Improving Pedagogy by Analyzing Relevance and Dependency of Course Learning Outcomes

THOMAS DEVINE*, MAHMOOD HOSSAIN†, ERICA HARVEY‡, and ANDREAS BAUR‡
*Department of Computer Science and Electrical Engineering, West Virginia University
†Department of Computer Science, Math, and Physics, Fairmont State University
‡Department of Biology, Chemistry, and Geoscience, Fairmont State University

Many educators utilize an outcomes-based approach these days and maintain student performance records with regards to the individual learning outcomes. However, extracting meaningful information from these ever-growing datasets is a daunting task for even a skilled statistician. In this paper, we describe the implementation of a user-friendly software tool called DMOBE (Data Miner for Outcome Based Education). This tool was developed to extract key learning patterns from student performance records accumulated by educational programs following an outcome-based instructional paradigm. This tool allows instructors of such courses to mine their data and interpret the results in such a way as to provide insights into course optimization and more effective teaching methods. Specifically, this tool uses supervised feature selection to discover the relevant learning outcomes in a course based on their ability to predict student performance in a subsequent course and then uses dependency mining to determine whether mastery of any other outcomes in the course will influence mastery of a given outcome.

1. INTRODUCTION

Outcome-based education [William 1994] has recently garnered a lot of attention in higher education. It is a learning paradigm that requires students to demonstrate specific competencies at the end of a set of learning experiences . It is based on defining a clear set of learning outcomes and establishing a learning environment that enables the students to achieve mastery of those outcomes. The analysis of outcome-based assessment data can aid in understanding student learning patterns and the results of such analysis may provide instructors with better insights into student competency and facilitate the development of effective pedagogical methods. With the increased focus by accrediting bodies on the definition and assessment of student learning outcomes, increasing number of educators today utilize an outcomes-based approach in their classrooms and maintain records of student performance on specific learning outcomes. However, extracting meaningful information from this ever-growing collection of data is a challenging task. Data mining methods can be extremely helpful in developing a computational framework for the analysis of course assessment data.

Data mining methods have become very effective data analysis tools in various application domains, primarily because of their ability to deal with large volumes of structured and unstructured data and their ability to discover relevant and non-trivial information without prior knowledge. While traditional database queries can answer questions like "find the students who received an A in CS101" or "find the learning outcomes that Smith could not master in CS102", data mining can provide answers to more abstract questions like "find all students who will possibly succeed in the computer science program" or "identify the critical learning outcomes of CS101". Data mining has recently been used in education research. Examples include discovering potential student groups with similar characteristics [Chen et al. 2000], predicting student grades based on logged data in an online course [Minaei-Bidgoli et al. 2003], predicting student grades from

This work was supported by the Summer Undergraduate Research Experience (SURE) Grant at Fairmont State University.

test scores [Minaei-Bidgoli and Punch 2003], predicting student grades based on actions taken by students in solving homework and exam problems [Minaei-Bidgoli et al. 2004], finding individual learning patterns [Yudelson et al. 2006], building personalized web-based educational systems by discovering learners' needs [Despotovic et al. 2008], predicting student performance based on web usage data in an online course [Romero et al. 2008], and applying data mining in a course management system to extract interesting information for instructors [Romero and García 2008]. These works have primarily focused on test/assignment scores and usage data in online course management systems. To the best of our knowledge, data mining has not been applied to outcome-based assessment data.

The goal of this work was to develop a user-friendly software tool called DMOBE (Data Miner for Outcome Based Education) to provide data mining support for analyzing outcome-based assessment data. We believe that the application of data mining techniques may help instructors improve teaching methods, thereby increasing the quality of education received by their students. On a larger scale, application of data mining to course assessment data could become a powerful vehicle for supporting programmatic assessment efforts at an institution. We addressed two specific knowledge extraction problems. The first task was to extract the relevant learning outcomes in a course based on their influence on student performance in a subsequent course. The second task was to discover the dependency of these relevant outcomes on the mastery of other outcomes. We presented a modified association mining [Agrawal et al. 1993] framework, called *dependency mining*, to address the second task.

