ambiguous attributes (minor attributes), which might result in the marginally acceptable CF values of weak embedded rules, could be refined or new attributes could be added to improve the classification ability. If the initial data type of a minor attribute is too rough to describe the object, a superior data type is recommended and the values of the attribute in both original object and new evolved object should be modified.

For example, the BOOLEAN data type may be refined to SINGLE VALUE data type (Hwang & Tseng, 1990). If changing the data type still cannot discriminate the new variants from original objects, acquiring new attributes from domain experts will be suggested in the new objects acquisition module. According to the complexity of relations between objects and attributes or even the relations between different tables, it is hard for experts to cooperate with each other in building every column and every row for each table. Finally, the result of new objects and corresponding attributes can be used to construct the AT increment.

3.3. Trend evolution analysis

Although the original idea of constructing AOT makes *EMCUD* more adaptive to elicit embedded meanings, the relative importance of all attributes to each object could be adjusted since the dynamic knowledge may change or evolve with the times. It means that some embedded rules, which are recommended by experts now, may become uncertain in the near future. Each object in the AOT is decomposed to record the relative importance of each attribute to the object with the times. Since the traditional Repertory Grid-based KA methods do not record the evolved trend of each new object and the *EMCUD* is difficult in deciding the ordering of all attributes of the object

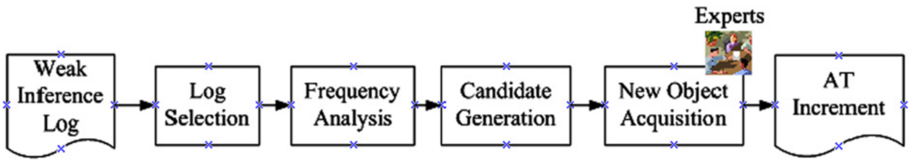

Fig. 2. The flow of frequency events analysis.

by experts, the trend evolution analysis, which can discover the evolution of the relative importance of each attribute to each object with the times, is proposed to help experts monitor the significant importance changing of all attributes to each object in a time interval.

As shown in Fig. 3, the object can be singled out of the old object according to the viewpoints of experts or the learning results of the frequency events analysis. Each attribute can be simply assigned as "0" or "1" in each time point for indicating whether it is important to each object or not, where "0" represents the attribute is considered as the unimportant attribute to the object and "1" represents the attribute is important to the object. The domain expert can then decide which attributes are required to be traced with the times if some ordering values of the attributes are hard to be decided immediately.

The "0" or "1" is called an attribute event e_t of each object in a time point t, and the attribute event sequence of "0" and "1" is recorded in a table called the Attribute Signal Table (AST) to capture the evolved behavior of each object. Hence, the AOT increment can be generated for evolving the relative importance of each attribute to each object (ordering values) according to the sequence of "0" and "1" events recorded in AST with the times using time series analysis. Since the "1" means an attribute is important to an object, the consecutive "1" recorded in consecutive time points indicates that relative importance of the object should become higher. On the contrary, the consecutive "0" indicates that the relative importance of the object should be lower. Hence, a simplified time series analysis is proposed to capture the trend meaning and incrementally adjust the CF value of each rule. Let the initial value of each signal sequence be the original AOT value of the attribute to the object.

3.3.1. Dynamic AOT adjusting function

Since the knowledge will be updated or evolved in a dynamic environment, the CF value of each embedded rule may be adjusted because the relative importance of the object may change. A Dynamic AOT Adjusting Function (2) is designed to generate the updated AOT value at time t by accumulating the collection of attribute event e_t at time point t based upon the previous AOT value at time $t-1$. If the attribute event e_t is assigned as 1 then γ is set to 1, which represents the increment is added into the AOT value at time $t-1$. Otherwise, γ is set to -1 if the e_t at time t is 0, which represents the decrement subtracted by the AOT value at time $t-1$. Hence, the ordering values can be refined with the times according to the collected information in a changing environment:

$$\mathrm{AOT}(t)=\mathrm{AOT}(t-1)+\gamma\times f(g(t)),\quad \begin{cases}\gamma=1, & \text{if } e_t=1\\ \gamma=-1, & \text{if } e_t=0\end{cases} \tag{2}$$

