Discovering decision knowledge from web log portfolio for managing classroom processes by applying decision tree and data cube technology

Gwo-Dong Chen, Chen-Chung Liu, Kuo-Liang Ou

Institute of Computer Science and Information Engineering,
National Central University, Chung-Li, Taiwan
Email:{chen, christia, klou}@db.csie.ncu.edu.tw
Tel: 886-3-4227151 ext 4504
Fax: 886-3-4222681
Baw-Jhiune Liu
Department of Computer Engineering and Science
Yuan Ze University, Chung-Li, Taiwan
Email: bjliu@cs.yzu.edu.tw

Abstract

In conventional classrooms, teachers attempt to enhance instruction by monitoring students’ learning processes and analyzing their performances by paper records and observation. Similarly, distance learning systems on the Web should be designed to record students’ behaviors to assist teachers in assessing performance and making decisions related to curriculum. Recent developments in web server systems can record the students’ access to the learning systems in web logs. Information processing analysis on the historical classroom processes can help teachers to develop knowledge for applying proper teaching strategies according to available information in web logs. However, teachers can not easily infer the pedagogical meaning of web logs and discover the pedagogical rules of students’ behavior patterns in the web logs to refine teaching strategies. Therefore, to use web logs for pedagogical purposes, this paper adopts decision tree and data cube information processing methodologies to observe students’ behaviors and discover the pedagogical rules on students’ learning performance from web logs. The architecture and guidelines of utilizing the data cube and decision tree methodologies for pedagogical purposes are also presented. Consequently, teachers can efficiently
estimate and explain the effectiveness of pedagogical strategies, ultimately improving instruction with
decision tree and data cube software.

I Introduction

A major component of effective teaching is the need to make decisions, and so, as a decision process,
teaching creates a valuable learning effect [4]. Educators have conferred that a teacher should be a
manager of the learning experience rather than an information provider [18]. Restated, in addition to
providing concepts, teachers must also observe students and make decisions to apply effective teaching
strategies that would facilitate learning. Wilhelm [36] proposed a method to analyze learning factors so
that teachers can accurately predict students’ learning performance at an early stage. Peterson et al. [22]
also indicated that teachers must make decisions to adapt proper courseware, instructional activity, and
strategies. For example, Shavelson and Stern [27] examined how teachers should adopt proper
instruction pace to improve group learning for different types of students. Meanwhile, teachers in a
distance learning environment must also observe student learning behaviors and analyze the
relationship among students’ learning behaviors, characteristics, and pedagogical strategies.

Courses are widely offered on the World Wide Web. Many distance learning systems, including WebCT
[35], TopClass [32], and Virtual-U [34], allow students to conveniently learn via the Internet. Through
the web-based learning system, students can perform various learning activities in a virtual classroom
[11], such as reading, messaging, conferencing, accessing documents, and participating in interactive
activities. Kuechler [12] indicates that providing additional communication channel, offering
substantive contents, and allowing students to engage in active learning are the major ways to use Web
in undergraduate classrooms. Regardless of what function a web learning system can provide, teachers
must manage classroom processes to enhance instruction. Peterson and Clark [21] indicated that
effectively managing classroom processes involves instruction, observation, and decision making to
enforce proper particular strategies. Conventionally, teachers monitor students’ learning processes by
paper records as well as interactive observation including students’ behaviors in discussion, homework,
and projects. The learning process records are an integral part of student portfolios [20] for assessing
learning performance and improving instruction [31]. However, teachers of web-based learning
systems can not interact with students face to face. Under such a circumstance, teachers must diligently strive to observe students’ learning behaviors and utilize observed phenomenon to effectively manage classroom processes.

To assist teachers in assessing the students’ performance and making decisions related to curriculum, distance learning systems must be designed to record students’ behaviors. Recent advances in general web server systems such as Microsoft IIS(Internet Information Server) can record the students’ access in web logs. Some analytical works have attempted to process the logs for web server management, e.g. AccessWatch, Analog, Gwstat, and WebStat [15, 16, 29, 33, 38]. However, these logs are not designed and maintained for pedagogical purposes. Current analysis instruments for the logs can not satisfy pedagogical requirements. The teachers have to process the web logs in detail to infer their pedagogical meaning, such as time distribution of discussion behavior, number of articles read, and number of questions asked to understand students’ statuses. In addition, estimating the effectiveness of a particular pedagogical strategy from web logs is extremely difficult. Teachers must tediously process the web logs to discover the effectiveness of the pedagogical strategy for different types of students according to the teachers’ previous experience from classroom processes.

**Issues of decision making in web-based learning systems**

Decision making of classroom processes involves observing students’ behavior, analyzing historical data, and estimating the effectiveness of pedagogical strategies. While using web-based learning systems, teachers encounter several difficulties when attempting to analyze and observe students by the web logs. These difficulties are summarized as the following three problems.

First, understanding and improving classroom processes depend on examining how teachers and students behave and think [10]. To understand a student’s learning performance, the teachers must observe students from various dimensions, e.g. behaviors of students, learning performance at different time, and content that the students’ behaviors is related to. For instance, teachers must answer questions such as “How many times did a student offer an opinion?”, “How many times did the student answer others’ questions?”, “How did the students’ behaviors change after a particular pedagogical strategy is enforced?”, and “How many articles related to a special topic did the students read?”. Although the students’ behavior on the web-based learning system is recorded in web logs, the web logs are
organized for web server managers to modify the web site structure and performance. The behavioral records in the web logs are in an improper format for pedagogical analysis and observation. Moreover, obtaining necessary learning records to observe students from the web logs is extremely difficult. We refer to this issue as the behavior observation problem.

Second, viewing instruction as a decision process requires consideration of when a particular action or strategy can be adapted [26]. That is, the teachers must know the relationship among pedagogical strategies, students' learning behavior, and student characteristics. For instance, the teacher may want to know “How does a pedagogical strategy impact different types of students?” and “Does a pedagogical strategy improve the interaction between students who rarely discuss topics with others?” The web logs record many learning records of experienced learning processes, allowing teachers to make decisions by examining similar circumstances in the historical classroom processes. Conventionally used analytical techniques such as k-nearest-neighbor methodology assume a linear relationship between variables and work optimally when variables are independent of each other [30] [36]. Nevertheless, predicting the effectiveness of a strategy from dozens of available learning variables by using a liner decision boundary is extremely difficult. Such difficulty is owing to that the learning records contain a large amount of various types of complex data and interweave with each other. These analytic methodologies require statistical expertise in analyzing the data in the web logs. Therefore, a teacher requires convenient support for inducing the decision rules that can explain the relationship among teaching strategy, students learning behavior, and student characteristics from learning records in historical classroom processes. We refer to this issue as the decision rule problem.