The remainder of the paper is organized as follows. In section 2, we present the architecture of our tool and briefly describe the different data mining methods that were incorporated into our tool. We also present a modified association mining framework for discovering dependency between course outcomes. In section 3, we describe our datasets and present the experimental results. Finally, in section 4, we will provide concluding remarks and scope of future research.

2. DATA MINING FOR OUTCOME BASED EDUCATION

2.1 Overview

Figure 1 shows the data mining workflow that we utilized for discovering relevance and dependency of course learning outcomes using course outcome assessment data. Our system first allows users to select and prepare data that is currently saved in a CSV (comma separated values) or spreadsheet file (e.g., a spreadsheet exported from Microsoft Excel, or some similar program). The data is then analyzed using different data mining tasks as directed by the user. To accomplish this, our tool uses the Java Runtime Environment to invoke the embedded data mining modules provided by the open-source data mining package Weka[1] with appropriate parameters. The results of mining the assessment data are saved and analyzed for errors before being streamlined to remove non-essential information and displayed to the user in a meaningful way. We created a user friendly graphical user interface (GUI) that will allow educators to evaluate their own outcome-based assessment data with minimal computing skills. It provides intuitive and readily understandable controls to allow the selection of the user's data, run the data through the desired data mining task, and display the results of the chosen task in a format easily understood by users with a minimal knowledge of the technical details of the algorithms utilized. In the remainder of this section, we provide a brief description of the specific data mining methods we utilized and describe how we have incorporated each method to streamline our software tool. We will cover three main aspects of our application: data cleaning and preprocessing, extracting relevant outcomes, and identifying dependency of outcomes.

[1] http://www.cs.waikato.ac.nz/ ml/weka

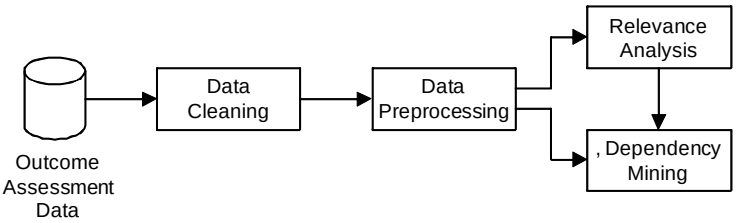


Fig. 1. The data mining workflow for extracting useful knowledge from outcome assessment data.

2.2 Data Cleaning and Preprocessing

In order to perform any data mining task on the assessment data, it is very important to prepare the data by cleaning and preprocessing it. The purpose of the data cleaning step is to remove irrelevant items from the data. This involves transforming data by creating new attributes and reducing the number of attributes. We accomplish this goal in three steps.