where $f(g(t))$, which is formally defined in formula (3), is used to decide the increment or the decrement of the corresponding the AOT value at each time point t, α, which is used to adjust the curvature of the AOT Delta Function, increases resulting in rapidly increasing or decreasing of the CF value, and β, which means the weight of the number of consecutive "1" or consecutive "0" received, decreases resulting in larger increment or decrement. In order to limit $f(g(t))$ between 0 and 1, the constant c is suggested to be smaller than -3:

$$f(g(t))=\begin{cases}0, & \text{if } e_t\neq e_{t-1}\\ \frac{1}{1+e^{\alpha\times(c+g(t)\times\beta)}}, & \text{if } e_t=e_{t-1}\end{cases} \tag{3}$$

where $g(t)$, the Continuous Events Accumulating Function given in formula (4), is used to record the number of consecutive "1" or consecutive "0" received at time t:

$$g(t)=\begin{cases}1, & \text{if } e_t\neq e_{t-1}\\ g(t-1)+1, & \text{if } e_t=e_{t-1}\end{cases} \tag{4}$$

3.4. Grid merging

In order to maintain the new discovered new object, we propose grid merging algorithm to integrate the AT increment and AOT increment into the main AT and the main AOT, respectively. Therefore, the small AT and the small AOT instead of the whole large main AT and the main AOT are used to update the embedded rule base.

Algorithm 2: The grid merging algorithm

Input: The main AT, main AOT, AT increment AT', and AOT increment AOT'.
Output: The updated main AT and main AOT
Step 1: Integrate the AT increment AT' into the main AT.

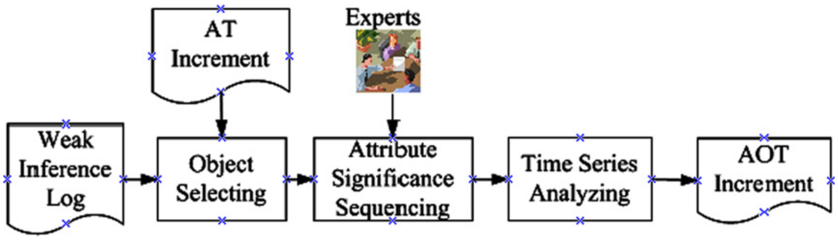

Fig. 3. Trend evolution analysis.

Step 1.1: Append each new object and each new attribute in the AT' as a new column and row in the main AT, respectively.

Step 1.2: Ask experts to fill the values of the modified attributes of other objects in the main AT if necessary.

Step 1.3: Ask experts to examine the values of the new attributes of other objects in the main AT if necessary.

Step 2: Integrate the AOT increment AOT' into the main AOT.

Step 2.1: Expand the size of the main AOT according to the main AT updated in the **Step1**.

Step 2.2: Fill the corresponding AOT values according to the AOT'.

Step 2.3: Refine the values of all old attributes to each old object in the main AOT using the *Trend Evolution Analysis* if necessary.

Step 3: Reset the AT increment AT' and the AOT increment AOT'.

To merge the AT increment into the main AT, each new evolved object should be appended as a new column in the main AT and each new added attribute should be appended as a new row in **Step 1.1**. In order to maintain the correctness of the main AT, the values of all modified or new added attributes to each object should be acquired by experts if necessary. Since the size of AOT need equal the size of AT, the size of the main AOT should be expanded in **Step 2.1** according to the main AT updated in **Step 1**. Besides the value of all attributes to each new object in AOT increment, the other values of all old attributes to each old object could also be learned using the trend evolution analysis to obtain the relative importance at time t.

4. The collaborative knowledge integration framework

Although the neo-learning module can be used to single the new objects out of extended object class and to generate their corresponding rules, some new evolved objects which may occur infrequently in each local *KBS* (but may be frequent in the collaborative *KBS*) cannot be found. Therefore, a collaborative framework is proposed to help experts collect the relevant information and discover these new evolved objects.

