9

Proposing Two Algorithms to Acquire Learning Knowledge in Problem-Based Learning Environment

Akcell Chiang
*Department of Digital Media Design,
Tungnan University, Taipei,
Taiwan, R.O.C.*

1. Introduction

Discussion-based Internet forums or interactive chat rooms are an effective educational help system for both individuals and teams of learners. Many companies, particularly computer vendors (ComputerHope, 2004; IBM, 2004; Microsoft, 2004), adopt this technology to provide product training, learning or Q&A on their web sites. Nevertheless, the current state of chat forum technology only provides a platform for information exchange and organization (K.Kaye & J.Johnson, 2004). It is still unable to stimulate the users to learn from looking at the problem from different perspectives (Roberts & Ousey, 2003). The work reported in this chapter extends this technology further by incorporating critical thinking as a stimulus in teamwork discussions at algorithm one. We claim that such a method would sharpen web-based educational services. Separating the discussion topic from instant chat would further enhance the system into a comprehensive problem-based learning environment.

Problem-based learning (PBL) (Torp & Sage, 2002) and critical thinking (Fisher, 2001) skills have been used widely in education. These features are particularly noticeable in nursing education (Conway & Sharkey, 2002; Cooke & Moyle, 2002). This problem-based discipline can also apply to vocational education programs, such as, computer troubleshooting, which could be taught as a case study through Internet discussion. As we know, a trainee in a professional discipline, such as a computer technician, is more than just a passive learner but also an active problem-solver in real world situations. When a problem is encountered first in a learning process, it can be used as an initiative to build up learners' problem solving or reasoning skills. In the process, learners would appreciate this learning discussion and then embrace this added responsibility.

Essentially, there is a fundamental difference in the philosophy between nursing education and IT education (K.Kaye & J.Johnson, 2004; Lewis, Davies, Jenkins, & Tait, 2001). The difference in value between a life vs. a machine is such that discussion is not needed. A nursing-problem learning is very conservative in its approach; they often faced with poorly defined problems, incomplete information and etc. But a computer can be reloaded or reformated without any problems. It means we can redo the troubleshooting on a machine again without any major loss. Therefore one of our system design philosophies is to grant more control to the system, by using statistics for decision-making. Thus the learning system becomes more intelligent and reduces the workload of educator. On the other hand, we relegate some of the problem-based learning principles into the development of the on-line intelligent learning tool.

Internet forums allow users to post their topics/learning-issues for discussion. In a well facilitated web site, after having chosen a topic, users can post their suggestions or switch to chat room for on-line discussion. Our system is designed to accept a broader view of problems within the learning domain and deeper discussion. This means that we can allow different problem-descriptions to be posted from individual learners within the set up learning domain. We are not confined to single problem and may achieve superior understanding of the problem in the discussions.

This chapter introduces two algorithms/models to design the MALESA learning system – one is MALESAbrain for learning discussion and another is MALESAassessment for learning assessment. The former is a discussion system in facilitating a PBL classroom. It encourages learners to judge or criticize the solutions posted by others before exploring further knowledge-content. The system then sums up the judgment scores as its knowledge-weight in order to pass the thresholds set up for ranking/arranging the learning issues. The latter MALESAassessment is an assessment system for evaluating learners' performances after learning discussion. It uses a real world practical problem to assess the learners' solving-plan and fixing-processes as the educator's marking standard. These two systems co-operate to transform chat forum technologies for the problem-based learning in IT education.

2. MALESA learning system

The first system learning design has borrowed the threshold and knowledge-weight concepts from machine learning (Mitchell, 1997) to build up the intelligent learning tool. The educator needs only to give a learning domain and a few beginning questions (even beginning questions are not compulsory) to the system before the learning discussion starts. The system then asks learners to judge (or criticize) other's proposed solutions. Through threshold evaluation of the knowledge-weights, knowledge pieces are automatically ranked and arranged by the intelligent learning system. This kind of teamwork learning would then integrate and synthesize previous and current learning to increase the knowledge base.

2.1 The learning model of MALESAbrain[1]

The algorithm one is called MALESAbrain. The design of its learning model is to form a critical thinking methodology for problem-based discussion. Accordingly, we develop an intelligent system to help learners think critically and learn topics like computer troubleshooting through an Internet workshop. The model takes an active role in sharpening the learners' contributions towards viewpoints on the discussion issues. In discussion, the learning system would highlight the importance of those issues which help the learners pay more attention to consensus solutions for better discussion and problem solution. The model consists of three main stages (Fig. 1) to facilitate learners in problem-based discussion:

[1] The acronym for "Machine-Learning-Expert-System Algorithm for brainstorming"

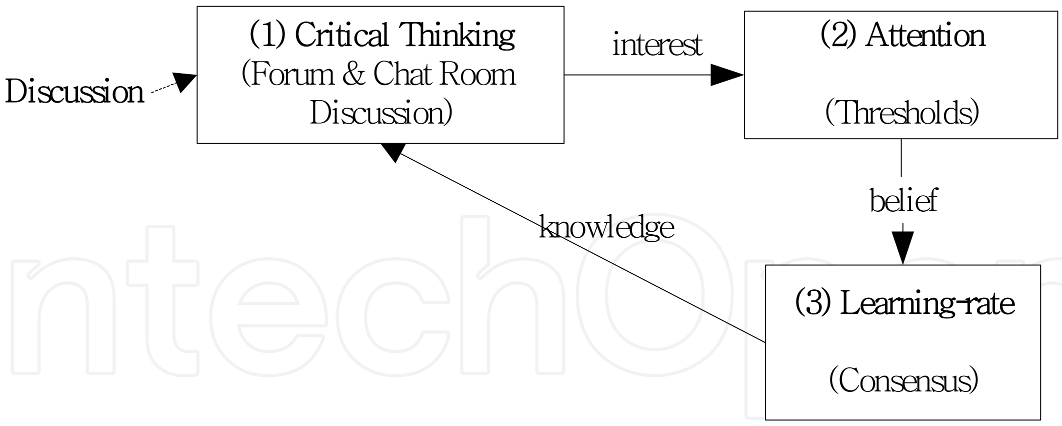


Fig. 1. The design model.

1. The first stage redesigns chat room and forum by adding the critical thinking function (Cooke & Moyle, 2002). This stage stimulates the learners to think about alternative aspects of the problem. They need to judge others' posted solutions by
   - giving personal preference or judgment on solutions posted by others;
   - contributing personal problem-solution suggestions for the feedback of preference from the judgments of others.
2. The second stage is to help learners to pay "attention" (S. Paul, Haseman, & Ramamurthy, 2004) to important issues.
   - Learners need to pay attention and think about why certain issues accumulate higher scores than others.
   - The highly-scored issues are highlighted by the system to stimulate more discussions on them.
   - Those extensively discussed issues therefore end up with more meaningful content to help solve the problems.
3. The third stage is to help learners and educator understand the current "learning-rate" (Culvenor, 2003). It indicates how many problems have consensus solutions, and how many problems are without consensus. The without consensus issues will become the learning issues that the learners still need to further investigate or research.
   - An indicator will show in the learning tool to help the educator and learners perceive what percentage of the discussion-problems has resulted in consensus;
   - also how many discussion-problems did not result in consensus.
   - How many discussion-problems with consensus results enable the educator and learners to understand the progress of the discussion and the learning issues on demand.

2.2 The assessment model of MALESAassessment[2]

The second algorithm is called MALESAassessment. The principle of the system design is to reuse learners' discussion knowledge, which retained in MALESAbrain learning system, for answering the follow-up performance test. In this chapter the assessment system consults the learners for organizing their own problem-solving plan for answering their test

[2] The acronym for "Machine-Learning-Expert-System Algorithm for learning assessment"

problems. It is not just to give a simple solution from the computer suggestion. The answer must be planned by learners and consulted by MALESAassessment.
Fig. 2 shows a rapid prototyping and user testing model, called "task-artifact cycle" (Finneran & Zhang, 2003; Vicente, 1999). MALESAassessment borrows the iterative improvement and concepts-in-action design (Sutcliffe & Carroll, 1999) to develop its assessment interface. It iteratively stimulates learners to improve their tasks (troubleshooting plans) and their actions (fixing problems). The four-stage continuous cycles, in MALESAassessment, help the learners to exercise a proto-type in ways that can contribute as much as possible. The open-ended design allows the learners to exercise their plans, rather than provide a raw solution which may be useless in a practical problem. The interface (see Fig. 13) offers advice to the learners to help them to build up a viable working plan to cope with the difficulty of solving a testing problem.

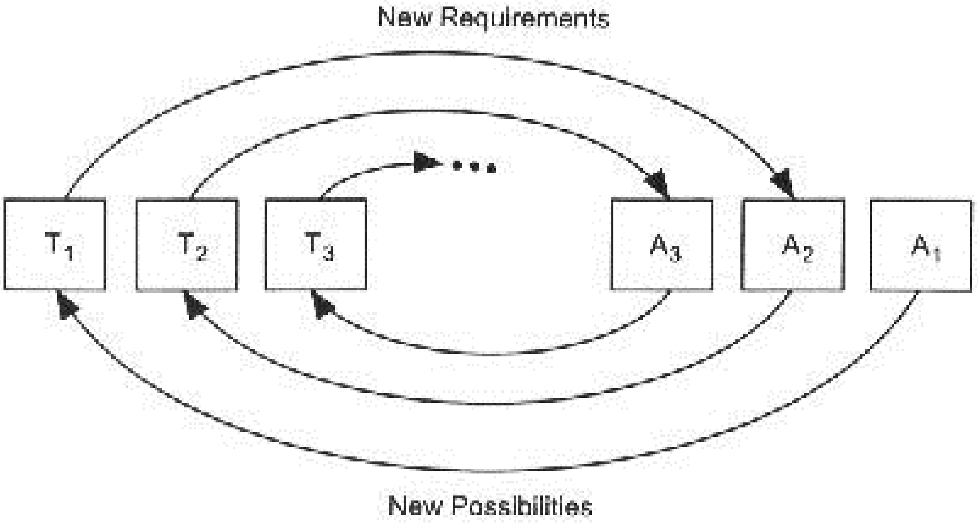


Fig. 2. Iterating through the task artifact cycle (reprinted from Fig 4.3 Vicente (1999) with permission).