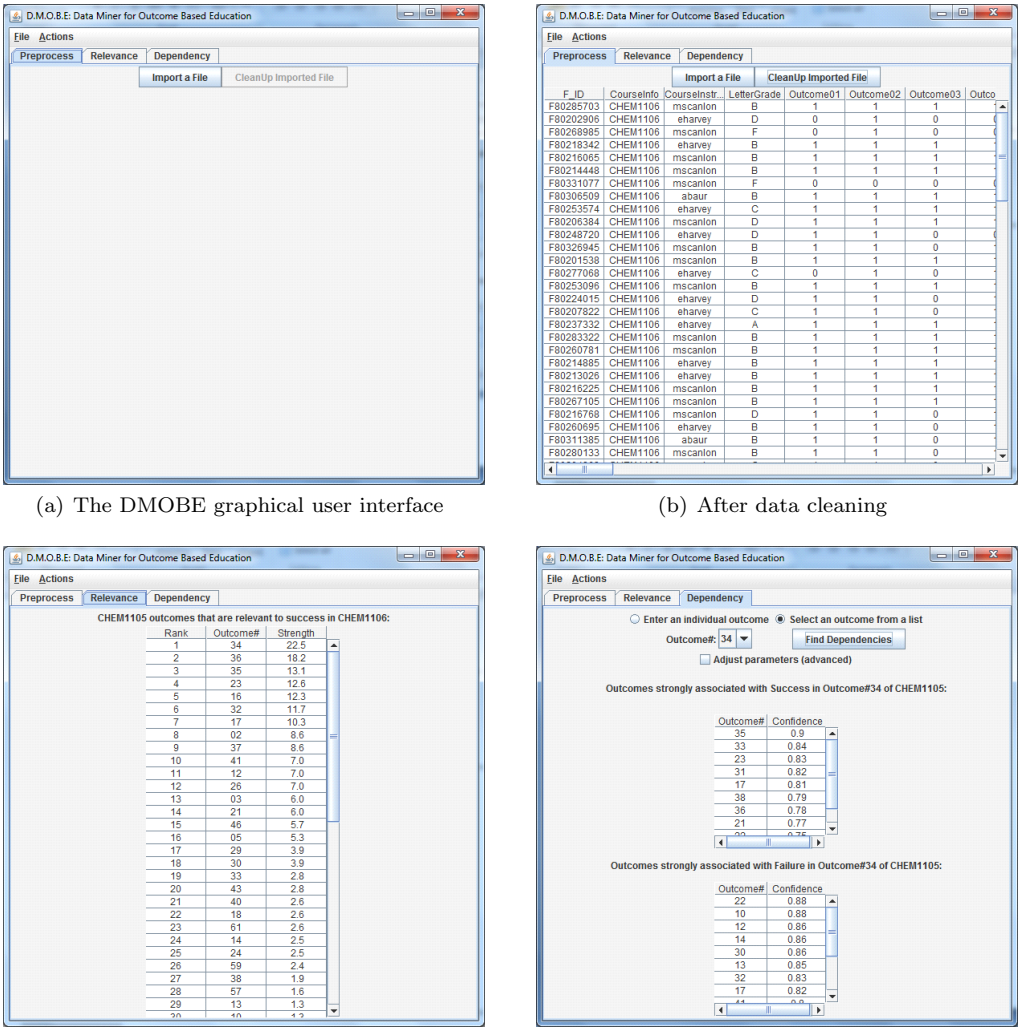
The first step allows the user to select a course assessment data file. This file is presumed to be a .xls, .csv, or a tab-delimited .txt file that can easily be created/exported by popular spreadsheet packages such as Microsoft Excel or OpenOffice. Furthermore, the data must be compiled from one specific course and contain a header row with the following attribute names in order: a student identifier, one or more course attributes (including at least a final letter grade), and a list of several learning outcomes denoted as "Outcome x" or "outcome x". The assessment of a given outcome is presumed to be represented as '1' indicating mastery of the outcome or '0' indicating otherwise. Figure 2(a) shows the initial screenshot of the DMOBE graphical user interface.

The second step is required to "cleanup" data files. In this step, all non-essential characters (if any) are removed that may have been added to the data by the exportation process. Then the data is scanned for any student receiving a grade other than 'A' through 'F'. This limits our dataset to students who have seen the course through to completion. Next, the program fills in all missing outcome results for each student record (if any), converting any numerical entries greater than 1 to 1 and converting blank or non-numerical entries to 0. Finally, to reduce the number of outcomes considered and therefore the time and space requirements of the analysis, all individual outcomes in the file with identical performance by every student are removed. This removes any erroneously entered columns of all 0s and discards outcomes which were mastered by every student as trivial. Figure 2(b) shows the screenshot of the DMOBE graphical user interface after this step is performed on a sample course outcome assessment dataset.

The third step is hidden from user view and creates a ".arff" file, which is the required file type used by the Weka data mining package. This file only contains the student identification and outcomes from the imported file, is created in the same directory as the imported file, and is deleted when the program closes.

2.3 Extracting Relevant Course Outcomes

The goal of this component is to determine those learning outcomes in a given course, mastery level in which influence student performance in the same or a subsequent course. We utilize supervised feature selection for this purpose. In general, the purpose of feature selection in any data mining task is to identify relevant features from a dataset. The goal is to remove noisy or redundant features that make the discovery of meaningful patterns from the data difficult. The general approach in feature selection is to compute a score for each attribute and then to select the attributes with the best scores.