Third, while estimating the effectiveness of a pedagogical strategy, teachers have to answer what-if questions such as “What would the students’ reaction be if a particular pedagogical strategy were to be applied in the current learning course?” To answer this question, the teacher must compare the learning records of students in the current classroom process with similar processes. Teachers can also apply decision rules that are discovered from experienced classroom processes to predict the reaction of students of different types. However, as mentioned earlier, the learning records of students in the web logs are not in an improper format for pedagogical analysis. Moreover, teachers have difficulty in comparing the learning records to estimate the students’ reactions to particular teaching strategies. We refer to this issue as the decision estimation problem.
Purpose of Study

This paper presents a novel methodology to assist teachers in managing classroom processes. The methodology utilizes database systems and decision tree technology on web logs to solve teachers’ problems in decision making. Recent database management systems such as Microsoft SQL Server 7.0 provide data warehouse and cube facilities to effectively manage students’ learning records and codify them to easily observe students’ behavior performance. Previous study investigated a data cube framework to assist teachers in exploring the web logs from multiple dimensions. The database system can solve the behavior observation problem. However, teachers find it too complex to analyze the relationship between strategies and students’ characteristics when many dimensions are involved.

Thereafter, the proposed methodology utilizes decision tree technology to facilitate teachers not only in analyzing the relationship among learning behavior, student characteristics, and pedagogical strategies, but also in estimating the effectiveness of a particular pedagogical strategy. Decision tree software can help teachers to extract pedagogical rules and represented the extracted rules as a flow-chart like decision tree. With this methodology, teachers can go online and efficiently use the web logs to analyze the web-based classroom for improving instruction. By way of a database system, teachers of web-based learning systems can easily understand the progress of students from the web logs, thus allowing them to interact with learners. Meanwhile, historical data in the web logs can be used to reach decisions on how to effective implement pedagogical strategies through the medium of decision tree technology.

The rationale of the use of decision tree software relies on the information processing theory. In conventional classrooms, teachers manage classroom process according to their experience and pedagogical knowledge of teaching. Similarly, in distance learning environment, exactly how to adapt an effective strategy to a certain type of students requires pedagogical guidelines. Teachers may process the data of experienced classroom processes to obtain pedagogical decision knowledge. Both artificial neural network and information process analysis can help teachers to simulate and learn pedagogical knowledge from past classroom processes. However, the learned pedagogical knowledge by neural network software, generally a complex mathematical function, is often difficult for humans to interpret. Learned pedagogical knowledge is less easily communicated to humans than
learned decision rules [19]. On the other hand, decision tree software learns the pedagogical rules from the past classroom processes based on the information processing model. It represents the learned pedagogical rules as a set of decision rules which resemble a flow chart. Hence, teachers can easily perceive and utilize the learned pedagogical knowledge to improve the effectiveness classroom processes.

The rest of this paper is organized as follows. Section II presents our web-based learning environment and the method used to analyze students’ performance from web log portfolios. Section III describes how data cube and decision tree program software can help teachers to observe and analyze students’ performance in the learning environment. Section IV illustrates the architectures and procedure deemed necessary to analyze students’ performance in a web-based learning system. Conclusions are finally made in Section V.

II The Illustrative Learning Environment for Using Web Log Portfolios

This work presents a web-based learning system to elucidate teachers’ role in effectively managing classroom processes and the method used to analyze students’ learning performance. The web-based learning system is designed and implemented for students at National Central University (Chungli, Taiwan) as a conductive environment to learn C++ programming language. Eighty-five students participated in the learning programs. Students are divided into groups for discussing concepts. The discussion activity is based on IBIS discussion model [13]. In IBIS model, as illustrated in Fig. 1, students start with an issue. Many positions are then proposed to resolve the issue. Many arguments are then proposed to support or object to a certain position. The IBIS model is effective in collaboratively resolving design problems. Conklin and Begeman developed a graphic interface to represent the IBIS model in hypertext style [6].
In the web-based learning system, students can perform various types of IBIS discussion-related actions, such as reading articles, asking questions to raise issues, offering opinions to resolve issues, and replying to an article to support or object to an opinion. The teacher of the web-based learning system must observe how students behave in the IBIS discussion. They need to know the following variables that reflect the discussion performance of each student:

- frequency of login to use the learning system ($\text{LoginF}$),
- frequency of reading articles ($\text{ReadF}$),
- frequency of posting opinions ($\text{OpinionF}$),
- frequency of asking questions ($\text{AskF}$), and
- frequency of replying to opinions ($\text{RepF}$).

The teacher must also observe the discussion performance of students at different time periods as listed in Appendix A. For example, the teacher may require the above variables before ($\text{LoginF_B}$, $\text{ReadF_B}$, $\text{OpinionF_B}$, $\text{AskF_B}$, $\text{RepF_B}$) and after applying a teaching strategy ($\text{LoginF_A}$, $\text{ReadF_A}$, $\text{OpinionF_A}$, $\text{AskF_A}$, $\text{RepF_A}$). Therefore, the teacher must observe discussion performance from the perspectives of behavior types and time.

```
<table>
<thead>
<tr>
<th>IP</th>
<th>Student ID</th>
<th>Time</th>
<th>Method</th>
<th>File requested</th>
<th>Result</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>140.115.50.201</td>
<td>Linms</td>
<td>11/May/1998/11:02:41</td>
<td>GET</td>
<td>/cgi/vc/Login.exe</td>
<td>200</td>
<td>236</td>
</tr>
<tr>
<td>140.115.50.201</td>
<td>Linms</td>
<td>11/May/1998/11:03:05</td>
<td>GET</td>
<td>/cgi/vc/Read.exe?article=980231</td>
<td>200</td>
<td>556</td>
</tr>
<tr>
<td>140.115.50.201</td>
<td>Linms</td>
<td>11/May/1998/11:03:05</td>
<td>GET</td>
<td>/cgi/vc/chap.jpg</td>
<td>200</td>
<td>20</td>
</tr>
</tbody>
</table>
```
Table 1. A snapshot of web logs

Current web servers record each web access history in web logs. Original web log records contain the following information: IP address, the user login id, the date and time, the method (GET or POST), the file name requested, the result of the request (success, failure, or error), and the size of file transferred. Table 1 displays a snapshot of the web logs. The web logs are maintained to manage the web server. The logs contain only the files that students have read and application programs requested. The teacher can not easily observe students’ discussion behavior from the web logs, accounting for the need for a methodology to extract learning variables such as ReadF and OpinionF for each student from the web logs. Therefore, the teacher needs a repository to store discussion behavior history and allow teachers to view discussion performance from the perspectives of different behaviors and time intervals.