4.1. The concept of collaborative knowledge integration

In a dynamic environment, new object could be discovered using the collaborative framework shown in Fig. 4, which analyzes the related importance of the evolved objects.

The infrequent inference logs, profiles, and the discovered knowledge of new objects of each local *KBS* could be collected in our collaborative framework. As shown in Fig. 4, the neo-learning module is firstly used to discover some new evolved objects from the infrequent logs reported from all *KBS*s. Next, some new evolved objects may occur in some *KBS*s with similar profiles, e.g., the SQL server running on Windows operation system, the correlations between inference logs and profiles might be useful for helping experts discover them. Finally, for the discovered object, the CF value of the new embedded rule should be recalculated. Hence, a CF adjusting method is proposed to combine the knowledge of new objects discovered in each KBS and the collaborative *KBS*.

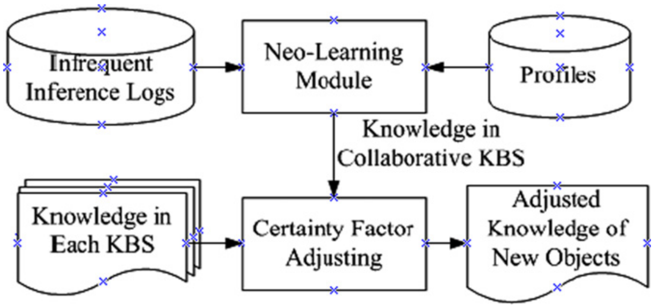
Infrequent Inference Logs
Neo-Learning Module
Profiles
Knowledge in Collaborative KBS
Knowledge in Each KBS
Certainty Factor Adjusting
Adjusted Knowledge of New Objects

Fig. 4. The concept of knowledge integration in the collaborative *KBS*.

4.2. Adjusting certainty factor of dynamic knowledge

Assume there are n local *KBS*s and each new evolved object may be discovered in p local *KBS*s, different CF values of a given embedded rule could be generated in each *KBS*.

(1) For $p > 0$, the CF Adjusting Function shown in formula (5) is proposed to help experts obtain the average of different CF value of a given embedded rule in each local *KBS* and adjust the scale of the CF increment or decrement (ΔCF) according to the discover of the new object in the collaborative *KBS*.

For each new embedded rule R_i, let the CF value be CF(R_i) and let the CF(R_i^j) be the CF value of each embedded rule R_i discovered in the jth local *KBS*.

$$\mathrm{CF}(R_i) = \frac{\sum_{j=1}^{n}\mathrm{CF}(R_i^j)}{p} + \delta \times \Delta\mathrm{CF} \tag{5}$$

Depending on whether the new objects are discovered in the collaborative *KBS* or not, the coefficient δ can be defined as follows:

Case 1: the new object can be discovered in the collaborative *KBS*.

δ is set to p/n.

Case 2: the new object cannot be discovered in the collaborative *KBS*.

δ is set to $(p - n)/n$.

(2) For $p = 0$, since the new object cannot be discovered in any local *KBS*, the new object could be discovered in the collaborative *KBS* according to the correlations of profiles. Therefore, the CF Adjusting Function could

be reduced to formula (6), where the $\mathrm{CF}(R_i^c)$ is the CF value of the new discovered rule in the collaborative *KBS* due to different configurations of profile.

$$\mathrm{CF}(R_i^c) = \mathrm{CF}(R_i^c) + \delta \times \Delta\mathrm{CF} \tag{6}$$

$\delta$is set to -2.

For example, the SQL Slammer uses UDP port 1434 to exploit a buffer overflow in a MS SQL server to simply switch off this port of the victim host. The collaborative *KBS* can learn this knowledge based upon the infrequent logs reported from some local *KBS*s according to the same service stored in profile. Some new objects may occur in similar environment.

5. Case study and experiments

Up to now, many antivirus products have been developed to discover worms, virus or Trojan horse in a computer system. However, these products are hard to automatically discover the variants of worms because the signature based approach fails when the signatures are changed. To overcome the weakness, we propose a worm detection prototype system, which has neo-learning module, to enhance the ability of commercial antivirus products by the collaborative framework.