(a) The DMOBE graphical user interface

(b) After data cleaning

(c) After extraction of relevant learning outcomes

(d) After dependency analysis for a given outcome

Fig. 2. Screenshots of DMOBE (Data Miner for Outcomes Based Education).

Let $O = \{O_1, O_2, \ldots, O_m\}$ be the set of m learning outcomes in a course and C be the class label that can be a categorical measure of assessment (e.g., the letter grade) in the same or a subsequent course. Our goal is to find the relevant outcomes, i.e., the outcomes whose mastery level can predict student performance in terms of C. This is accomplished by ordering the original outcomes into $O' = \{O'_1, O'_2, \ldots, O'_m\}$ so that $S(O'_j) \geq S(O'_{j+1})$, where $S(O'_j)$ represents the score of O'_j that measures how O'_j can discriminate among the different values of C. We evaluate each outcome (a feature) independently by computing the Chi-squared (χ^2) statistic with respect to C. The χ^2 value is computed by aggregating the deviation of observed values from expected values and thus a higher value indicates stronger relevance of an outcome with respect to C. If a_i is the number of instances having the i^{th} value of an attribute, c_j is the number of instances having the j^{th} class label, p_{ij} is the number of instances having the i^{th} value of the attribute and j^{th} class label, and n is the total number of instances, then the χ^2 value of an attribute is computed as [Altidor et al. 2011]:

$$\chi^2 = \sum_{i,j} \frac{(p_{ij} - a_i c_j/n)^2}{a_i c_j/n}$$

In our tool, in order to identify the relevant outcomes in a given course, the user also needs to upload and clean the assessment data for a "target" course based on which relevance is determined. The given course and the target course are scanned for matching student ids. Any student found to have not taken the target course is discarded as irrelevant. For a student who has taken both the courses, the grade in the target course is inserted as the class label. A value of '1' is used for students receiving an A or B and '0' otherwise. Then the χ^2 method is applied for computing the relevance of each outcome. The final result is the ordered display of each outcome according to its χ^2 value. The more this value is, the more relevant an outcome is in predicting success in the target course. Figure 2(c) shows the screenshot of the DMOBE graphical user interface after relevant learning outcomes are extracted from a sample course outcome assessment dataset.

2.4 Identifying Dependencies between Course Outcomes

We attempt to give educators insight as to which outcomes within a single course can be strongly associated with other outcomes within that course. This may provide answers to such questions as, "In Chemical Principles I, is the mastery of any particular outcome strongly associated with the mastery of outcome#34?" Our goal is to show the various strong dependencies that may exist between particular outcomes in a course. This will allow instructors to know which key outcomes in the course to emphasize in order to increase students' mastery in a selected target outcome. This component of our tool is based on the framework of association mining.

The goal of association mining is to derive correlations among multiple features of a dataset [Agrawal et al. 1993]. An association rule is an implication of the form $X \Rightarrow Y_{[Supp,Conf]}$, where X and Y are disjoint itemsets, $Supp$ is the *support* of $X \cup Y$ indicating the percentage of total records that contain both X and Y, and $Conf$ is the *confidence* of the rule that is defined as $Supp(X \cup Y)/Supp(X)$. The intuitive meaning of such a rule is that records of the dataset that contain X tend to contain Y. A typical example of an association rule from the outcome based education domain can be $O_i \Rightarrow O_{j[.1,.8]}$. This implies 80% of the time when students perform well on O_i, they will also perform well on O_j and 10% of the student records have students performing well on O_i and O_j together. Here the confidence of the rule is 80% and the support of the rule is 10% .