To effectively manage the IBIS discussion, the teacher must know how pedagogical strategies, students’ characteristics, and discussion performance are related. For instance, the teacher must answer the following questions to regulate effective strategies:

- Does a pedagogical strategy increase the discussion interaction between students?
- How does a pedagogical strategy affect different types of students with respect to gender, age, job and discussion behavior before applying such a strategy?
- What kind of students’ discussion performance would remain unaffected by a teaching strategy?

Analysis of discussion history in the web logs can help a teacher understand how pedagogical strategies affect discussion performance and, in doing so, estimate the effectiveness of a particular strategy. In other words, the teacher can induce potential decision rules that explain the effectiveness of
a particular pedagogical strategy for different types of students from the web logs. The induction involves analyzing the students’ features and the relation between these features and behavior change after a pedagogical strategy is enforced. From the induced results, the teacher can know how students’ features such as gender, age, and job as well as the behavioral features LoginF, ReadF, OpinionF, AskF, and RepF affect students’ performance after the pedagogical strategy is enforced, i.e. LoginF, ReadF, OpinionF, AskF, and RepF. Since the induction process involves a large amount of students’ features and discussion performance history, the teacher needs analytical facilities to induce the relationship between students’ features, teaching strategies, discussion behavior, as well as to estimate the effectiveness of these strategies.

![Decision tree](image)

**Figure 2. Decision making support with database and decision tree technology**

Figure 2 illustrates how teachers of web-based learning systems utilize web logs to analyze and observe learning behaviors via a database and decision tree technologies. The database system provides data cube functions for teachers to observe student behaviors from different perspectives. The database system functions as a repository to obtain necessary variables for advanced analysis. In addition, teachers generally learn pedagogical knowledge to know in which condition a strategy will improve students’ learning effect according to their experiences. Information processing analysis on the historical classroom processes can help the teachers to develop knowledge for applying proper teaching strategies according to available information such as behavioral performance in web logs. The decision tree software programs facilitate the teachers in discovering decision rules from historical data and represent it as a flow-chart like decision tree. The teachers can then utilize the experienced rules in the decision trees to estimate, predict the effectiveness of strategies, and diagnose classroom processes.
Ill Reasons for using database and decision tree technologies

Exactly how pedagogical strategies affect students’ behaviors in a distance learning system has received considerable attention. For example, Wissick et al. [37] investigated the condition to improve students’ involvement in electronic mail learning activities. Lee [14] also investigated strategies for a better on-line instruction in a web-based learning environment. Those investigations have provided guidelines on how to employ feasible teaching strategies for maximizing learning on the Internet. That is, the conditions under which students learn more efficiently have been of particular concern. However, teachers of a web-based learning system require more than simple guidelines on how to implement pedagogical strategies. Instead, teachers must inspect how the strategies affect different types of students. Integrating the data cube and the decision tree technologies can provide online data processing functions to analyze relationships among pedagogical strategies, learning behavior, and learning performances from the web logs. Data cube can manage student learning records in the web logs and provides a relatively easy means of instantaneously observe students’ reactions and behaviors from multiple perspectives. In addition, the decision tree technology can locate potential decision rules from the web logs to explore how pedagogical strategies, students’ reaction, and students’ significant features are related. The steps of using data cube and decision tree on web log portfolios for such pedagogical purpose are listed as follows:

- Reorganizing the web logs for easy understanding of students’ behavior using database query language SQL,

- Calculating and observing students’ performance from multi-dimensional views using data cube software,

- According to the observation, identifying the performance to be analyzed and the features of students that may affect the performance,
Analyzing the relationship between performance and students’ features to obtain decision rules by using decision tree software,

Estimating or predicting students’ performance according to feasible students’ feature before applying a strategy using the discovered decision rules.

In the following, we describe in detail how integrating database cube and decision tree technologies satisfies the teachers’ requirement in analyzing students’ learning performance and making decisions.

Reorganization of web logs

To help teachers observe students’ behavior, web logs must initially be reorganized into a structure that is appropriate for pedagogical observation and analysis. A repository is required to provide various views from behavior and time dimensions. A database management system can be employed as a repository of students learning behavior since it provides facilities for reorganizing the web logs, obtaining multi-dimension views, and acquiring learning variables for advanced analysis.

According to Table 1, the original web logs contain only the accessing history of files in a web learning system. Teachers can not easily understand the pedagogical meaning of the access history. Therefore, the initial step for utilizing the web logs in observation involves inferring the pedagogical meaning of each file access entry in the web logs. For instance, the second entry of the web logs in Table 1 indicates that student Linms read an article at 11/May/1998/11:03:05 since students can only read articles by requesting the application program file Read.exe.

However, in many cases, web site managers may implement an application program file for different learning activities. For instance, web site managers can implement an application program argue.exe for students to support and object to a proposed position since the support and object-to activities are solely write arguments into web servers. In this case, the web server will record only that students requested application program argue.exe. We can not easily distinguish whether a student supported or objected to a position unless we explore the content of the argument. Such design and implementation of a web learning system cause inconvenience for teachers to observe students’
learning behavior.

Therefore, to precisely infer a pedagogical meaning of the web logs, the web-based learning system must be designed and implemented according to the teachers’ observation requirement. That is, the learning behaviors that the teachers need to observe, e.g. reading articles, posting opinions, and asking questions, must be implemented as distinguishable application program files. Thereafter, the teachers can infer the pedagogical behavioral history from the web logs. Table 2 summarizes the mapping from web logs entries to pedagogical behaviors in our web learning system.