5.1. Computer worm detection

Nimda, an incredibly sophisticated worm that made headlines worldwide, is taken as an example. Assume a simple Nimda concept tree is created in Fig. 5 after series of Nidma cases diagnosis, and it can be transformed into a worm Acquisition Table (AT) like Table 2. The following attributes are considered: the name of the e-mail attachment used by worms, the medium used by worms to upload, and the name of the file used by worms to start execution on servers. After constructing the worm AT, we construct the AOT table shown in Table 3.

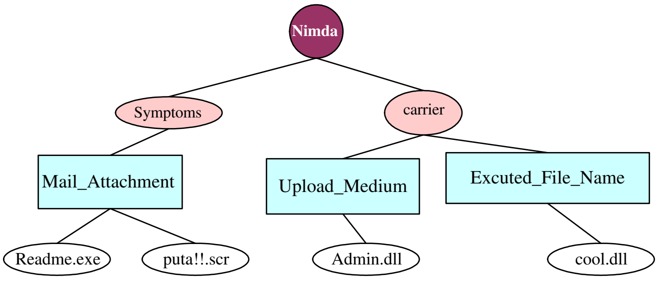

Fig. 5. Example of initial Nimda concept tree.

Table 2
An example of original Nimda AT

Attribute/object	Nimda
Mail_Attachment	Readme.exe
Upload_Medium	Admin.dll
Executed_File_Name	Riched20.dll

With both AT and AOT, the *EMCUD* method can be processed. The eight embedded rules are generated and some of them have low CF value such as rule R_1: "*IF Not Mail_Attachment* = *Readme.exe and Upload_Medium* = *Admin.dll and Executed_File_Name* = *Riched20.dll Then Nimda*" with CF value = 0.67. Therefore, suppose that in the inference process, the rule R_1 above is learned by neo-learning module almost all the time during a period, and suppose in the last two time points the embedded rule R_2: "*IF Not Mail_Attachment* = *Readme.exe and Not Upload_Medium* = *Admin.dll and Not Executed_File_Name* = *Riched20.dll Then Nimda*" with CF value = 0.4 is fired, the AST in Table 4 can be obtained.

Suppose that Nimda is the latest worm occurred in the world, its ordering value of each attribute cannot be easily determined because its variants may soon be broken out. The expert may define an AST with several time points, and then assign 0 in the first attribute, N_1, at first time point in Table 4. The attribute event N_2 at the second time point is set to zero. Next the time series analysis method first calculates the AOT according to the AST.

In Table 4, the Mail_Attachment attribute is calculated by time series analysis method, and the attribute is assigned a new ordering value = 1 since it is very possible to be changed again, subsequently, ordering value = 3 are assigned for both attributes Upload_Medium and Executed_File_Nam according to the AST. Therefore, the CF value of the rule R_1 is leveled up from 0.67 to 0.74. More-

Table 3
An example of original Nimda AOT

Attribute/object	Nimda
Mail_Attachment	2
Upload_Medium	3
Executed_File_Name	4

Table 4
An example of Nimda AST

Attribute/object	N_1	N_2	N_3	N_4	N_5	N_6	N_7
Mail_Attachment	0	0	0	0	0	0	0
Upload_Medium	0	1	1	1	0	0	0
Executed_File_Name	1	0	0	0	1	0	0

Table 5
An example of updated Nimda AT after discovering Nimda.B

Attribute/object	Nimda.A	Nimda.B
Mail_Attachment	Readme.exe	puta!!.scr
Upload_Medium	Admin.dll	Admin.dll
Executed_File_Name	Riched20.dll	Riched20.dll

Table 6
An example of integrated Nimda AT

Attribute/object	Nimda
Mail_Attachment	{Readme.exe; puta!!.scr}
Upload_Medium	Admin.dll
Executed_File_Name	Riched20.dll

Table 7
An example of updated Nimda AOT after discovering Nimda.B

Attribute/object	Nimda
Mail_Attachment	1
Upload_Medium	3
Executed_File_Name	3

over, several new attribute-values are learned by neo-learning module with Mail_Attachment = puta!!.scr in R_1, a new worm variant Nimda.B shown in Table 5 can be integrated into Table 6, and also an AOT is updated as shown in Table 7.