Association mining can reveal the fact that a group of students performing well on a set of outcomes also perform well on another set of outcomes. This type of mining can be applied at two levels; in a single course or across multiple courses. Let us assume that a course has several learning outcomes: $\{O_1, O_2, \ldots, O_m\}$. If association mining reveals that students performing well on O_i also perform well on O_j and students performing poorly on O_i also perform poorly on O_j, then it may be reasonable to reorganize the course materials so as to cover O_i before O_j. For different courses, if an outcome O_i shows strong association to an outcome O_k in a subsequent course, then the instructor may put emphasis on explaining concepts related to O_i in the former course.

The goal in a particular application is to find all association rules that satisfy user-specified minimum support and minimum confidence constraints. Association rules are generated in two steps. The itemsets having minimum support (called *large itemsets*) are discovered first and then these large itemsets are used to generate the association rules with minimum confidence. The *Apriori* association mining algorithm [Agrawal and Swami 1994] has widely been accepted as the algorithm of choice in many applications. The process of

generating large itemsets in Apriori consists of several passes and the large itemsets found in one pass are used to generate large itemsets for the next pass. In the k^{th} pass, the candidate itemsets of length k (C_k) are generated by joining large itemsets of length $k-1$ (L_{k-1}) and leaving out itemsets containing any non-large subset. Formally, $L_{k-1} * L_{k-1} = \{X \cup Y | X, Y \in L_{k-1}, |X \cap Y| = k-2\}$. All candidate k-itemsets having support values greater than the minimum support threshold constitute the large k-itemsets L_k. Formally, $L_k = \{X | X \in C_k, Supp(X) \geq Supp_{min}\}$. After all the large itemsets are generated, for every large itemset L, the following set of rules are generated: $\{A \Rightarrow (L - A) \mid A \subset L, A \neq \emptyset, Supp(L)/Supp(A) \geq Conf_{min}\}$.

The traditional notion of association rule mining is based on the presence of items in the datasets, i.e., the focus is on discovering only positive rules of the form $X \Rightarrow Y$. The capture of negative rules of the form $\neg X \Rightarrow \neg Y$ is not supported. But negative rules can be important in an application. For example, in the outcome based education domain, if $O_i \Rightarrow O_j$ reveals that students performing well on O_i also perform well on O_j, then $\neg O_i \Rightarrow \neg O_j$ will reveal that students performing poorly on O_i also perform poorly on O_j. The notion of negative association rules was introduced in Silverstein et al. [1998] where both the presence and the absence of items are considered for generating rules. Also, in the traditional framework, rules are generated to discover the association between different possible combinations of items. But in the outcome based education domain, it is important to identify which outcomes may influence a given outcome, i.e, to identify the outcomes that success in a given outcome may depend on. In such a case, it becomes important to specify a target outcome and extract only rules that have a single outcome on the left and a single outcome (the target outcome) on the right.

To address the above issues, we define a modified rule mining framework for our application. We call this *target dependency mining* and we define two types of dependencies–*positive dependency* and *negative dependency*. Let $O = \{O_1, O_2, \ldots, O_m\}$ be a set of m learning outcomes and $O_j \in O$ be a target outcome. Then we define a *dependency rule* of O_j as an implication of the form $O_i \Rightarrow O_j$, where $Supp(O_i \cup O_j) \geq Supp_{min}$ and $Supp(O_i \cup O_j)/Supp(O_i) \geq Conf_{min}$. We also represent positive and negative dependencies by defining the *positive determinants* and *negative determinants* of an outcome. The positive determinants of O_j are defined as $\mathcal{D}_j^+ = \{O_i \mid O_i \Rightarrow O_j\}$, i.e., the set of all outcomes that individually imply O_j by satisfying the minimum support and minimum confidence constraints. Inversely, the negative determinants of O_j are defined as $\mathcal{D}_j^- = \{O_i \mid \neg O_i \Rightarrow \neg O_j\}$, i.e., the set of all outcomes whose false values individually imply the false value of O_j by satisfying the minimum support and minimum confidence constraints. Our modified framework not only makes the rules easier to interpret, but also substantially reduces the computational time required to generate the rules by bypassing most of the large itemsets generation.