<table>
<thead>
<tr>
<th>Pedagogical meaning</th>
<th>Actions recorded in web logs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login</td>
<td>Request Login.exe in Web Server</td>
</tr>
<tr>
<td>Read Article</td>
<td>Request Read.exe in Web Server</td>
</tr>
<tr>
<td>Ask Question</td>
<td>Request Ask.exe in Web Server</td>
</tr>
<tr>
<td>Post Opinion</td>
<td>Request Opinion.exe in Web Server</td>
</tr>
<tr>
<td>Reply Opinion</td>
<td>Request Reply.exe in Web Server</td>
</tr>
</tbody>
</table>

Table 2. Pedagogical meaning of the web logs

To obtain a complete pedagogical behavior history, we have to infer the pedagogical meaning from the web logs. In addition, the web logs contain many irrelevant log entries for observing the learning behavior. The irrelevant events include the events of server request failure, authentication failure, and image file request. For instance, the third and sixth entries in Table 1 are the request for an image file that embeds in an article. Such log entries are irrelevant for the pedagogical observation. Our experience of C++ programming language shows that the web server records over 60,000 log entries in three months. Therefore, the teacher needs data processing facility to process the large amount of the web logs to obtain the students’ behavior history.

<table>
<thead>
<tr>
<th>Student ID</th>
<th>Pedagogical behavior</th>
<th>Article</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linms</td>
<td>Login</td>
<td></td>
<td>11/May/1998/11:02:41</td>
</tr>
<tr>
<td>Linms</td>
<td>Read Article</td>
<td>980231</td>
<td>11/May/1998/11:03:05</td>
</tr>
<tr>
<td>Jcc</td>
<td>Post Opinion</td>
<td>989256</td>
<td>11/May/1998/11:03:27</td>
</tr>
<tr>
<td>Jcc</td>
<td>Read Article</td>
<td>980512</td>
<td>11/May/1998/11:05:12</td>
</tr>
<tr>
<td>Jcc</td>
<td>Reply to Opinion</td>
<td>980512</td>
<td>11/May/1998/11:06:54</td>
</tr>
</tbody>
</table>

Table 3. Pedagogical behavior history

Query Language SQL (Structured Query Language) is a powerful data processing facility of a database. Most database management systems support SQL interface. SQL provides an easy interface for teachers to process the data into a single table and relationship between multiple tables. With the
SQL data processing functionality, the original web logs (Table 1) and the pedagogical meaning of the web logs (Table 2) can be joined to obtain the behavior history from a pedagogical perspective. The join operation of SQL allows us to either infer the meaning of a web log entry or filter out the irrelevant log entries, and produce pedagogical behavior history in Table 3 for teachers to observe students’ discussion behavior. Therefore, the use of database management systems diminishes many efforts to reorganize the web logs for pedagogical observation and analysis.

Multi-dimension views on behavioral history

The following describes how data cube technologies can be used to observe the students’ behavior from multi-dimension views. By applying data cube operations, teachers can easily calculate the students’ performance in different learning behaviors. For instance, teachers can obtain the students’ performance in posting and reading articles. Teachers can also trace in detail the level of the learning performance in posting behavior such as proposing opinions, asking questions and answering others’ questions. From the behavior type dimension, the teachers can perceive each student’s behavioral characteristic. In addition, Teachers can observe students’ learning performance at different time periods. For instance, they can apply data cube operations to obtain students’ learning performance before and after a pedagogical strategy is applied. Consequently, the teacher can understand the effectiveness of the pedagogical strategy.

In this study, the teacher regulated a question and answer (Q&A) activity to encourage students to reflect on their comprehension of learning concepts after class. In the Q&A activity, the teacher raised a question and encouraged students to answer by announcing that the participation in the Q&A activity will contribute to a certain percentage of the final score. The teacher requires knowing how the Q&A strategy affects students’ behaviors in IBIS discussion. The teacher can on-line monitor the discussion behaviors according to behavior type and time dimensions. The teacher can initially define the behavior types to be monitored and the hierarchy relationship among these behavior types. At the most abstract level, the teacher can observe how students behave in login, reading, and posting articles. To drill down the posting behavior, the teacher can observe how students behave in proposing opinions, asking questions, and replying to opinions. Those actions are in the category of the posting articles. In addition, the teacher can define time intervals to observe these discussion behaviors at different time
intervals. They can define the behavior hierarchy and time intervals, as depicted in Table 4 and Table 5, to direct the data cube functions of a database management system to calculate the summary information. By manipulating the time and behavior type dimension on the data cube, the teacher can observe how students’ behaviors change over time and strategies.

Table 4. Behavior hierarchy of observed behavior types

<table>
<thead>
<tr>
<th>Time</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 days before Q&amp;A (BQ&amp;A)</td>
<td>17/May/1998-6/Jun/1998</td>
</tr>
</tbody>
</table>

The behavioral type and time dimension form two basic axes to observe the students’ behavioral performance. The data cube technology provides several operations, including rolling up, drill-down, cross-tabulation, pivot, flexible period definition, and sub-total, to manipulate cube repository [17]. For instance, the teacher can initially observe how students behave in Login, Reading, and Posting Behavior, then drill down into Posting behavior to know how students behave in Ask, Opinion, and Reply behavior, or drilling down into desired time intervals to know how students behave in these intervals. Table 6 presents the hierarchical behavioral summary information on behavior type and time dimensions. The data cube operations allow teachers to observe how students behave towards different behavioral types and different time intervals in any detail level.

Table 5. Time interval definition

Table 6. Cube on behavior and time dimension

The teacher can easily obtain summary information of the students’ behavioral performance. The
cube operations can also facilitate the teacher in acquiring behavior variables of each student for advanced analysis, e.g. LoginF, ReadF, OpinionF, AskF, RepF. LoginF, ReadF, OpinionF, AskF, and RepF. Considering the dimension of Student dimension allows us to acquire the variables of each student. Table 7 displays the learning variables of students that acquire from drilling down the cube in Table 6 into each student level.