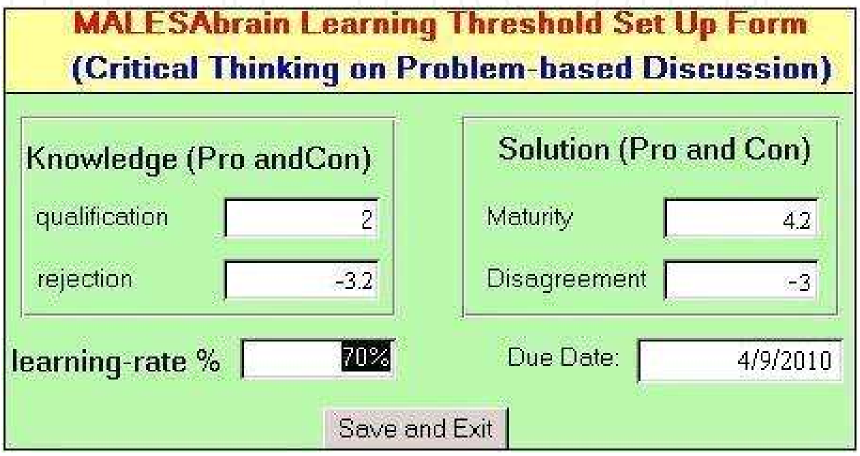
In the following sections, we further discuss these two algorithms. Each algorithm is separated into two sections. The first section applies our experiment result as example to picturize a whole concept to the readers. The second section offers the definitions, the algorithm and the knowledge retained in the system to clarify the previous example. After the algorithm discussions, we evaluate the system based on the participants' comments. At last, we conclude and highlight the contributions of these two algorithms.

3. Using MALESAbrain in problem-based discussion

To illuminate the learning algorithm in MALESAbrain, we use an example to explain the problem-based discussion guided by the learning system. In this section, we explain how an educator set up the learning thresholds, and then discuss the functions in the learning system.

3.1 Learning thresholds set up

Before discussion, the educator needs to set up learning thresholds to enable MALESAbrain to recognize the importance of the discussion-issues. In Fig. 3, the educator has set up "2" as *knowledge qualification-threshold*, when knowledge-weight ≥ 2 then the knowledge becomes a qualified knowledge, which is the minimum requirement to join the competition for promotion to a higher order of discussion position; "-3.2" is a *knowledge rejection-threshold*, when knowledge-weight < -*3.2* MALESAbrain would then delete the knowledge; "4.2" is a *solution-maturity threshold*, when the solution-weight ≥ 4.2, then MALESAbrain would consider this solution is able to solve a discussion-problem; "-3" is a *solution-disagreement threshold*, if solution-weight < -3 then MALESAbrain would delete the solution. However, to help MALESAbrain decide when to suggest learners to stop the discussion, the educator also needs to set up *learning-rate* "70%" and due-date "4/9/2010".



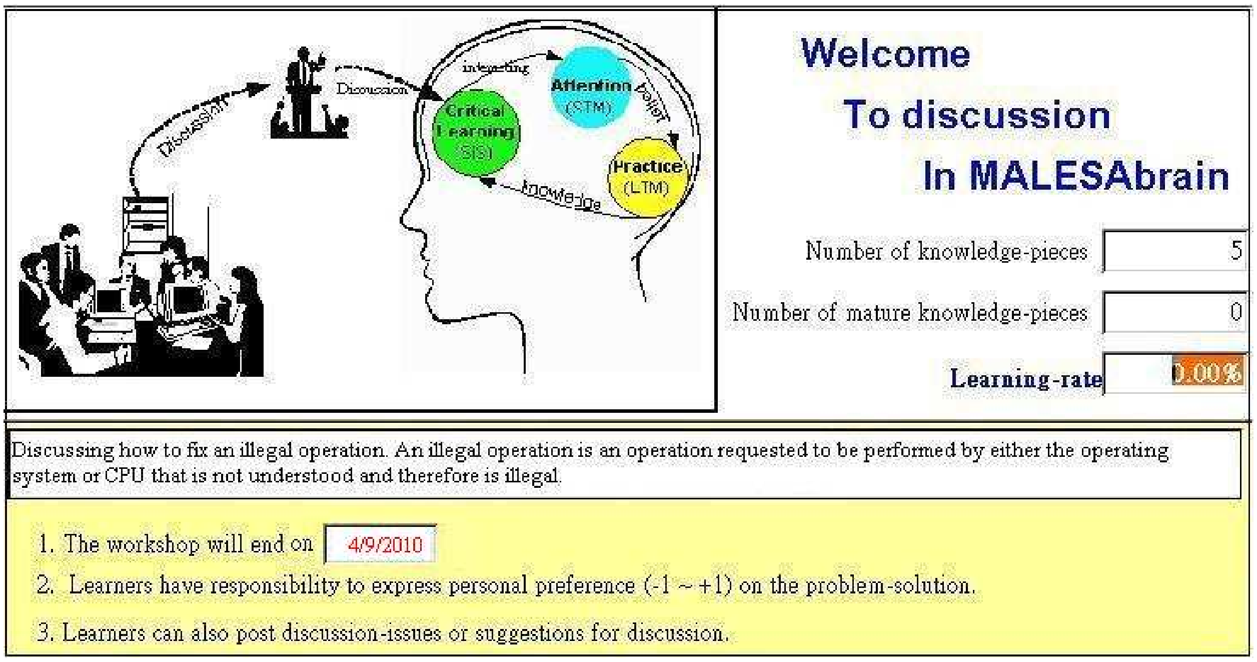
MALESAbrain Learning Threshold Set Up Form
(Critical Thinking on Problem-based Discussion)

Knowledge (Pro andCon)
qualification 2
rejection -3.2

Solution (Pro and Con)
Maturity 4.2
Disagreement -3

learning-rate % 70%
Due Date: 4/9/2010

Fig. 3. Educator set up learning thresholds.



Welcome
To discussion
In MALESAbrain

Number of knowledge-pieces 5
Number of mature knowledge-pieces 0
Learning-rate 0.00%

Discussing how to fix an illegal operation. An illegal operation is an operation requested to be performed by either the operating system or CPU that is not understood and therefore is illegal.

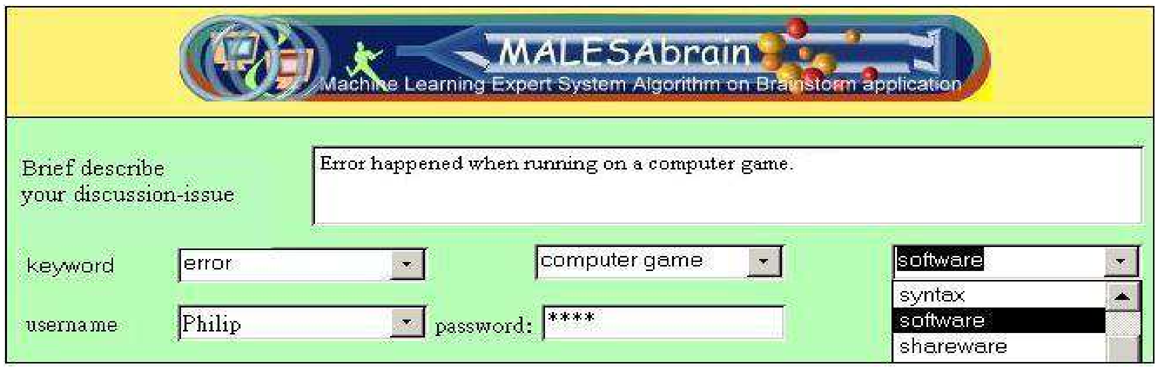
1. The workshop will end on 4/9/2010
2. Learners have responsibility to express personal preference (-1 ~ +1) on the problem-solution.
3. Learners can also post discussion-issues or suggestions for discussion.

Fig. 4. Welcome page encourages learners to follow the learning rules for discussion.

As shown in Fig. 4, MALESAbrain will keep assessing current learning-rate "0.00%" and checking the due-date "4/9/2010". During discussion, there are three figures shown on the welcome page (see Fig. 4), which help the educator to assess the current retained number of knowledge pieces "5", the current matured number of knowledge pieces "0" and the current learning rate “0.00%”. They help the educator and learners to understand the progress of the discussion. Nevertheless, at the due date “4/9/2010”, if the learning-rate is still lower than the setup learning-rate 70% then the educator will need to decide whether to extend the discussion. The educator may calibrate the learning factors by re-setting the *knowledge-qualification-threshold, knowledge-rejection-threshold, solution-maturity-threshold, solution-disagreemen- threshold* or *learning-rate* (see Fig.2) and arrange another discussion by extending the due-date. At the end of the learning discussion, the educator will stop the discussion and then assesses the achievement of total retained knowledge and individual learners’ contribution for which their suggestions have succeeded in the important area of the learning issues.

3.2 Problem-based learning discussion

After the educator sets up the learning thresholds, learners can start to discuss their problems. Fig. 5 shows a learner entering his/her problem descriptions by completing a ‘question enquiry form’. Once submitted, the query will be matched with MALESAbrain’s retained knowledge according to the chosen keywords. Firstly the query will be matched by the keywords. If none is found, it will be seconded to match the problem description details. As a result, the locations of the matched knowledge will be the output to which the learner is being advised to advance.


MALESAbrain
Machine Learning Expert System Algorithm on Brainstorm application

Brief describe your discussion-issue: Error happened when running on a computer game.