DMOBE provides two options of selecting a target outcome before a user can discover the dependencies. The user can select "any" available outcome. This option can be used without any prior extraction of relevant outcomes. Alternatively, after a set of relevant outcomes have being extracted, an outcome may be selected from the resulting list of ranked outcomes. After a target outcome is selected, all the positive and negative dependencies are extracted by executing dependency mining. Note that the dependency mining has been implemented by modifying Weka's Apriori implementation. We used a default minimum support of 0.2 and a default minimum confidence of 0.75, but the user has the option of adjusting these parameters and specifying the maximum number of rules to be generated. Once the rules are generated, the results are meaningfully displayed in a table along with the respective confidence values. Figure 2(d) shows the screenshot of the DMOBE graphical user interface after the dependency analysis is performed for a given relevant outcome.

3. EXPERIMENTS

3.1 Datasets

We evaluated our work using data provided by the Department of Chemistry at Fairmont State University, which uses mastery-based course assessment in their lower-level courses. These courses are CHEM 1105 (Chemical Principles I), CHEM 1106 (Chemical Principles II), CHEM 2201 (Organic Chemistry I), and CHEM 2202 (Organic Chemistry II). These courses are taken in this sequence. Each course has approximately sixty outcomes. Each student is assigned a "mastery level" in each outcome and the final letter grade is assigned based of the percent of outcomes mastered. We used the outcome mastery data from 2004 to 2007 for evaluation purpose. This data provides us with a strong sample set with which to evaluate our methods while also giving the chemistry department immediately usable assessment information.

3.2 Analysis of Results

We found some very helpful information in the list of course outcomes that are relevant to success in a subsequent course. Figure 2(c) shows the ranked outcomes in CHEM 1105 that are relevant to success in CHEM 1106. The top three relevant outcomes are three difficult outcomes from near the end of CHEM 1105 that build on several underlying principles and skills. Two of these are quantitative, multi-step stoichiometry outcomes that require the synthesis of several previously learned skills. One of these, outcome#35 (multi-step stoichiometry) includes extraneous information and requires students to apply knowledge from seven previous outcomes. Another relevant outcome in CHEM 1105 with respect to success in CHEM 1106 is outcome#36 that requires the students to be able to "list and rank intermolecular forces". This sounds like a simple outcome but in fact it requires that students count valence electrons, draw a Lewis structure, predict a molecular shape using the valence shell electron pair repulsion model, evaluate and draw bond dipoles, pictorially or vectorially add the bond dipoles to get a net dipole for the molecule, and draw the net dipole. Then they are asked to do the same thing for three more compounds, in order to predict and rank the intermolecular forces for each of those compounds! The strong predictive nature of outcome#36 called our attention to the complexity of this outcome, and the outcome was subsequently broken down into smaller chunks to support student learning.

The next three outcomes in CHEM 1105 that are relevant to success in CHEM 1106, outcomes #23, #16, and #32, are all based on math skills, including graphical interpretation. Outcome#16 requires that students solve an algebraic expression involving logs or exponents, and explain each step. Outcome#23 requires students to extract information from linear regression parameters; they are given a graph with linear regression parameters and an equation to which they can match the graph in order to extract information. Outcome#32 requires that students interpret a spectrum using information in a provided data table. They match observed peaks to information provided in a data table, then match the assigned peaks in the spectrum to functional groups in a selection of molecules.

The data set for which analyses are provided in this paper includes the years 2004-2007. We are looking forward to comparing the patterns discussed here to patterns that emerge using data from the subsequent years. We have made some changes each year to the teaching methods and in some cases to the assessments themselves, to improve student learning based on a simpler analysis of percent class mastery of various outcomes. As a result, we would expect some of the relevance patterns to change as more students become successful on certain outcomes. Examples of this might be outcomes #36 and #38 in CHEM 1105.