<table>
<thead>
<tr>
<th>Var</th>
<th>LoginF</th>
<th>ReadF</th>
<th>OpinionF</th>
<th>AskF</th>
<th>ReplyF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>35</td>
<td>10</td>
<td>2</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>32</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>113</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>15</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>32</td>
</tr>
<tr>
<td>2</td>
<td>62</td>
<td>148</td>
<td>13</td>
<td>10</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>98</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>332</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>33</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>40</td>
</tr>
<tr>
<td>3</td>
<td>23</td>
<td>72</td>
<td>3</td>
<td>3</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>67</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>237</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>29</td>
</tr>
<tr>
<td>4</td>
<td>29</td>
<td>95</td>
<td>18</td>
<td>5</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>44</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>179</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>23</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>30</td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>40</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>29</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>74</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6</td>
</tr>
</tbody>
</table>

Table 7. The behavioral variables of students acquired from data cube

Above description illustrates the methodology of using data cube on web log to observe students’ learning behavior from a multi-dimensional and hierarchical view. The multi-dimensional and hierarchical view enables teachers to select desired dimension and drill down to observe students in any desired detail. Teachers can observe students’ behavior, starting from post behavior, then drilling down into details of post behavior (i.e. asking a question, posting an opinion, and replying to a question), finally into specific time period of these post behaviors. Despite of this, data cube software allows teachers directly select any detail type of post behavior to obtain the summary information of this behavior type. For instance, teachers can select the asking questions on the behavior type dimension and the time period after Q&A activity on the time dimension. The data cube will summarize the frequency of asking question of students after Q&A. Teachers need not to observe students’ behavior by drilling down from the most abstract level of behavior type dimension.

The data cube enables teachers to observe online the students’ behavior from multiple dimensions and acquire learning variables. However, the problem of “How does a teaching strategy affect behavior of different characteristics of students?”; “What kind students do not improve in terms of the behavioral performance after a particular pedagogical strategy is implemented?” need data cube operations on all reticular dimensions that reflect the basic characteristics of students, e.g. gender, job, and age as well as
behavioral characteristics such as LoginF, ReadF, OpinionF, AskF, and RepF. It is a complicated task for teachers to analyze the relationship between strategies and students’ characteristics, particularly when many dimensions are involved.

In addition, by using data cube to analyze the students’ behavior, the teacher must determine the measure of each dimension. For instance, whether if the replying behavior before Q&A strategy (RepF_b) dominates the reaction towards the strategy must be determined. Correspondingly, the teacher can define the meaning of “active” and “inactive” in replying behavior and divide the students into “active” and “inactive” groups by a clear division of the frequency of the replying behavior. Data cube software can then explore the difference of reactions to the Q&A between “active” and “inactive” student groups. In doing so, the teacher must carefully examine the distribution of students’ replying behaviors and define the meaning of “active” and “inactive” groups. The teacher must avoid any significant group of students that only deviate slightly in replying behavior but are similar in their reaction to the pedagogical strategy being split into two different groups, thus losing some potential relationships among these dimensions during the data cube operations. This is also a complex task for teachers.

Decision tree technology can help teachers induce the relation between pedagogical strategies and characteristics of students from a variety of dimensions. Induction is made by continuously examining the distribution of each dimension from the historical data. The induction analysis discovers potential groups that have similar characteristics and reaction to a particular strategy. Hence, integrating data cube and decision tree technologies allows the teacher to efficiently and online perceive the reaction of different types of students to a teaching strategy. Figure 3 illustrates how to apply decision tree technology to analyze the reaction of students for a pedagogical strategy.
Decision tree analysis

This section describes the complexity that a teacher has in analyzing the relations among pedagogical strategies, students’ characteristics and reaction. The teacher can use the decision tree software programs to induce the relationship between pedagogical strategies and students’ behavioral reaction to such strategies. The induction process is based mainly on the entropy of students. Entropy refers to the impurity of a group of students. The induction process of decision tree programs discovers the decision rules for classifying students into several groups according to the values of the identified learning variables such as gender, age, job, and behavior patterns, as illustrated in Table 7. Each group of students has the lowest entropy in some variables and reaction to pedagogical strategies. Therefore, students in a group have similar features and tend to react to a pedagogical strategy similarly. Notably, the induction process is based on historical data.

Recently developed decision tree software programs such as C5.0/See5 [23][24] and IBM DB2 intelligent miner assist decision makers in exploring the decision rules from historical records. Many business decision support systems such as Blue Martini Software [1] and Clementine [5] applied decision tree technology to perform tasks such as predict market share, assess financial risk, and diagnose faults. Feigenbaum et al. [7] demonstrated how to use a decision rule system to assist credit authorizers. The decision rule system can discover the decision rules from customer credit histories to classify whether a transaction is good or bad.
Decision tree software programs also provide instruments to discover decision rules from historical learning records. After feeding the historical values of students’ features, behavioral patterns, and their behavioral performance after enforcing a strategy, decision tree software program learns [19] the decision tree that describes how the identified learning variables affect the effectiveness of a pedagogical strategy. The term “learn” of the decision tree software program refers to the induction process in which the decision tree software program determines not only which learning variables of students dominate the behavioral performance but also how the value distribution of the learning variables affect the behavior performance.

In our experiment, decision tree software C5.0 is utilized to discover how different types of students react to the Q&A strategy. Before the C5.0 analysis process, the teacher initially evaluates the students’ reaction to the Q&A strategy by examining the behavioral change after executing the strategy. The behavioral change include the frequency change of reading articles, proposing opinions, asking questions, and replying to others’ opinions. The behavioral change, denoted as Performance_A, is classified into three classes, High, Middle, and Low which refers to the three levels of effectiveness of the Q&A strategy. Then, C5.0 is used to analyze how student’s gender (gender), age (age), job (job), as well as the frequency of login (LoginF_b), reading (ReadF_b), proposing opinions (OpinionF_b), asking questions (AskF_b), and replying to others’ opinions (RepF_b) before the Q&A strategy affect the effectiveness (Performance_A) of the Q&A strategy. After feeding the value of variables gender, age, job, LoginF_b, ReadF_b, OpinionF_b, AskF_b, RepF_b and Performance_A of the students, C5.0 generates a decision tree to explain how different types of students react to Q&A strategy.
Figure 4 illustrates the simplified decision tree induced by C5.0 in our experiment. The details of the induced decision tree is listed in Appendix B. The experiment accepts eighty five students’ features in gender, age, job, ReadF_B, OpinionF_B, AskF_B, RepF_B and their behavioral performance Performance_A in our C++ programming language course. Each student group (black node) in the decision tree represents a type of students who tend to have a similar behavioral performance in Performance_A after Q&A. Therefore, a group of students in the decision tree depicts a decision rule to explain the Performance_A from available features. The significant rules that have over six students in corresponding groups in the decision tree are described as follows:

- **Group G1** refers to students who rarely reply to others’ opinions (RepF_B <=1) and never propose opinions (OpinionF_B <=0) before Q&A; moreover, they tend to have Low Performance_A after Q&A. Indication 25/0 reveals that all twenty five students of this type had Low Performance_A from the historical data.