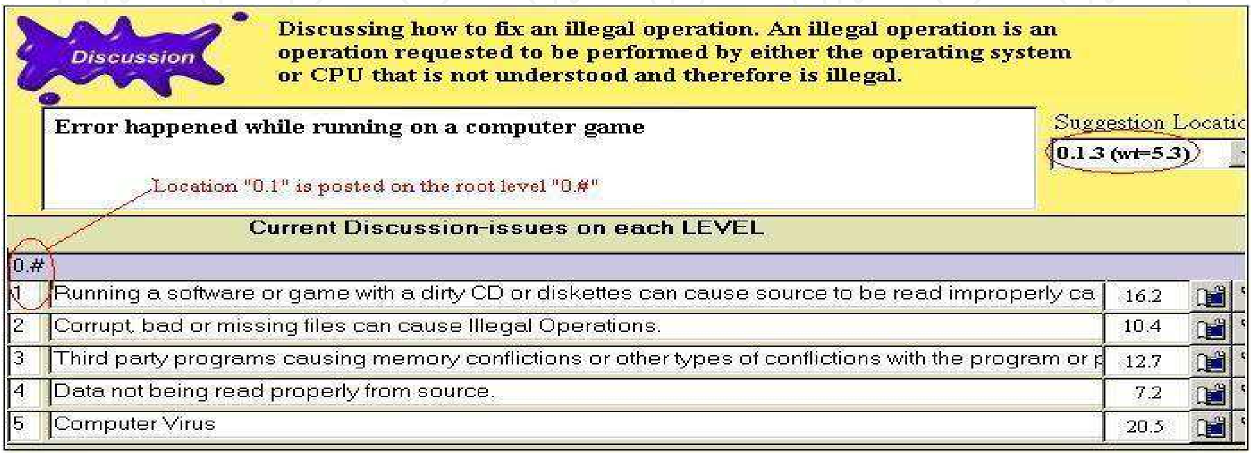
keyword: error | computer game | software

syntax
software
shareware

username: Philip password: ****

Fig. 5. A learner “Philip” enters his discussion-issue for joining MALESAbrain discussion.

In Fig. 6, the system suggests some locations for learners to join discussion. The suggestion location “0.1.3” weight “5.3” is on the lower/next level of location “0.1”. Learners can click



Discussing how to fix an illegal operation. An illegal operation is an operation requested to be performed by either the operating system or CPU that is not understood and therefore is illegal.

Error happened while running on a computer game

Suggestion Locatio
0.1.3 (wt=5.3)

Location "0.1" is posted on the root level "0.#"

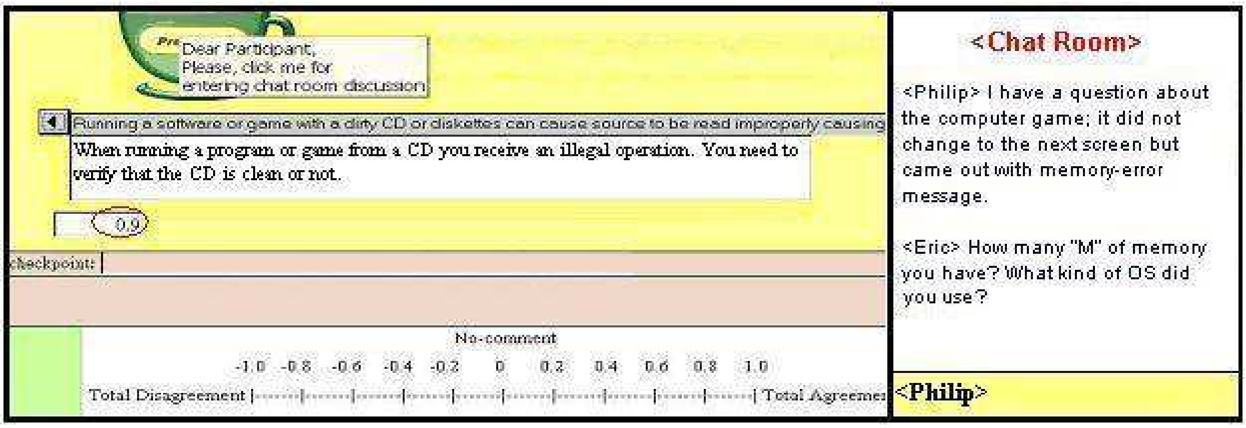
Current Discussion-issues on each LEVEL

0.#		
1	Running a software or game with a dirty CD or diskettes can cause source to be read improperly ca	16.2
2	Corrupt, bad or missing files can cause Illegal Operations.	10.4
3	Third party programs causing memory conflictions or other types of conflictions with the program or p	12.7
4	Data not being read properly from source.	7.2
5	Computer Virus	20.5

Fig. 6. After matching, the system suggests some locations for learners to join discussion.

location “0.1” to browse next level “0.1.#” which includes location “0.1.3” or directly click suggestion location “0.1.3”. Simply speaking, “0.1.3” is an example of a location address - about different level separated with “.” Like the “dot” in Internet address, “0.1.3” is an address in MALESAbrain: “0” is the root level address that learner much choose 1 to go to “0.1” and then choose 3 to go to “0.1.3”.

In Fig. 7, the system asks the learner about his/her preference. The learner then express his/her preference score “0.9” from his own judgment before being allowed to enter chat room for discussion. During exploration, MALESAbrain actively questions the learners about their *preferences* - a numerical measure of the learner’s degree of support (or lack of support) for a posted-solution to the problem. Learners should answer these questions prior to moving on to the next piece of content or chat room for discussion. The preference value ranges from 1 for total agreement, to 0 for no comment, and to -1 for total disagreement. Learners must judge or criticize another’s proposed solution, for its ability to solve a problem.



Dear Participant,
Please, click me for
entering chat room discussion

Running a software or game with a dirty CD or diskettes can cause source to be read improperly causing

When running a program or game from a CD you receive an illegal operation. You need to verify that the CD is clean or not.

0.9

checkpoint:

<Chat Room>

<Philip> I have a question about the computer game; it did not change to the next screen but came out with memory-error message.

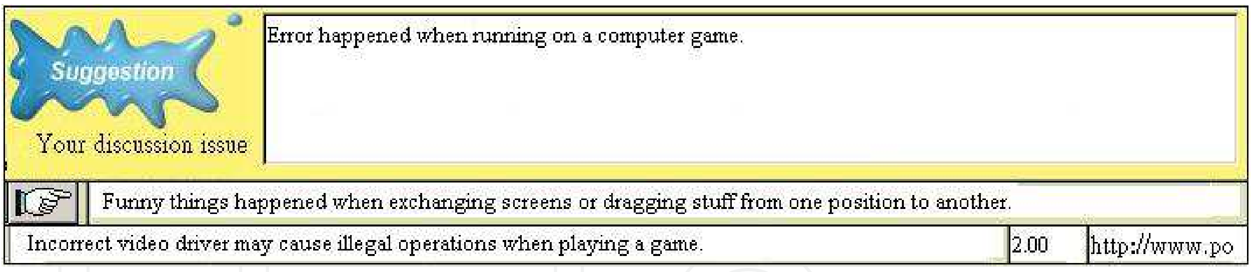
<Eric> How many "M" of memory you have? What kind of OS did you use?

<Philip>

Fig. 7. In response to the learner’s chosen issue, the system would actively ask its user about his/her preference.

Subsequently, learners can understand each issue better in the chat forum for their discussions. Such a device provides a window of opportunity for individual learners to review other problems from different perspectives and subjectively evaluate the advantages and disadvantages of other learners’ works. It is an important mechanism installed on the model, which encourages learners to critically think about a problem-solution from others’ suggestions and carefully judge their own preference scores. In certain situations, the educator will encourage each of the learners to pick up a learning issue for further investigation and research, such as, go to library, discuss with an expert, use Internet searching or test an experiment on the laboratory before going back for further discussion.

In Fig. 8, the user browses deeper into a few levels from a suggested location, and check a problem of interest, such as, “Funny things happened when exchanging screens or dragging stuff from one position to another” which attached a solution of “Incorrect video driver may cause illegal operations when playing a game”. By this kind of critical thinking and browsing others’ posted problem-solutions, learners would build up their knowledge regardless of the previous learned or current development.



Your discussion issue

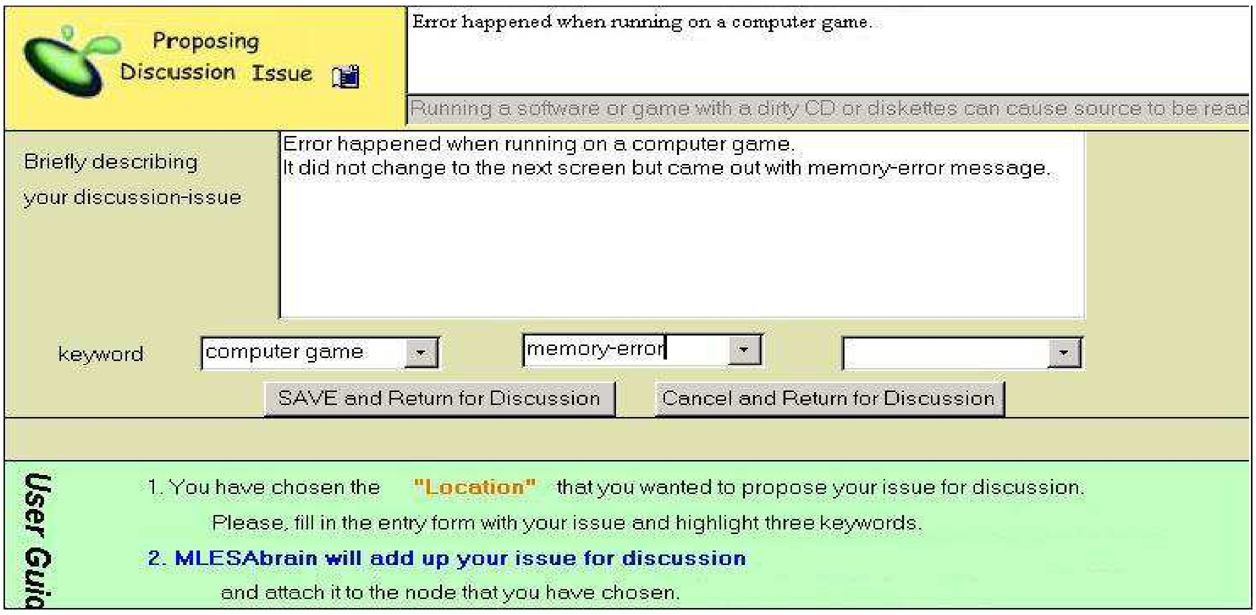
Error happened when running on a computer game.