4. CONCLUSION

With outcome based learning paradigm increasingly being adopted in higher education, the importance of maintaining outcome assessment data is on the rise. These knowledge rich datasets can be extremely beneficial in improving the quality of eduction. But a meaningful analysis of these ever-growing datasets presents a challenge to the educators. We have developed a software tool that allows educators to easily apply data mining techniques to analyze several key aspects of their pedagogy. Using this tool, an instructor can (a) import his own course assessment data, (b) extract useful, meaningful, and otherwise unattainable information by applying sophisticated data mining techniques, and (c) view simplified, readily understood results through a flexible graphical user interface. Specifically, this tool allows an educator to discover which outcomes of a course are relevant to success in a subsequent course and which outcomes within a course strongly influence the mastery of a given outcome. We used supervised feature selection to extract relevant outcomes. To discover the dependency of a given outcome on other outcomes, we modified the traditional framework of association rule mining which we refer to as "dependency mining". We believe that this tool can help improve the overall quality of education in a number of ways. It will enable the streamlining of curricula and allow for the early identification of "at risk" students. Furthermore, this tool will provide educators a way of identifying a course's key learning outcomes and a means of empirically evaluating the relationships between outcomes.

REFERENCES

Agrawal, R., Imielinski, T., and Swami, A. 1993. Mining association rules between sets of items in large databases. In *Proceedings of ACM SIGMOD International Conference on Management of Data, Washington, DC, May 26-28*. 207–216.

Agrawal, R. and Swami, A. 1994. Fast algorithms for mining association rules. In *Proceedings of 20th international conference on very large databases, Santiago, Chile, September 12-15*. 487–499.

Altidor, W., Khoshgoftaar, T. M., and Hulse, J. V. 2011. Robustness of filter-based feature ranking: A case study. In *Proceedings of 24th Florida Artificial Intelligence Research Society Conference (FLAIRS-24), Palm Beach, FL, May 18-20, 2011*. 453–458.

Chen, G., Liu, C., Ou, K., and Liu, B. 2000. Discovering decision knowledge from web log portfolio for managing classroom processes by applying decision tree and data cube technology. *Journal of Educational Computing Research 23,* 3, 305–332.

Despotovic, M., Bogdanovic, Z., Barac, D., and Radenkovic, B. 2008. An application of data mining in adaptive web based education system. In *Proceedings of Web-Based Education, Innsbruck, Austria, March 17-19*. 182–185.

Minaei-Bidgoli, B., Kashy, D., Kortmeyer, G., and Punch, W. 2003. Predicting student performance: An application of data mining methods with an educational web-based system. In *Proceedings of Frontiers in Education Conference, Boulder, CO, Nov 5-8*. 13–18.

Minaei-Bidgoli, B., Kortmeyer, G., and Punch, W. 2004. Enhancing online learning performance: An application of data mining methods. In *Proceedings of International Conference on Computers and Advanced Technology in Education, Kauai, HI, Aug 1618*. 173–178.

Minaei-Bidgoli, B. and Punch, W. 2003. Using genetic algorithms for data mining optimization in an educational web-based system. In *Proceedings of Genetic and Evolutionary Computation Conference, Chicago, IL, Jul 12-16*. 2252–2263.

Romero, C. and García, S. V. E. 2008. Data mining in course management systems: Moodle case study and tutorial. *Computers & Education 51,* 1, 368–384.

Romero, C., Ventura, S., Espejo, P., and Hervas, C. 2008. Data mining algorithms to classify students. In *Proceedings of International Conference on Educational Data Mining Conference, Montreal, Canada, Jun 20-21*. 182–185.

Silverstein, C., Brin, S., and Motwani, R. 1998. Beyond market baskets: Generalizing association rules to dependence rules. *Data Mining and Knowledge Discovery 2,* 1, 39–68.

William, S. 1994. *Outcomes Based Education: Critical Issues and Answers*. American Association of School Administrators: Arlington, VA.

Yudelson, M. V., Medvedeva, O., Legowski, E., Castine, M., Jukic, D., and Crowley, R. S. 2006. Mining student learning data to develop high level pedagogic strategy in a medical its. In *AAAI Workshop on Educational Data Mining, Boston, MA, July 16-17*. 82–89.