- **Student group G2** depicts the students who rarely replied others opinion (RepF_B <=1), proposed opinion at least once (OpinionF_B >0), rarely ask questions (AskF_B <=0) and logined the learning system less than ten times (LoginF_B <=10); they tend to have Low Performance_A. Among nineteen students of this type, fourteen (by the indication of 19/5) have a Low Performance_A.
Student group G8 indicates the students who reply to opinions in the range of 1 and 3 times (RepF_B > 1 and RepF_B <= 3), and rarely ask questions (AskF_B <= 1); moreover, they tend to have Middle Performance_A after Q&A.

Student group G11 refers to female (Gender=Female) students who reply to others' opinions and ask questions more than once (RepF_B > 1 and AskF_B > 1); moreover, they tend to have High Performance_A after Q&A.

<table>
<thead>
<tr>
<th>Conditions</th>
<th>Performance_A</th>
<th>Pr</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1 RepF_B &lt;= 1 and OpinionF_B &lt;= 0</td>
<td>Low</td>
<td>1</td>
</tr>
<tr>
<td>G2 RepF_B &lt;= 1 and OpinionF_B &gt; 0 and AskF_B &lt;= 0 and LoginF_B &lt;= 10</td>
<td>Low 0.736</td>
<td></td>
</tr>
<tr>
<td>G4 RepF_B &lt;= 1 and OpinionF_B &gt; 0 and AskF_B &gt; 0 and Gender=Male and OpinionF_B &lt;= 1</td>
<td>Middle 0.5</td>
<td></td>
</tr>
<tr>
<td>G8 RepF_B &gt; 1 and AskF_B &lt;= 1 and RepF_B &lt;= 3</td>
<td>Middle 1</td>
<td></td>
</tr>
<tr>
<td>G11 RepF_B &gt; 1 and AskF_B &gt; 1 and Gender=Female</td>
<td>High 0.857</td>
<td></td>
</tr>
<tr>
<td>G13 RepF_B &gt; 1 and AskF_B &gt; 1 and Gender=Male and RepF_B &lt;= 15 and OpinionF_B &lt;= 3</td>
<td>High 0.75</td>
<td></td>
</tr>
<tr>
<td>G14 RepF_B &gt; 1 and AskF_B &gt; 1 and Gender=Male and RepF_B &lt;= 15 and OpinionF_B &gt; 3</td>
<td>Middle 1</td>
<td></td>
</tr>
</tbody>
</table>

Table 8. Significant decision rules with support of over four students

Decision rules in the decision tree provide a guideline on how to estimate the effectiveness of the Q&A strategy. For instance, the decision rule of G2 indicates that if a student behaves with RepF_B <= 1, OpinionF_B > 0, AskF_B <= 0, and LoginF_B <= 10 before Q&A, the student has a probability of 73.6% (by 14/19) to have Low behavior performance after Q&A. Table 8 selects the significant decision rules in which over four students support the rules.

The teacher can examine the current students’ distribution and estimate the effectiveness of the pedagogical strategy by using the induced decision tree. The teacher can initially define the performance gain for each class of behavioral performance after Q&A. By using the defined performance gain, current students’ distribution, and the probability of performance gain for each group, the teacher can acquire the expectation of performance gain of a particular pedagogical strategy. For instance, the teacher may define the performance gain for High Performance_A as 5 score. If ten
students satisfy the decision rule of G11, then the expectation of performance gain for this group of student is 42.85 (by 5*10*0.857). Repeating this process for each group allows the teacher to predict and estimate the total performance gain of the strategy. Therefore, the teacher can estimate the effectiveness of the strategy using the induced decision tree.

Teachers can also utilize statistical methods such as k-nearest-neighbor to classify unseen cases in statistics. Quinlan [23] described the difference between decision tree technology and statistical method: “As a general rule, statistical techniques tend to focus on tasks in which all attributes have continuous or ordinal values. Many of them are also parametric, assuming some form for the model and finding the appropriate values for the models’ parameters form the data. For instance, a linear classifier assumes that class can be expressed as a linear combination, and then finds the particular linear combination that gives best fit over the training data.” The statistical methods require accuracy to determine the model among attributes. On the other hand, the decision tree method does not assume any model among attributes. Instead, induction of the decision tree method is performed by continuously examining the distribution of attributes and classifying objects with minimal entropy. Decision tree methodology scans the complete data set and finds any significant relationship among the historical data. The decision tree method is easily implemented and does not presume advanced statistical knowledge. Consequently, it provides a relatively fast and easy means of online analyzing the students’ behavior and discovering potential rules to improve instruction.

In sum, the database and decision tree technologies satisfies a web-based distance learning teacher’s requirement of effectively managing classroom processes because it overcomes the difficulties in online observing learning behavior and decision making. First, the data processing and data cube functions of the database management systems can help the teacher to efficiently process learning records in web logs and summarize required learning variables from multi-dimension views. Second, decision tree software programs discover decision rules from the historical data. Teachers can perceive the relationships among teaching strategies, student characteristics and reactions. Third, the teacher can probabilistically estimate the effectiveness of a pedagogical strategy by using the decision rules discovered from experience. Table 9 summarizes the rationale of using database and decision tree methodologies for managing classroom processes.
### System architecture and implementation

Current web environment is based on Hypertext Transfer Protocol (HTTP). Students can request an HTML (Hypertext Markup Language) file in a web server to read an article. In addition, the web server might be equipped with a set of external CGI (common gateway interface) programs that allow students to write articles into the web server. By using these external programs, students can perform various discussion functions. Current web servers also record each access to the web server in the web logs. Teachers can then use the logs to analyze the students’ behaviors. However, current browsers use a proxy server to accelerate the browsing speed, making the behavior history in the web logs incomplete since the browsers directly download the articles from the proxy server. Furthermore, if the generated articles are stored in individual HTML files, the students’ behavior can not be easily associated with the content of articles. For instance, knowing how many times a student asks about “Inheritance” involves...
processing the web logs and parsing the HTML files that the student wrote into the web server. Therefore, the database technology must be integrated into the web server to completely record students’ behavior and ease the processing learning behavior.