Funny things happened when exchanging screens or dragging stuff from one position to another.

Incorrect video driver may cause illegal operations when playing a game. 2.00 http://www.po

Fig. 8. shows a learner browsing deeper into a few levels from a suggested location.

However, if there are no suitable learning issues for their discussions, learners might post their own problems. Fig. 9 shows a learner posting his/her own problem, about “Error happened when running on a computer game. It did not change to the next screen but came out with memory-error message”.


Proposing Discussion Issue

Error happened when running on a computer game.

Running a software or game with a dirty CD or diskettes can cause source to be read

Briefly describing your discussion-issue

Error happened when running on a computer game.
It did not change to the next screen but came out with memory-error message.

keyword: computer game | memory-error |

SAVE and Return for Discussion
Cancel and Return for Discussion
User Guid

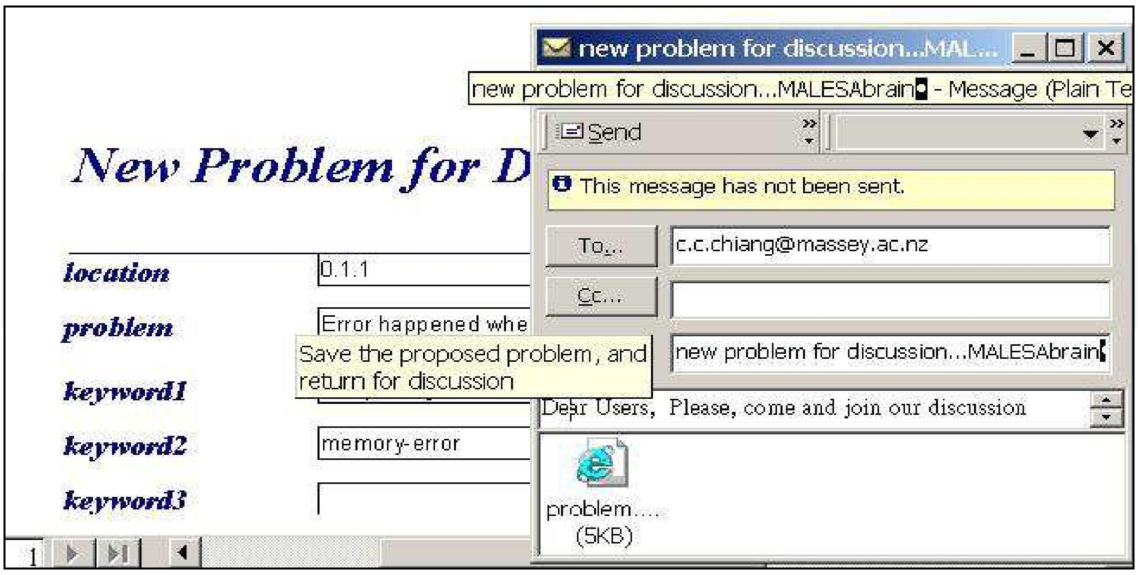
1. You have chosen the "Location" that you wanted to propose your issue for discussion.
Please, fill in the entry form with your issue and highlight three keywords.

2. MLESAbrain will add up your issue for discussion
and attach it to the node that you have chosen.

Fig. 9. A learner posts his/her own problem for joining discussion.

Fig. 10 shows MALESAbrain sending the user’s posted problem to other learners. This is a *broadcasting function* that gives the posted problems a chance to be discussed from other learners’ viewpoints or judgments. In this process of a problem-solution discussion, the learners who do not agree with a problem might propose another problem to clarify the original problem; and those who do not agree with a solution might contribute another solution to clarify or specify the original solution. However, when any new knowledge is added, MALESAbrain would notify other learners to encourage them to join the discussion.

This broadcasting function can also be considered as a feedback mechanism, which stimulates the learners to brainstorm more knowledge among participants. In Fig. 11, the system adds the problem to the location “0.1.4”. The user now needs to wait for the feedback from the other learners’ judgments on his/her proposal. These broadcasts and discussions will continue until MALESAbrain can identify the knowledge according to the set up thresholds. The more discussions which are returned the more changes occur in individual knowledge-weights because of the contribution from the learners’ preferences.



new problem for discussion...MAL...

new problem for discussion...MALESAbrain - Message (Plain Te

This message has not been sent.

To... c.c.chiang@massey.ac.nz

Cc...

new problem for discussion...MALESAbrain

Dear Users, Please, come and join our discussion

problem.... (5KB)

New Problem for D

location 0.1.1

problem Error happened whe

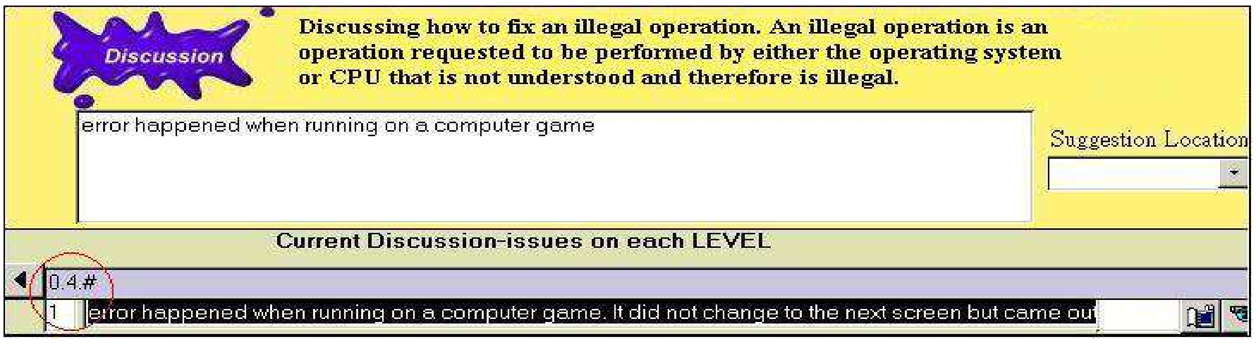
Save the proposed problem, and return for discussion

keyword1

keyword2 memory-error

keyword3

Fig. 10. MALESAbrain sends the user's posted problem to other learners.



Discussing how to fix an illegal operation. An illegal operation is an operation requested to be performed by either the operating system or CPU that is not understood and therefore is illegal.

error happened when running on a computer game

Suggestion Location

Current Discussion-issues on each LEVEL

0.4.#

1 error happened when running on a computer game. It did not change to the next screen but came out

Fig. 11. The system adds the problem to the location "0.1.4".

4. The algorithm of MALESAbrain

In this section, we explain the first algorithm – MALESAbrain. The definitions precisely define the symbols used in the algorithm. The algorithm shows how an educator set up the learning thresholds; and how MALESAbrain facilitates an on-line discussion for knowledge acquisition. The calculation example discusses the knowledge-weights of the retained knowledge.

4.1 Definitions of MALESAbrain

Definition 1-1. A piece of *knowledge-content* φ_i in MALESAbrain for problem-based discussion is defined as a pair of problem and solutions:
$\varphi_i = (p_i, \bigcup_j s_{i,j})$ where p_i is a *problem* $\bigcup_j s_{i,j}$ is the collection of suggested solutions associated with p_i

Definition 1-2. The knowledge *preferences* ***Pref*** in MALESAbrain is defined as a continuous function of real value ranged from -1 to +1

$$\textbf{\textit{Pref}}\text{:}(\, learner_k\,,\, s_{i,j}\,) \rightarrow agreement_{k,i,j} \ (\text{-}1 \sim +1)$$

where ($learner_k$, $s_{i,j}$) is a pair such that

$learner_k$ = a learner,

$s_{i,j}$ = a solution in MALESAbrain (see the defined solution $s_{i,j}$ in Definition 1-1).

$agreement_{k,i,j}$ = the preference score of an learner's, $learner_k$, judgment of a solution $s_{i,j}$ (value from –1 ~ +1).