Therefore, with the accumulated inference logs from distributed sensors, the trend evolution analysis mechanism can also update the knowledge frequently. Assume neo-learning module learns another new attribute values including Mail_Attachment = sample.exe, Upload_Medium = cool.dll, and Executed_File_Name = httpodbc.dll in R_2 while the rule R_2 has always been fired in each time point in a short period, then a new variant Nimda.E is found. Finally, based upon the updated tables shown in Tables 8 and 9, the built system will give a whole picture of worms to guide the users who are not familiar in the domain for preventing or removing the malicious worms.

5.2. Worm detection prototyping system

As we know, many antivirus products (F-Secure, 2005; Symantec, 2006; Trend, 2006) have been proposed to discover worms, virus or Trojan horse in a computer system. However, these products are hard to automatically discover the variants of worms because the signature based approach fails when the signatures are changed (Moore, Shannon, Voelker, & Savage, 2003). To overcome the weakness, the worm detection prototype system is proposed to enhance the commercial antivirus products instead of replacing them. By only updating the knowledge base, the detection system can modify the defense mechanism for the variants of worm; as a result, the system can be easily maintained. Since the growth of the knowledge of worms is very fast, we propose a collaborative architecture for the adaptive worm detecting problem.

Table 8
An example of integrated Nimda AT after discovering Nimda.E

Attribute/object	Nimda
Mail_Attachment	{Readme.exe; puta!!.scr; sample.exe}
Upload_Medium	{Admin.dll; cool.dll}
Executed_File_Name	{Riched20.dll; httpodbc.dll}

Table 9
An example of updated Nimda AOT after discovering Nimda.E

Attribute/object	Nimda
Mail_Attachment	1
Upload_Medium	2
Executed_File_Name	2

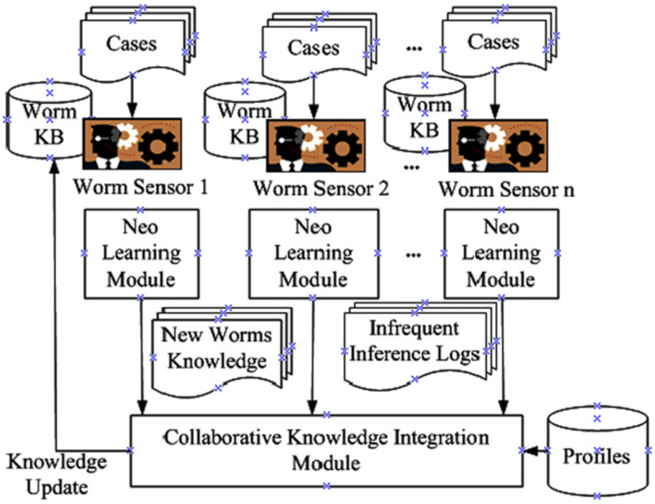

Fig. 6. The collaborative framework for worm detection.

Fig. 6 shows the collaborative framework for worm detection. In the architecture, the neo-learning module helps each worm sensor constructing AST to reconstruct AOT increment and update main acquisition table using AT increment (monitoring the frequent inference logs of weak embedded rules of worms with the times), where each sensor has its own Worm KB. By collecting the new worms knowledge and infrequent inference logs and consulting the Profiles, the collaborative framework can integrate the new worm knowledge.

Each worm sensor provides a web interface to collect or discover all the symptoms of worm cases by user and scanning tools. For example, when worm infects a victim system, the user can scan the host computer by some general antivirus software or can call for help from the Internet. The system collects all the information and infers the information based upon the worm knowledge with embedded meaning constructed by *EMCUD*. Consequently, the result of inferring will be passed to the users to teach the way of recovering the system. Moreover, the statuses which satisfy certain embedded rules will be considered to learn the new knowledge of new variant worms by neo-learning module.