Figure 5 depicts the system architecture to analyze the classroom process in our experiment. The web-based learning system contains three primary components: Microsoft IIS web server, Microsoft SQL server 7.0, and C5.0 decision tree software. The web server accepts the students’ requests and sends students the requested articles. To allow the students to perform discussion activities and to make web server completely record students’ behaviors, the web server contains a set of ASP(Active Server Page) external programs. ASP programs can easily collaborate with SQL server to record and process the students’ discussion articles. The external ASP programs include the functions to read articles, ask questions, propose opinions, and reply to others’ opinions. These functions either write the students’ articles into the database system or retrieve articles from the database to students’ browsers. We recommend encoding these functions in different ASP programs. Each access to these ASP functions causes the web server to record the access information in web logs. Teachers can then perceive what action students have performed from the web logs. Notably, the browser must directly communicate with web server rather than any proxy server since students can only access the web-based learning system by requesting the ASP external programs. Therefore, each learning behavior is faithfully recorded in the web logs.

The SQL server in the architecture is the repository of students’ discussion articles and web logs. This server also provides data processing and cube facilities to reorganize the generated information. Notably, the SQL interface can be utilized to associate the students’ behavior with the content of articles. In addition, the data cube interface allows teachers to view students’ behaviors from multiple perspectives. Teacher can acquire required variables to form a feature space. The feature space records the values of required variables of students for decision tree analysis, e.g. Gender, LoginFb, ReadFb, and Performance. C5.0 decision tree software can then induce the potential rules among these variables from the feature space. In the following sections, we describe the database schema for storing students learning records, data cube construction, and the usage of C5.0.
Database schema of the learning records

Student learning information is stored into a database. Four types of learning records include (1) students’ information such as name, gender, and age (2) discussion articles and the relationship among the articles, (3) actions that students can perform, and (4) the concepts that the students’ behaviors are related to. Figure 6 illustrates the relationship among the learning information by using an Entity-Relationship diagram [3]. Entity-Relationship diagram is commonly used to describe information and data structure. The diagram depicts the records (Rectangle) that should be stored and the relationships (lozenge) among these records. The relationships among these records includes:

- Relationship among ACTION, STUDENT, and ARTICLE denotes that a student may perform actions on discussion articles.
- Relationship between STUDENT and ARTICLE indicates that a student can create many discussion articles.
- Relationship between ARTICLE and CONCEPT implies that an article is related to a set of concepts.
- Recursive relationship of ARTICLE represents the hyperlink among articles. An article may be linked to related articles.

![Diagram of Entity-Relationship schema of learning records](image)

Figure 6. Entity-relationship schema of learning records
Based on the database schema, a relational database contains the following tables to record the learning information of students.

- Table `Student(Student_ID, Name, Gender, Age, Job)` records students’ information.
- Table `Behavior(Student_ID, Action, Article_ID, Time)` tracks each student’s learning behavior.
- Table `Article(Article_ID, Type, Title, Content, Author, Link_to)` records all generated articles. The types of articles include an `Issue`, `Position`, and `Argument` in IBIS model. In addition, an article may include hyperlinks to other related articles through the `Link_to` attribute.
- Table `Relate-To(Article_ID, Concept)` depicts what concepts an article is related.

Recording the records of table `Student` in a database is trivial. Teachers can use database applications to store such records in a database. Furthermore, the web site managers of the learning system can store the articles generated by students or teachers in the table `Article` by using ASP external programs. Teachers can also monitor students’ learning behavior from the web logs (as mentioned in the previous section) and record them in table `Behavior`. However, knowing what topic articles are related to requires knowledge of how to process the context of articles. The SQL query interface of SQL server provides context match function to determine whether if an article is related to a particular topic. For example, the following SQL statement can be used to determine the articles that are related to the topic “Multiple Inheritance”.

```sql
SELECT Article_ID
FROM Article
WHERE Content LIKE ‘% Multiple Inheritance %’
```

The query returns the article identifiers that contain the string “Multiple Inheritance”. The LIKE operator performs wildcard searches of valid search string values. Therefore, the SQL query interface can be used to depict what concepts an article is related to in the table `Relate-To`. 
Data cube construction

Above database system records the students’ learning records in four interrelated tables. However, the SQL server 7.0 does not allow teachers to directly define hierarchy of dimensions and automatically calculate the data cube according the teachers’ hierarchy from the interrelated tables. Notably, a table that contains the original behavior records and the meaning of these records on the required dimensions is required to enable the data cube operation on students’ behavior. We need the table

Behavior-With-Dimensions (Student_ID, Action, Article_ID, Time, Q&A, First level behavior, Post behavior)

containing original behavioral records and the meaning of each behavioral record on the following dimensions:

- **Q&A**: whether the learning behavior occurred before or after Q&A. If a record of Behavior-With-Dimensions has Time attribute between 17/May/1998 and 6/Jun/1998, the behavior occurred before Q&A and set the attribute Q&A=BQ&A. If the Time of the record are between 7/Jul/1998 and 26/Jun/1998, the behavior occurred after Q&A and set the attribute of Q&A=AQ&A.

- **First level behavior**: what type of learning behavior the behavioral record represents. In the first level, the behavior types include login, read, and post.

- **Post behavior**: what type of post behavior the behavioral record represents. The post behavior may be one of asking questions, proposing opinions, or replying to opinions.