Definition 1-3 The *knowledge-weight* w_i in MALESAbrain is defined as

$w_i = \sum_{j=1}^{m} w_{i,j} \cdot |\, s_{i,j}\,|$ where $w_{i,j}$ is the summation of all learner $learner_k$ preferences towards $s_{i,j}$

$w_{i,j} = \sum_{k} \ agreement_{k,i,j}$ (see Definition 1-2 ***Pref***: < $learner_k$, $s_{i,j}$ > → $agreement_{k,i,j}$)

Note: we define a symbol " | | " here, which will be used to test the existence of a solution, for transfer the existence of a solution into the value "0" or "1", to allow the knowledge-weight w_i calculation. $|\,x\,| = \begin{cases} 0 & when\ (\neg\exists x) \\ 1 & when\ (\exists x) \end{cases}$ where χ is a solution. (see Example 1: the calculation of knowledge weight)

Definition 1-4. An *artificial-knowledge-CELL* (AK-cell) k_i in MALESAbrain is a combination definition of Definition 1-1 and Definition 1-3, which is defined as a pair of knowledge-content and knowledge-weight:

$k_i = <\varphi_i, w_i>$ where

φ_i is the knowledge-content (see Definition 1-1 $\varphi_i = (p_i\,,\ \bigcup_j\ s_{i,j})$) and

w_i is the corresponding knowledge-weight (see Definition 1-3 $w_i = \sum_{j=1}^{m} w_{i,j} \cdot |\, s_{i,j}\,|$)

Definition 1-5. The *learning threshold* θ is defined as a collection of two decision pairs $\theta = \{<\theta_{kq}, \theta_{kr}>, <\theta_{sm}, \theta_{sd}>\}$, for comparing the retained AK-cells and their respective solutions, where

θ_{kq} is an *AK-cell qualification threshold*, when $w_i \geq \theta_{kq}$ then k_i becomes a qualified AK-cell, which is the minimum requirement to join the competition for promotion to a higher order of discussion position

θ_{kr} is an *AK-cell rejection threshold*, when $w_i < \theta_{kr}$ then delete the AK-cell k_i

θ_{sm} is a *solution maturity threshold*, when $w_{i,j} \geq \theta_{sm}$, the learning group agrees the solution $s_{i,j}$ is able to solve the problem p_i .

θ_{sd} is a *solution disagreement threshold*, if $w_{i,j} < \theta_{sd}$ then delete the solution $s_{i,j}$

Whenever any of the thresholds are triggered by an AK-cell or a solution, MALESAbrain will re-organize the knowledge structure.

Definition 1-6. The *growth-factor* γ, an integer number, in MALESAbrain is defined as the limit for constraining the posted number of AK-cells at each level, which converts the architecture of knowledge base from linear structure into a hierarchical structure, of γ-branch tree, in the forum.

Definition 1-7. The *learning-rate function* $\frac{|M|}{|K|} \geq \alpha$ is defined to help the educator and learners to understand the progress of the discussion, where the *convergent-factor* "α" is the educator's training target set up and

the learning-rate $\frac{|M|}{|K|}$ is a percentage of the discussion-problems resulted in consensus, where

$|\boldsymbol{M}|$ is the number of mature AK-cells, where $\boldsymbol{M} = \{k_i \mid k_i \in \text{MALESAbrain} \wedge \exists s_{i,j} \text{ where } w_{i,j} \geq \theta_{sm}\}$

$|\boldsymbol{K}|$ is the number of current retained AK-cells , where $\boldsymbol{K} = \{k_i \in \text{MALESAbrain}\}$

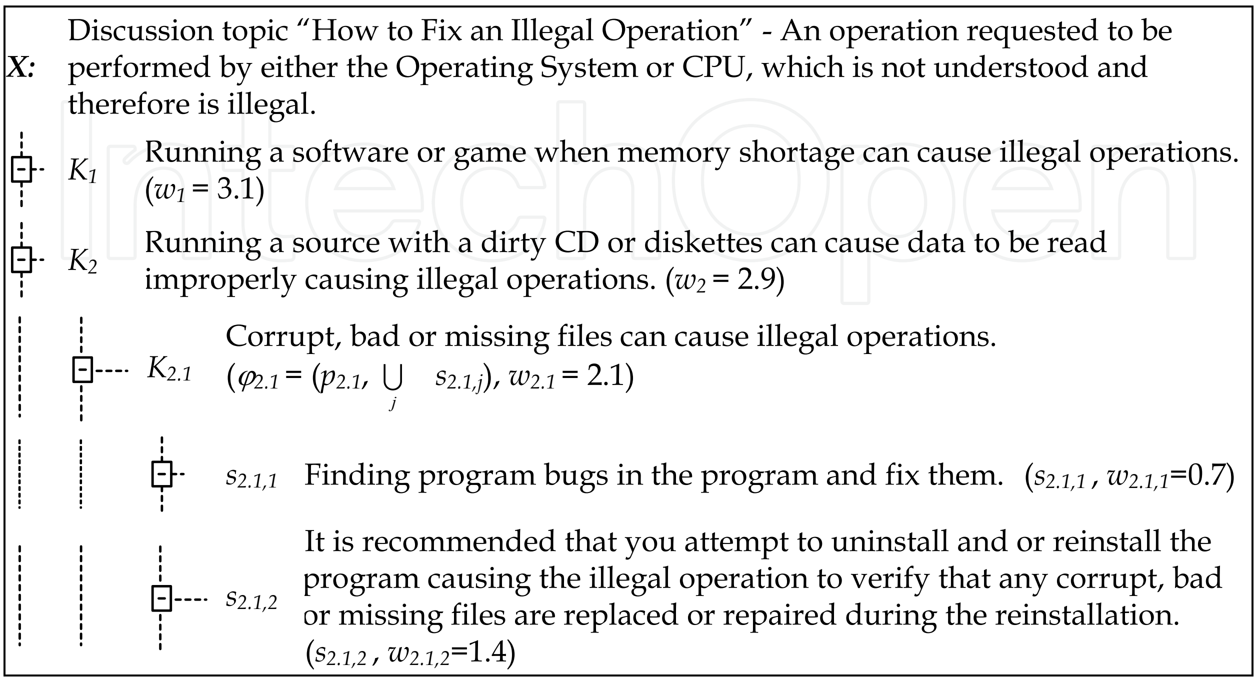
4.2 Algorithm of MALESAbrain

To trace the algorithm, we suggest readers browsing the definitions first. Then reading the instructions, comments in the algorithm and checking definitions when tracing. Afterwards compare the algorithm with the example in previous section.

```
MALESAbrain($\theta_{kq}$,$\theta_{kr}$ ,$\theta_{sm}$,$\theta_{sd}$,$\gamma$,$\alpha$,due-date )
        SET $\theta_{kq}$ / * the minimum requirement of an AK-cell to join the competition for promotion to
        a higher discussion position - called qualification threshold (Definition 1-5)*/

        SET $\theta_{kr}$, / * the worthless AK-cells will be deleted whenever lower than the rejection threshold
        (Definition 1-5) */

        SET $\theta_{sm}$ / * a consensus of the learning group agrees the solution is able to solve the
        problem whenever a solution reached the solution maturity threshold (Definition 1-5)*/

        SET $\theta_{sd}$ / * the worthless solutions will be deleted whenever lower than the solution
        disagreement threshold (Definition 1-5) */

        SET $\gamma$ / * growth-factor limits the posted number of AK-cells at each level, which converts
        the architecture of knowledge base from linear structure into a hierarchical structure of $\gamma$-
        branch tree in the forum. (Definition 1-6) */

        SET $\alpha$ / * convergent-factor is used to decide whether the percentage of mature AK-cells has
        achieved the educator's training target (Definition 1-7) */

        SET due-date / * due-date is an expected date to end the discussion */

    REPEAT
        /* Critical thinking */
        COMPARE personal viewpoint "x" with retained knowledge pieces "$k_i$" in MALESAbrain $K = \{k_i \in$
        MALESAbrain$\}$ /* Definition 1-4 */
        Learners INPUT their Pref:( $learner_k$ , $s_{i,j}$ ) → $agreement_{k,i,j}$ (-1 ~ +1) /* learners' show their
        judgment score to others' problem-solutions while browse the forum (Definition 1-2) */

        IF (Pref between –1 ~ +1 but not nil) THEN allow ENTER chat-room or MOVE to next pieces
        of knowledge  /* Learners must give their preference scores prior to moving on to chat
        room or next pieces of knowledge for discussion  (Definition 1-2) */

        MALESAbrain COMPUTE the knowledge-weight $w_i = \sum_{j=1}^{m} w_{i,j} \cdot | s_{i,j} |$ /* Definition 1-3 */
        MALESAbrain DISPLAY (or POST) learners' discussion issues based on "growth-factor $\gamma$"  /*
        Definition 1-6 */

        /* Attention */
        IF $w_i \geq \theta_{kq}$ THEN  /* the AK-cell is a qualified AK-cell  (Definition 1-5,7)*/
            INCREMENT a "qualified-mark" AK-cell $k_i$ to knowledge base $K = \{k_i \in$
            MALESAbrain}
            IF $w_i$ > parent($w_i$ ) THEN  /* higher weight AK-cell should be in the front level */
                SWAP($k_i$ ,parent($k_i$ ) )
            END IF
        END IF

        IF $w_i < \theta_{kr}$ THEN  /* the AK-cell is a worthless knowledge-piece (Definition 1-5)*/
            DELETE $k_i$
        END IF

        IF $w_{i,j} \geq \theta_{sm}$ THEN  /* the AK-cell becomes an matured AK-cell in the knowledge base
        (Definition 1-5,7)*/
            DISPLAY a "mature-mark" on solution $s_{i,j}$
            INCREMENT a "mature-mark" AK-cell $k_i$  to $M = \{k_i \mid k_i \in$ MALESAbrain $\wedge \exists s_{i,j}$
            where $w_{i,j} \geq \theta_{sm}\}$
        END IF

        IF $w_{i,j} < \theta_{sd}$ THEN  /* the solution is worthless solution  (Definition 1-5)*/
            DELETE $s_{i,j}$
        END IF

        IF (dd:mm:yy = due-date) THEN
            IF ( $\frac{|M|}{|K|} \geq \alpha$) THEN  /* if learning-rate $\frac{|M|}{|K|}$ is greater or equal to convergent-factor
            "$\alpha$" then end discussion (Definition 1-7) */
                PRINT (MALESAbrain meeting with $\frac{|M|}{|K|}$ % learning-rate)
                endDiscussion = TRUE  /* stop the meeting */
            ELSE
                endDiscussion = FALSE
                CALL MALESAbrain($\theta_{kq}$,$\theta_{kr}$ ,$\theta_{sm}$,$\theta_{sd}$,$\gamma$,$\alpha$,due-date )   /* re-calibrate the learning
                thresholds and start another session of discussion whenever the learning-rate
                lower than $\alpha$ when time is due */
            END IF
        END IF

    UNTIL (endDiscussion = TRUE)  /* learning-rate */
END MALESAbrain
```

4.3 A calculation example

In this subsection, we use a knowledge retained snapshot and a calculation example to look into the internal storage structure of the knowledge base of MALESAbrain.