5.3. Experimental results

We implemented a dynamic *EMCUD* web-based system, and used computer worm as an experimental domain,

Table 10
The ratio of discovering new evolved worm

	Original worms	New worms (%)	Inference costs (%)
Frequent based	100	60	193
Frequent based^{R}	100	100	294
Collaborative	100	100	200

where three computers were equipped with the dynamic *EMCUD* system and each one constructed its own knowledge base to evaluate the performance of discovering variants. We generated all kinds of test samples including the behaviors of original worms and variants (polymorphic worms) to randomly attack the network. The experimental result in Table 10 shows that the collaborative framework can successfully discover the variants by neo-learning module. Since some critical weak embedded rules may be ignored in the beginning of knowledge based construction, some specific variants which cannot be discovered by any individual neo-sensor can be detected by the rules.

The frequency based analysis module can detect only 60% variants. Obviously, when the knowledge is reconstructed, neo-learning module can learn all of the variants. However, it needs extra inference costs, the growth of the number of rules, about 100. On the other hand, our collaborative framework can learn all of the variants, it needs only seven extra inference costs to reach the goal in this example. Thus, our collaborative framework could be efficient in discovering new knowledge.

6. Conclusion

In this paper, the dynamic *EMCUD* based upon Repertory Grid is proposed to elicit the embedded meanings of knowledge. Dynamic *ENCUD* can generate an AT increment and an AOT increment to represent the evolved objects and to record the relative importance of each attribute to each object for capturing the embedded meanings with acceptable CF value by relaxing or ignoring some minor attributes. A collaborative knowledge acquisition framework is proposed to analyze the correlations of interesting inference logs of embedded rules between multiple local *KBS*s in a dynamic environment to discover the new evolved objects. Each *KBS* can monitor the frequent inference behaviors of weak embedded rules to construct an AT increment and analyze the significant change of the importance to evolved objects to construct an AOT increment for adjusting the relative importance of each attribute to each object with the times. Moreover, three cases are used to assist experts in adjusting the CF values of the discovered knowledge of the new evolved objects from the collection of inference logs. We implemented a worm detection prototype system to evaluate the performance of dynamic *EMCUD* to incrementally integrate evolved knowledge into knowledge base.

Acknowledgement

This work was partially supported by National Science Council of the Republic of China under Grant No. NSC95-2752-E-009-015-PAE.

References

Adams, J. (1985). Probabilistic and certainty factors. In B. Buchanan & E. Shortliffe (Eds.), *Rule-base expert systems: The MYCIN experiments of the stanford heuristic programming project*. Reading, MA: Addison-Wesley.

Boegl, K. (1997). *Design and implementation of a web-based knowledge acquisition toolkit for medical expert consultation systems*. Doctorial thesis, Technical University of Vienna, Austria.

Boicu, M. (2001). Automatic knowledge acquisition from subject matter experts. *IEEE International Conference on Tool with Artificial Intelligent*.

Boose, J. H. (1984). Personal construct theory and the transfer of human expertise. In *Proceedings of AAAI-84 conference* (pp. 27–33), California.

Boose, J. H. (1985). A knowledge acquisition program for expert systems based on personal construct psychology. *International Journal of Man–Machine Studies, 23*(5), 495–525.

Boose, J. H., & Bradshaw, J. M. (1986). NeoETS: Capturing expert system knowledge in hierarchical rating grids. In *IEEE Expert System in Government Symposium*.

Boose, J. H., & Bradshaw, J. M. (1987). Expertise transfer and complex problems: Using AQUINAS as a knowledge-acquisition workbench for knowledge-based systems. *International Journal of Man–Machine Studies, 26*(1), 3–28.

Cairo, O. (1998). KAMET: A comprehensive methodology for knowledge acquisition from multiple knowledge sources. *Expert Systems with Applications, 14*(1), 1–16.

Castro-Schez, J. J., Jennings, N. R., Luo, X. D., & Shadbolt, N. R. (2004). Acquiring domain knowledge for negotiating agents: A case of study. *International Journal of Human–Computer Studies, 61*(1), 3–31.