<table>
<thead>
<tr>
<th>Student ID</th>
<th>Action</th>
<th>Article</th>
<th>Time</th>
<th>Q&amp;A</th>
<th>First Level Behavior</th>
<th>Post</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linms</td>
<td>Login</td>
<td></td>
<td>11/May/1998/11:02:41</td>
<td>AQ&amp;A</td>
<td>Login</td>
<td>X</td>
</tr>
<tr>
<td>Linms</td>
<td>Read Article</td>
<td>980231</td>
<td>11/May/1998/11:03:05</td>
<td>BQ&amp;A</td>
<td>Read</td>
<td>X</td>
</tr>
<tr>
<td>Jcc</td>
<td>Read Article</td>
<td>980512</td>
<td>11/May/1998/11:05:12</td>
<td>BQ&amp;A</td>
<td>Read</td>
<td>X</td>
</tr>
<tr>
<td>Jcc</td>
<td>Reply to Opinion</td>
<td>980512</td>
<td>11/May/1998/11:06:54</td>
<td>AQ&amp;A</td>
<td>Post</td>
<td>Reply</td>
</tr>
</tbody>
</table>
Table 10. Original behavior records and the meaning on the observed dimensions

Based on the table with the meaning in the required dimensions, the SQL server provides a friendly interface for teachers to manipulate a data cube. Teachers can click and drag the attribute of a table to form a hierarchy. SQL server will calculate the cube according to the specified hierarchy. For instance, we can drag the attribute First Level Behavior in Table 10 as the top of a dimension hierarchy. We can then drag the attribute Post, Q&A as descendants of First Level Behavior to obtain the data cube in Table 6.

Usage of the decision tree software programs

To use the decision tree software programs C5.0, the teacher must define the specification of the induction analysis and input the data according to the format specified in the specification. The specification of the induction analysis includes the classes of the object to be classified by the decision tree programs. In our experiment, we must specify the classes of students’ behavioral performance after Q&A, i.e. Performance. The performance is classified as High, Middle, Low according to teachers’ measurement criteria. In addition, the specification also includes the attributes to represent students’ features. C5.0 can accept two types of attributes. One is continuously-valued attributes such as login frequency, and age. The other is nominal-valued discrete attribute, e.g. job type, and gender. Figure 7(a) illustrates the specification of our experiment and Fig. 7(b) displays a portion of the data to induce the decision tree in Fig. 4.

-------------------------------------
High, Middle, Low

Gender: Male, Female
Job: Education, Business, Government, Engineer, Others
Age: continuous
LoginFb: continuous
ReadFb: continuous
OpinionFb: continuous
AskFb: continuous
RepFb: continuous

Figure 7(a). Induction analysis specification of C5.0

-------------------------------------
Female, Business, 25, 10, 35, 10, 2, 21, High
Male, Engineer, 34, 62, 148, 13, 10, 21, High
Male, Education, 39, 23, 72, 3, 3, 14, High
Female, Government, 43, 9, 36, 0, 0, 1, Low
Male, Others, 29, 13, 40, 4, 3, 4, Middle
V Conclusion

This paper describes how a web-based distance learning teacher should effectively manage Internet classrooms. However, in web-based distance learning systems, the teacher encounters difficulties in observing the learning processes and analyzing the students’ learning behaviors. Current analysis instruments on web logs can not satisfy teachers’ requirement. Consequently, a data cube methodology is proposed herein to observe the students’ learning behavior in web logs from multiple perspective. This paper also applies decision tree technology to help teachers analyze historical learning records in the web logs. The experience and guideline for utilizing the data cube and decision tree methodology in pedagogical purpose is depicted by a group discussion example. The decision tree technology can help teachers discover potential decision rules in the web logs in an online and efficient manner. Hence, teachers of a web-based learning system can extend the use of data cube and decision tree technologies to analyze learning record in web logs for pedagogical purposes with relative ease.

Acknowledgments

The authors would like to thank the National Science Council of the Republic of China for financially supporting this research under Contract No. 89–2511–S–008–002–.

Reference


[18] L. C. Meredith, Supporting and facilitating self-directed learning, ERIC Digest, No. 93.


[34] Virtual-U, URL is http://www.vlei.com/ developed by Simon Fraser University.


Appendix A

The use of decision tree software needs to identify the variables that reflect students’ behavior. The terms of measured variables are listed as follows:

- **LoginF**: the frequency of login to use the learning system,
- **LoginF_B**: the frequency of login to use the learning system before enforcing a teaching strategy,
- **LoginF_A**: the frequency of login to use the learning system after enforcing a teaching strategy,
- **ReadF**: the frequency of reading articles,
- **ReadF_B**: the frequency of reading articles before enforcing a teaching strategy,
- **ReadF_A**: the frequency of reading articles after enforcing a teaching strategy,
- **OpinionF**: the frequency of posting opinions,
Appendix B

Figure 7 illustrates the details of the simplified decision tree in figure 4. The induction process classifies students into fourteen groups (size=14). Each student group in the decision tree represents a type of students who tend to have a similar behavioral performance in PerformanceA after Q&A. For example, among nineteen students of group G2 who behaved in the pattern RepFB <=1, OpinionFB >0, AskFB <=0 and LoginFB <=10, fourteen (by the indication of 19/5) have a Low PerformanceA. In addition, the induction process consumes 0.1 seconds and has 11.8% error rate. The error of the induction process originates from a situation in which some students are considered as noise during the induction processes. For example, among twenty three students who have Middle PerformanceA after Q&A, two students are classified into groups with High PerformanceA and five students in groups with Low PerformanceA.

See5 INDUCTION SYSTEM [Release 1.09a]
Read 85 cases (8 attributes) from discussion-Q&A.data

Decision tree:

RepFB <= 1:
  ... OpinionFB <= 0: Low (25.0)--------------------------------------------------------------------------(G1)
  : OpinionFB > 0:
    : ... AskFB <= 0:
      : ... LoginFB <= 10: Low (19.0/5.0) ----------------------------------------------------------(G2)
      : LoginFB > 10: High (2.0/1.0) ---------------------------------------------------------------(G3)
    : AskFB > 0:
      : ... Gender = Male:
Evaluation on training data (85 cases):

Decision Tree

<table>
<thead>
<tr>
<th>Size</th>
<th>Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>10(11.8%) &lt;&lt;</td>
</tr>
</tbody>
</table>

(a) (b) (c) <-classified as

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>1</td>
<td>(a): class High</td>
</tr>
<tr>
<td>2</td>
<td>16</td>
<td>5 (b): class Middle</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>41 (c): class Low</td>
</tr>
</tbody>
</table>

Time: 0.1 secs

Figure 7. Decision tree induced by C5.0 decision tree software program