X: Discussion topic "How to Fix an Illegal Operation" - An operation requested to be performed by either the Operating System or CPU, which is not understood and therefore is illegal.

K_1 Running a software or game when memory shortage can cause illegal operations. (w_1 = 3.1)

K_2 Running a source with a dirty CD or diskettes can cause data to be read improperly causing illegal operations. (w_2 = 2.9)

$K_{2.1}$ Corrupt, bad or missing files can cause illegal operations. ($\varphi_{2.1} = (p_{2.1}, \bigcup_j s_{2.1,j})$, $w_{2.1}$ = 2.1)

$s_{2.1,1}$ Finding program bugs in the program and fix them. ($s_{2.1,1}$, $w_{2.1,1}$=0.7)

$s_{2.1,2}$ It is recommended that you attempt to uninstall and or reinstall the program causing the illegal operation to verify that any corrupt, bad or missing files are replaced or repaired during the reinstallation. ($s_{2.1,2}$, $w_{2.1,2}$=1.4)

Fig. 12. A snapshot of the retained knowledge pieces in MALESAbrain's knowledge base.

Fig. 12 shows a snapshot of knowledge pieces retained in the knowledge base. The knowledge posted by the learners has been organized in a tree-like manner according to the respective weights of individual nodes. In this example shown, the AK-cell $K_i = K_{2.1}$ includes a *knowledge-content* $\varphi_i = \varphi_{2.1}$ and a *knowledge-weight* $w_i = w_{2.1}$. For illustration the calculation on the knowledge-weight $w_i = w_{2.1}$ bonds with solution $s_{i,j} = s_{2.1,1}$ and solution $s_{i,j} = s_{2.1,2}$. Let us assume three visited learners have given their preferences on solution $s_{2.1,1}$ as 0.9, 0.5 and -0.7; and two visitors have given their preferences on solution $s_{2.1,2}$ as 0.6 and 0.8.

By Definition 1-3 $w_{i,j}$ is the learners' judgment scores towards $s_{i,j}$

$\because$ $s_{2.1,1}$ has three visited learners' judgment scores (0.9, 0.5 and -0.7)

$\therefore w_{2.1,1}$ = 0.9 + 0.5 + (-0.7) = 0.7…①

$\because$ $s_{2.1,2}$ has two visited learners' judgment scores (0.6 and 0.8)

$\therefore w_{2.1,2}$ = 0.6 + 0.8= 1.4…②

$\varphi_{2.1}$ includes the problem $p_{2.1}$ and two solution $s_{2.1,1}$ and $s_{2.1,2}$ (because the number of solutions is two, therefore m = 2), so the knowledge-weight is:

$$w_{2.1} = \sum_{j=1}^{m} w_{2.1,j} \cdot |s_{2.1,j}| = \sum_{j=1}^{2} w_{2.1,j} \cdot |s_{2.1,j}|$$

$$= w_{2.1,1} \cdot |s_{2.1,1}| + w_{2.1,2} \cdot |s_{2.1,2}|$$

$$= 0.7 \cdot |s_{2.1,1}| + (1.4) \cdot |s_{2.1,2}| \quad (\because \text{①} \text{ and } \text{②})$$

$$= 0.7 \cdot (1) + (1.4) \cdot (1) = 2.1 \ (\because |x| = \begin{cases} 0 & when\ (\neg\exists x) \\ 1 & when\ (\exists x) \end{cases} \text{ see Definition 1-3})$$

Example 1. The calculation of knowledge weight ($w_{2.1}$).

This value of 2.1 represents the weight ($w_{2.1}$) of the knowledge-content ($\varphi_{2.1}$) in the knowledge (AK-cell $K_{2.1}$), which provides a quantitative measure of the synergic viewpoint on $\varphi_{2.1}$ obtained from the discussion and it forms the basis for MALESAbrain's knowledge judgment capability. In the example the qualification threshold θ_{kq} is set to "2" (see Fig. 3). Then any AK-cell weights higher than 2 point will be qualified to join the competition for promotion (see Definition 1-5). This means that the moment the knowledge-weights "3.1" of K_1 and "2.9" K_2 are greater than $K_{2.1}$ "2.1" (see Fig. 12); otherwise the system will swap the positions of the lower-weighted AK-cell with the higher-weighted AK-cell. Furthermore, the *AK-cell rejection threshold* θ_{kr} is set up as "-3.2" and the knowledge-weight $w_{2.1}$=2.1, therefore $K_{2.1}$ will not be deleted. The *solution maturity threshold* θ_{sm} is set up as "4.2" and the solution-weight $w_{i,j}$ = $w_{2.1,2}$ = 1.4, therefore the learning system does not agree the solution $s_{2.1,2}$ is able to solve the problem $p_{2.1}$. The *solution disagreement threshold* θ_{sd} is set up as "-3" and the solution-weight $w_{2.1,1}$ = 0.7, therefore the solution $s_{2.1,1}$ will not be deleted by the system.

There is one definition worth to memtion here, the growth-factor, in definition 1-6, is introduced to convert the linear structure into a hierarchical structure of discussion-issues posted on the forum. The hierarchical structure helps our learners decide on different important positions, the learning-issues posted on the forum must not appear as a linear structure on the separate pages on the web site, but as hierarchical on demand. The growth factor normally is set up before discussion; however, it can also be changed after discussion if the educator wants to view the forum from a different angle. If the growth-factor has been set to 3 then the decision tree will be turned into three AK-cells on the top level and become a three-branch tree; if it has been switched to 5 then the decision tree will be turned into five best AK-cells on the top level for the learning decision and become a five-branch tree; if set up as one AK-cell then there is only one best decision to be made and become a linear-tree.

5. Using MALESAassessment in a test

The second algorithm – MALESAassessment - is to test students' performance after MALESAbrain's learning discussion. In this section, we use an example to explain how learners apply MALESAassessment for answering the testing problem.

After completing learning discussion, the educator will re-assign suitable keywords to each AK-cell according to his/her own understanding and viewpoint. This operation causes knowledge base reconstruction and turns it into a CBR expert system - called MALESAassessment. According to the matching of the reassigned-keywords, similar cases would be retrieved whenever users enter their inquiry. The similar cases retrieved are the AK-cells with attached knowledge-weight as the users' reference.

After set up CBR expert system, the educator will explain important cases to the learners with MALESAassessment. In the discussion the educator gives learners an understanding of what kind of cases that s/he considers good discussion-issues. It then gives a demonstration of how to use MALESAassessment to answer the questions in the assessment test.

After the demonstration, the educator will give learners a practical test to assess their understanding when handling a testing problem. For example, the educator can take off some memory from a motherboard and then run a heavy memory consuming game in a computer laboratory. S/he would ask the learners to make their own troubleshooting-plans and then to fix the problem. To make a troubleshooting plan the learners can seek advice from MALESAassessment. They can get the consultation cases from the original discussions, this helps them understand and cope with the testing problem. Their answers (troubleshooting plan) and thinking processes (log file) will also be recorded in the system to help educator in marking.

Fig. 13 shows the assessment interfaces in MALESAassessment. The CBR expert system consults the learners when making their troubleshooting-plans for answering the test. There are four screens or interfaces for consulting the learners to arrange their troubleshooting plans. First, the "*Inquiry*" screen can enter clients' query problem. Secondly, the "*consultation*" screen will offer the advice. Thirdly, the "*attention*" screen can help clients pay attention to their chosen cases/AK-cells. And the last/forth "*plan*" screen will help the learners make their problem-solving plan.

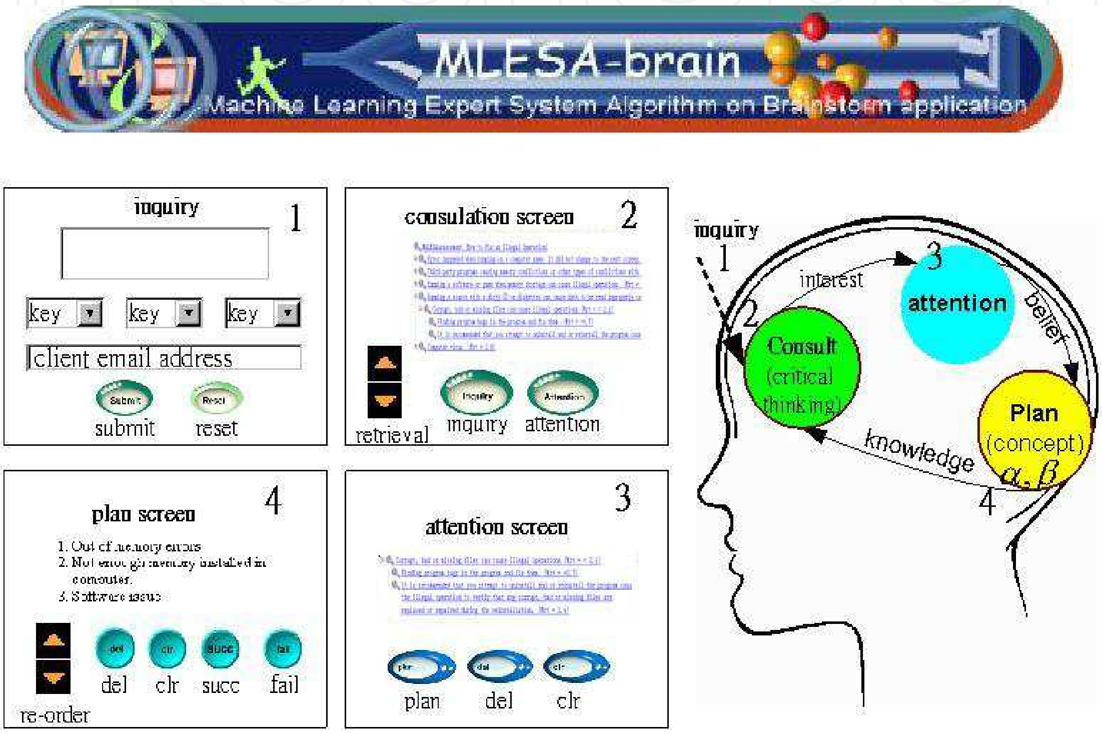


Fig. 13. The interfaces of assessment model in MALESAassessment.