Crowther, P., & Hartnett, J. (1996). Using repertory grids for knowledge acquisition for spatial expert system. In *Proceedings of Australia and New Zealand Conference on Intelligent Information Systems* (pp. 14–17), Adelaide, SA, Australia, November 18–20.

Davis, R. (1987). Interactive transfer of expertise. *Artificial Intelligent, 56*, 121–157.

Diederich, J., Ruhmann, I., & May, M. (1987). KRITON: A knowledge-acquisition workbench for knowledge-based systems. *International Journal of Man Machine Studies, 26*, 3–27.

Dixit, A. K., & Pindyck, R. S. (1994). *Investment under uncertainty*. Princeton University Press.

Eshelman, L., Ehret, D., McDermott, J., & Tan, M. (1987). MOLE: A tenacious knowledge acquisition tool. *International Journal of Man Machine Studies, 26*, 41–54.

F-Secure Corporation (2005). http://www.f-secure.com/.

Gaines, B. R. (1987). An overview of knowledge-acquisition and transfer. *International Journal of Man–Machine Studies, 26*, 453–472.

Ginsberg, A., Weiss, S. M., & Politakis, P. (1988). Automatic knowledge base refinement for classification systems. *Artificial Intelligence, 35*(2), 197–226.

Hwang, G. J. (1995). Knowledge acquisition for fuzzy expert systems. *International Journal of Intelligent Systems, 10*, 541–560.

Hwang, G. J., & Tseng, S. S. (1990). EMCUD: A knowledge acquisition method which captures embedded meanings under uncertainty. *International Journal of Man–Machine Studies, 33*, 431–451.

Hwang, G. J., & Tseng, S. S. (1991). On building a medical diagnostic system of acute exanthema. *Journal of Chinese Institute of Engineers, 14*(2), 185–195.

Kang, B. (1996). *Multiple classification ripple down rules*. Ph.D Thesis, University of New South Wales.

Kelly, G. A. (1955). *The psychology of personal constructs*, Norton, New York.

Kolousek, G. (1997). *The system architecture of an integrated medical consultation system and its implementation based on fuzzy technology*. Doctoral thesis, Technical University of Vienna, Austria.

Leitich, H., Kiener, H. P., Kolarz, G., Schuh, C., Graninger, W., & Adlassnig, K. P. (2001). A prospective evaluation of the medical consultation system CADIAG-II/RHEUMA in a rheumatological outpatient clinic. *Methods of Information in Medicine, 40*, 213–220.

Mcgraw, K. L., & Harbison-Briggs, K. (1989). *Knowledge acquisition: Principles and guidelines*. Prentice-Hill International Editions, pp. 1–27.

Moore, D., Shannon, C., Voelker, G. M., & Savage, S. (2003). Internet quarantine: Requirements for containing self-propagating code. In *Proceedings of INFOCOM 2003*, March 30–April 3, San Francisco, USA.

Pan, D., Zheng, Q. L., Zeng, A., & Hu, J. S. (2002). A novel self-optimizing approach for knowledge acquisition. *IEEE Transactions on Systems, Man, and Cybernetics, Part A, 32*(4), 505–514.

Politakis, P., & Weiss, S. M. (1984). Using empirical analysis to refine expert system knowledge bases. *Artificial Intelligence, 22*, 673–680.

Shaw, M. L. G., & Gaines, B. R. (1987). KITTEN: Knowledge initiation and transfer tools for experts and novices. *International Journal of Man–Machine Studies, 27*, 251–280.

Shaw, M. L. G., & Gaines, B. R. (1996). Web grid: Knowledge modeling and inference through the world Wide Web. In *Proceedings of tenth knowledge acquisition workshop* (pp. 65-1–65-14).

Shortliffe, E. H., & Buchanan, B. G. (1975). A model of inexact reasoning in medicine. *Mathematical Bioscience, 23*, 351–379.

Symantec Corporation (2006). http://www.symantec.com/corporate/.

Trend Corporation (2006). http://www.trendmicro.com/tw/home/enterprise.htm.

Wielinga, B., Schreiber, A., & Breuker, J. (1992). KADS: A modeling approach to knowledge engineering. *Journal of Knowledge Acquisition, 4*(1), 5–53.