5.1 Inquiry screen

The inquiry screen (1) shows that the learner enters his/her problem in order to seek advice. The learner types his/her question on the inquiry-board, choose one to three keywords, and input his/her contact email address if s/he wants the feedback from the educator after assessment; then submits or resets his/her request. In the mean time, behind the screen, MALESAassessment will retrieve the keyword-matched AK-cell from its CBR knowledge base; and output them to the consultation screen (2).

5.2 Consultation screen

When the learner changes his/her screen to consultation screen (2), the system will provide some preliminary suggestions to the learner according to the request on the inquiry screen (1). The retrieved cases are sorted according to their similarity and knowledge-weight. The learner still has to revise them to construct his/her own troubleshooting plan as the testing request. S/he will be encouraged to choose his/her interested AK-cells from screen (2); and output them into attention screen (3) for reuse and study. Whenever s/he clicks an AK-cell, it will change the AK-cell as a target for CBR and retrieve those cases of interest to the learner. These kinds of learner-browsing and system-retrieving operations will be repeated until s/he can find interested AK-cell on the screen.

Fig 13 shows there are two control buttons and two adjustment arrows on the consultation screen (2). To show his/her interest in an AK-cell, s/he can click "attention" button to put the chosen AK-cell to the attention screen (3). To re-enter his/her inquiring problem, s/he can click "inquiry" button for restarting a query. To change the number of AK-cells on the screen, the learner can also click either the "up" or "down" arrow to increase or decrease the number of the AK-cells on the screen.

This process of selecting interested AK-cells will be continued until the learner is satisfied with the selection of interested AK-cells on the attention screen (3).

5.3 Attention screen

After successfully collecting a list of interested AK-cells, the learner then exits consultation screen (2) and pays attention to the interested knowledge pieces on attention screen (3) for study and revision. This screen shows a list of interested AK-cells from learner's choices. Each AK-cell also includes a list of suggested solutions, obtained from the learning discussions. To answer the assessment test, learner now needs to pay attention to these solution-lists to discover suitable solutions to fix the real world problem.

There are three buttons on this screen. First, the learner can choose a solution then click a "plan" button, which will append the selected solution to the plan screen (4). Secondly, s/he can choose an unwanted AK-cell then click "delete" button for deleting the uninteresting AK-cell at the screen. Thirdly, if the learner cannot make a plan on the screen, s/he might click "clear" button to clear the whole screen and returns to consultation screen (2) for restarting and choosing more practical AK-cells.

5.4 Plan screen

After obtaining some solutions from the attention screen (3), the learner now revises and re-arranges the order of executive steps on the plan screen (4). On this screen, there are two arrows for "ascending" or "descending" the position of a selected solution, which helps the learner to organize his/her plan. There are also four other buttons on the plan screen with different usages. The "deleting" button is for deleting a selected solution. The "clearing" button is for clearing the whole screen and switching back to the attention screen (3) for restarting the selections again. The "successful" exit button is for the learner who has successfully made his/her troubleshooting plan and fixed the assessment problem. The "fail" exit button is for the learner who cannot establish a plan or cannot fix the assessment problem.

Behind the screen there is one thing worth noting. No matter whether the learner clicks "successful" exit or "fail" exit, the assessment reports will be sent to the educator for marking. The assessment reports include two parts:

i. the troubleshooting plan, and
ii. the log file which records whole learner's planning tasks with time stamps.

The educator then marks the assessment reports based on these two documents. When it is a successful case, the educator can concentrate on his/her troubleshooting plan for marking. When it is a failure case, the educator can concentrate on the log file and give him/her some proper advice for improving. Lastly, the marked reports would be retained in another CBR knowledge base as a refined second version expert system.

6. The algorithm of MALESAassessment

This section explains the second algorithm - MALESAassessment. It applies six definitions to clarify the symbols used in the instructions. The algorithm shows how the educator re-assigns suitable keywords to each AK-cell to reconstruct the system to a CBR expert system; and how learners apply MALESAassessment to answer the testing question. At last, we provide a successful retained case in the system during the experiment.

6.1 The definitions of MALESAassessment

Definition 2-1. The *re-organize(**K**)* action is used to reorganize the knowledge base, which includes two steps, are defined as follow:

1. keywords(educator, k_i), means the educator would re-assign the keywords to each AK-cell k_i in the knowledge base $\boldsymbol{K}$
2. CBR($\boldsymbol{K}$), means MALESAbrain $\boldsymbol{K}$ has been re-constructed as CBR expert system - called MALESAassessment.

Definition 2-2. The *inquiry(x)* function is defined as

$$Inquiry{:}x \xrightarrow[matching]{} \bigcup_{i=0}^{n} k_i \quad \text{where}$$

- x is participant's inquiring-input which includes inquiring-question, keywords and participant-email
- $\bigcup\limits_{i=0}^{n} k_i$ is retrieved AK-cells as CBR output to the consultation screen.

Definition 2-3-1. The *retrievalQuantity(σ±ε)* is defined as the number of AK-cells being retrieved into consultation screen where

$$\left| \bigcup_{i=0}^{n} k_i \right| \le \sigma \pm \varepsilon \text{ where}$$

- is the default retrieval number of AK-cells
- ±ε is an offset to adapt the retrieval quantity

Definition 2-3-2. A *consultation-action(c_i)* is defined as an action chosen by the participants on consultation screen, where

$$c_i = \begin{cases} inquiry(x)\ ;\ or \\ attention(k_i) \end{cases} \qquad \text{where,}$$

- *inquiry(x)* , means participant do not satisfy with the inquiry result on the consultation screen and want to re-enter another inquiry again.
- *attention(k_i)*, means participant chosen an AK-cell for reuse, and MALESAassessment then append it to the attention screen.

Definition 2-4. A *attention-action(a_i)* is defined as

$$a_i = \begin{cases} plan(s_{i,j})\ ;\ or \\ delete(k_i)\ ;\ or \\ clearAttentionScreen \end{cases} \qquad \text{where}$$

- *plan($s_{i,j}$)*, means participant chosen a solution on the attention screen for revision, and MALESAassessment then append it to the plan screen.
- *delete (k_i)*, means participant want to delete a chosen AK-cell on the attention screen.
- *clearAttentionScreen*, means participant does not satisfy with the whole attention screen and want to clear them.

Definition 2-5. A *plan-action(p_i)* is defined as

$$p_i = \begin{cases} position(s_{i,j})\ ;\ or \\ delete(s_{i,j})\ ;\ or \\ clearPlanScreen\ ;\ or \\ exit(successful)\ ;\ or \\ exit(fail) \end{cases} \qquad \text{where}$$

- *position($s_{i,j}$)*, means participant want to organize solution's execution steps; by either click ascending or descending on a chosen solution to the right execution step.
- *delete($s_{i,j}$)* , means participant want to delete a chosen solution on the plan screen.
- *clearPlanScreen*, means participant does not satisfy with the whole plan screen and want to clear it.
- *exit(successful)*, means participant have successfully made the plan and fixed the problem. MALESAassessment then retains the log file and the successful plan into a new CBR knowledge base as a successful case when s/he exits.
- *exit(fail)*, means participant cannot establish a plan or cannot fix the problem. MALESAassessment then retains the log file and the failure plan into a new CBR knowledge base as a failure case when s/he exits.

6.2 The algorithm of MALESAassessment

```
re-organise(K)  /* Preparation  Definition 2-1 */
        INPUT keywords(educator, $k_i$)
        CONSTRCT CBR($K$)

END re-organise

MALESAassessment(participant$_i$,$K$)
    CASE screen chosen by participant$_i$  OF  /* the participants can choose one of the four
    screens */
```

```
        Inquiry screen  (1): GOTO inquiry(x)    /* Inquiry screen definition 2-2*/
        Consultation screen (2): GOTO consultation-action($c_i$)  /* Consultation screen
        definition 2-3*/
        Attention screen (3): GOTO attention-action($a_i$)  /* Attention screen definition 2-4*/
        Plan screen (4):    GOTO plan-action($p_i$)  /* Plan screen definition 2-5*/
    ENDCASE
END MALESAassessment

inquiry(x)  /* Inquiry  Definition 2-2 */
        INPUT inquiring-question
        INPUT keywords
        INPUT participant-email
END inquiry

consultation-action($c_i$)  /* Consultation Definition 2-3*/
    CASE operation chosen by participant$_i$  OF  /* the participants can choose one of the
    four operations */
        "inquiry":          GOTO inquiry($x$)  /*Definition 2-2*/
        "attention":        DISPLAY-IN attention($k_i$)  /*Definition 2-3-2 */
        "up":               INCREMENT retrievalQuantity($\sigma + \varepsilon$)   /*Definition 2-3-1 */
        "down":             DECREMENT retrievalQuantity($\sigma - \varepsilon$)   /*Definition 2-3-1 */
    ENDCASE
END consultation-action

attention-action($a_i$)  /* Attention  Definition 2-4*/
    CASE operation chosen by participant$_i$  OF  /* the participants can choose one of the
    three operations */
        "plan" :            DISPLAY-IN plan($s_{i,j}$)
        "delete":           DELETE delete($k_i$)
        "clear":            CLEAR clearAttentionScreen
    ENDCASE
END attention-action

plan-action($p_i$)  /* Plan  Definition 2-5*/
    CASE operation chosen by participant$_i$  OF  /* the participants can choose one of the
    five operations */
        "position":         DETERMINE position($s_{i,j}$),
        "delete":           DELETE delete ($s_{i,j}$)
        "clear":            CLEAR clearPlanScreen
        "successful":       EXIT exit(successful)
        "fail":             EXIT exit(fail)
    ENDCASE
END plan-action
```

7. Discussion

IT education is in a state of change. Similar to nursing education, IT courses have a high content load to teach in the state of the art. However, the large student numbers and limited staff in the computer class often hinder interested educators combining PBL with structured teaching in their curricula. The question is "without an effective and efficient tool, it seems impractical to ask an educator handle with large number of group discussions in the discipline of PBL".

In this experiment, we test whether these two algorithms in the learning system can support PBL discussion in IT curricula or not? We want to know:

- Does the system help students organize and synthesize knowledge in discussion?
- Does the system save educator the efforts in helping discussion group effectively?
- Does the system efficiently save educator the efforts on the student performance test?

The first algorithm, MALESAbrain has been tested in our graduate laboratory, with "70%" learning rate setup and for two-week time of discussion. This experiment invited six of the postgraduate students to discuss the question: "How to fix an illegal operation? An illegal operation is an operation requested to be performed by either the Operating System or CPU that is not understood and therefore is illegal".

The evaluation after the experiment, with "56%" learning rate on the due date; and MALESAbrain has received some comments from the participants:

In considering of the methodology, the learning system

1. helps the students to think about the pros and cons of the proposed issues before they go into chat room for discussion.
2. highlights the issues with different levels of importance to help the students to identify the significance of the learning issues in the problem. .

In considering of the educators' benefits, the learning system

1. allows educator to monitor learner-groups' discussions on his/her screen to save the shortage of manpower and time limitation.
2. allows the educator to coach the meeting progressing and to guide learners in the right direction for discussion.
3. allows educator to encourage each of the students to pick up a learning issue for further investigation and research.

However, the strategies to set up the "learning thresholds" in MALESAbrain still remain to be evaluated and estimated. There are two reasons:

1. Firstly, it is because we are trying to change the learning thresholds setup by observing the number of learners in discussion at that time. In the Internet discussion, sometimes, it is not easy to control the number of participants who are really keen to learn and join discussion. Based on observing the number of participants and the discussion situation, we changed the learning thresholds to control the remaining learning issues to a number.
2. Second, we do not use "rejection threshold" to prune out worthless issues during discussion until due date for calculating the leaning-rate. It is because we do not want to lose any proposed issues. General speaking, it looks fine in this experiment. However, we are not sure it is still a proper procedure on the other experiment?

In the experiment, the thresholds' setup proves to be challenged by the participants – "why not let them decide the thresholds before the meeting, because it is their discussions and is their responsibility to answer the educator's request". This means we cannot approve a best thresholds setup at this stage. However, the following table is their suggestion setup.

MALESAbrain learning Threshold θ				
Threshold	*qualification threshold*	*rejection threshold*	*maturity threshold*	*disagreement threshold*
Symbol	θ_{kq}	θ_{kr}	θ_{sm}	θ_{sd}
Setup Value	2 points	-3.2 points	4.2 points	-3 points

Table 1. A suggestion learning thresholds set up in MALESAbrain with 6 participants.

The second experiment, of MALESAassessment, found the keywords in the AK-cells are not well matched the problem issues. It is because learners' assign keywords, during MALESAbrain discussion, will have individuals' viewpoints without integrated into an overview for reusing. It affects the searching results in the test. After we re-assign keywords to each AK-cell, MALESAassessment becomes quite significant for reusing the learned knowledge. The Assessment Report and log file, in section 6.3, show one of the successful retained cases after test. This means there are no difficulties to give a proper mark to a successful answered student and give a proper advice to a failure answered student in this experiment.

8. Conclusion

In this chapter, we propose two intelligent algorithms to save the educator the time for acquiring learning knowledge in PBL environment. The first algorithm builds up MALESAbrain as an intelligent system to acquire students' knowledge in PBL discussion. It will help students integrate their knowledge by critical thinking on different angles of a problem through on-line discussion. The second algorithm builds up MALESAassessment as an evaluation system to test students' performance after PBL discussion. These two algorithms work together to reduce educator's efforts for connecting PBL to IT education.

MALESAbrain algorithm saves the educator the effort of searching for important knowledge pieces generated by students' discussions through its automatic calculations. It reduces the pressure of time in the teaching schedule for coaching PBL discussions in the IT course. Consequently, MALESAbrain has contributed three notions to make the PBL discussion more effective and efficient for knowledge acquisition:

1. The first notion is the created data structure – coined as "AK-cell" (see Definition 1-4) – for cooperative learning, which combines knowledge-content and knowledge-weight. It allows the discussion-knowledge to become calculable in a threshold system. The knowledge-contents become mobile because of the combination of knowledge-weight and knowledge-content in the data structure. Knowledge-weight ranks AK-cells into different important locations, based on learners' judgments and the thresholds set up. It helps learners pay attention to consensus knowledge for more discussion; and to think about why certain issues accumulate different scores from other issues.
2. The second notion is the autonomous decision-making mechanism – called "learning threshold" (see Definition 1-5). It helps the learning system automatically arrange and order construction of a hierarchical knowledge base. It helps students to identify the importance of the issues in a problem.
3. The third notion is the dynamic structure of the knowledge base – called "growth-factor" (see Definition 1-6). It helps the data structure AK-cell to be constructed according to the educator's viewpoint; whatever is right in his/her coaching for knowledge acquisition.

The value of the second algorithm MALESAassessment is the assessment design, which uses the student performance test to refine the learned knowledge. It offers learners a chance to retrospect their learned knowledge and refine them to solve another problem in test. The test assesses learners' concept in making an action plan and examines their abilities to solve a physical troubleshooting problem.

The design feature of MALESAassessment is based on the principles described on the chapter "From case-based reasoning to problem-based learning" (Eshach & Bitterman, 2003) and four-stage CBR-cycle design on the book of "Applying Case-Based Reasoning: Techniques for Enterprise Systems" (Watson, 1997) (see Fig. 13). The four-stage open-ended cycle and concepts-in-action designs help to refine an existing case and construct a new case from the raw/previous cases. In the assessment, learners retrieve the similar cases from the CBR knowledge base; reuse the cases to attempt to solve the existing problem; revise the proposed solutions to solve the new problem; and retain the new solution as their answer to solve the test problem.

9. References

Biljon, J. A. v., Tolmie, C. J., & Plessis, J. P. d. (1999). MagixDan ICAE system for Problem-Based Learning. *Computers & Education, 32*, 65-81.

ComputerHope. (2004). *Computer Hope.com*, [web site]. Available: http://www.computerhope.com/ [15/4/2004].

Conway, J., & Sharkey, R. (2002). Integrating on campus problem based learning and practice based learning: issues and challenges in using computer mediated communication. *Nurse Education Today, 22*, 552-562.

Cooke, M., & Moyle, K. (2002). Students' evaluation of problem-based learning. *Nurse Education Today, 22*, 330-339.

Culvenor, J. (2003). Comparison of team and individual judgments of solutions to safety problems. *Safety Science, 41*, 543 –556.

Eshach, H., & Bitterman, H. (2003). From case-based reasoning to problem-based learning. *Academic Medicine, 78*, 491-496.

Fisher, A. (2001). *Critical Thinking an Introduction* University of Cambridge.

IBM. (2004). *IBM Education: Solutions and resources for IT and business professionals, educators and students*, [web site]. Available: http://www-306.ibm.com/software/info/education/ [16/4/2004].

K.Kaye, B., & J.Johnson, T. (2004). A Web for all reasons: uses and gratifications of Internet components for political information. *Telematics and Informatics, 21*, 197-223.

Lewis, M. J., Davies, R., Jenkins, D., & Tait, M. I. (2001). A review of evaluative studies of computer-based learning in nursing education. *Nurse Education Today, 21*(1), 26-37.

Microsoft. (2004). *Microsoft.com*, [web site]. Available: http://www.microsoft.com/homepage/nav.asp?pid=1 [16/4/2004].

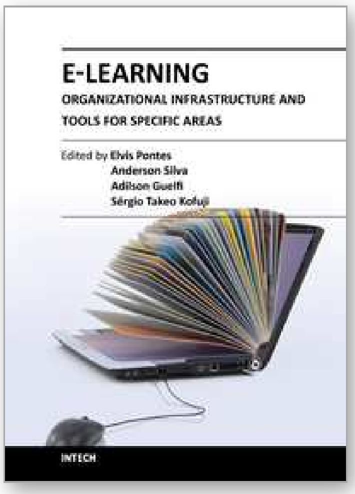
Mitchell, T. M. (1997). *Machine Learning* McGraw-Hill.

Paul, S., Haseman, W. D., & Ramamurthy, K. (2004). Collective memory support and cognitive-conflict group decision-making: an experimental investigation. *Decision Support Systems, 36*(3), 261-281.

Roberts, D., & Ousey, K. (2003). Problem based learning: developing the triggers. Experiences from a first wave site. *Nurse Education in Practice,* (3), 1-5.

Torp, L., & Sage, S. (2002). *Problems as Possibilities: Problem-Based Learning for K–16 Education* Association for Supervision & Curriculum Development.

Watson, I. (1997). *Applying Case-Based Reasoning: Techniques for Enterprise Systems* Morgan Kaufmann Publisers.



E-Learning-Organizational Infrastructure and Tools for Specific Areas
Edited by Prof. Adilson Guelfi

ISBN 978-953-51-0053-9
Hard cover, 182 pages
Publisher InTech
Published online 17, February, 2012
Published in print edition February, 2012

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821