The Academic Analytics Tool: Workflow and Use Cases*

Tamra Ross[1], Ting-Wen Chang[2], Cindy Ives[1], Nancy Parker[1], Andrew Han[1] and Sabine Graf[1]

[1] Athabasca University, Canada
{tross, cindyi; nancyp; andrewh; sabineg}@athabascau.ca
[2] Beijing Normal University, China
tingwenchang@bnu.edu.cn

Abstract. To meet the demand for timely analysis and revision of online courses, educators need ongoing, unfettered access to data about how students interact with courses and online resources. Currently available tools for exploring student data provide some important insights, but are typically focused on automated data mining, visualizations, or displaying pre-set reports. These tools also often require either high technical skills and/or installation of specialized software, making them inaccessible to most educators and learning designers. In this paper, we introduce the Academic Analytics Tool (AAT) and provide some hands-on examples on how the tool can be used. AAT is designed to allow people (e.g., educators, learning designers, etc.) without technical expertise to extract and analyse data from learning management systems (LMSs). AAT offers high usability and permits full exploration of LMSs' data on any computer with internet access to foster responsive analysis and improvement of online courses.

Keywords: academic analytics · data extraction and analysis · online learning

1 Introduction

Online learning is still a rather new educational option, and there is much to be learned about the best teaching methods and course designs for this format. The multi-year course revision process is simply not conducive to meeting the evolving demands of online students, or rapid changes in the online educational marketplace. To ease the burden on IT departments and ensure courses are monitored and revised frequently and appropriately, educators and learning designers should be empowered with direct access to data about student behaviour in online courses [1].

Learning management systems (LMSs) store vast quantities of data about student activities in their courses, including forum activities, access of online books and resources, grades on quizzes and exams, assignment submissions, and communications with instructors [2]. By analysing this information, educators can learn a great deal about what students are doing in their courses, and what factors affect student success.

* The authors acknowledge the support of Athabasca University and NSERC.

A number of tools exist to extract student behaviour data, but these typically come with limitations that make them difficult for educators to use or they limit the data educators are able to investigate. There is a need for tools designed for educators that allow for a full range of queries on all available data in an LMS, and that work with a wide range of LMSs and database formats [3,4].

The Academic Analytics Tool (AAT) [5] is a software tool designed to allow educators, learning designers and school administrators to perform their own investigations to gain a better understanding of how students interact with online course materials and resources. It is designed specifically for people who do not have experience with database systems or analytical software, and runs on any computer without additional software because it is browser-based.

AAT is different from other tools because it provides no pre-set reports, and does not perform any automated data discovery. Instead, it supports users in their own investigations into any data available in a LMS, using a wizard style interface that can be used by users without programming, analytical or database skills. The resulting reports can be output in a variety of formats to be used in other analysis tools (e.g., statistical tools, advanced visualization tools, etc.). AAT also facilitates sharing amongst users by allowing them to save their projects and results as "public" so they are available to others. Thus, AAT empowers educators and learning designers to better understand what is going on in their courses, responsively revise courses, and monitor the impact of changes to course designs and teaching methodologies.

While AAT can help answer a broad range of questions, a few examples are:

- How many students are using a given resource? Is its use correlated with better performance on exams?
- Are certain types of resources and activities (e.g., quizzes, forums) more helpful for students in one course or faculty than in another?
- How does performance in a junior level course correlate to performance in advanced courses?
- Do students who complete optional quizzes score better on the final exam?
- When teachers are more active in discussion forums, does such behaviour impact students' overall grades?
- When teachers share the best solution of an assignment with the class, does such behaviour impact students' performance on subsequent assignments?

The rest of the paper is organized as follows: section 2 describes how AAT works and section 3 provides a few hands-on examples. Section 4 includes our conclusions and goals for future work.

2 How AAT Works

Every investigation in AAT begins with a *Project*. The rest of the terminology is similarly in plain English, designed to accommodate users without technical backgrounds. For each *Project*, AAT steps users through the process of building a query on the

LMS database using a wizard-like interface. In the following, the basic process/workflow for all AAT investigations is described:

Step 1. Name and save a new *Project*: the *Project* acts as a container to store the selections for a given investigation.

Step 2. Select an LMS: an educational institution may use more than one LMS, but each *Project* is only retrieving and analysing data from one LMS.

Step 3. Select a *DataSet*: the *DataSet* consists of all of the courses that should be included in an investigation. The *DataSet* can consist of one course, several, or all courses. In this step, either a predefined *DataSet* can be selected or a new one can be built from all or any combination of available courses in the selected LMS.

Step 4. Build a *Pattern*: a *Pattern* specifies what should be investigated in terms of *Concepts* (i.e., students, quizzes, forums, etc.) and *Attributes* (i.e., student id, quiz grades, forum messages, etc.). In addition, limits can be added to use a variety of filters (e.g., include only students with quiz grades lower than 70%, etc.). A *Project* can have many *Patterns*. Therefore, many investigations can be performed on the same set of courses.

Step 5. Optionally, additional actions can be performed on a saved *Pattern*, such as *Cloning* (making a copy of a *Pattern* to edit or share), *Chaining* (linking two *Patterns* together to expand the results) and *Analysis* (performing calculations such as average, sum, count, min and max on a result set of a *Pattern* to analyse data in more detail).

Step 6. View the results of a *Pattern*, or export them in HTML, XML, or CSV format for use in other analysis tools, or for sharing with others.

3 Use Cases

In this section, three simple hands-on examples / use cases are provided to demonstrate AAT's functions and how it works.

The first example shows how a user can find the overall number of forum posts, posted by students during a given timespan (i.e., between March 1 and March 31, 2016) in a given course (i.e., "COMP101").

In order to answer this query, a user first needs to create a new *Project*, select the respective LMS and create a new *DataSet* with the respective course(s) to be investigated (e.g., "COMP101"). In the next step, a new *Pattern* needs to be built, selecting "Student" and "Post" as *Concepts*. Then, the *Attributes* are selected as "forum post id" and "time forum post created" from the POST grouping and "first name", "last name" and "user id" from the USER grouping. To limit the result set to only postings between March 1 and March 31, 2016, a limit on "time forum post created" can be set to be between "March 1, 2016" and "March 31, 2016". Afterwards, the *Pattern* is named and saved. In the next step, to count the number of postings, another *Pattern* is built using the "Analyse" button. The *Pattern* just created is selected from the list of available options as basis for the *Analysis*. Then, an *Analysis* with one value as result is selected and the COUNT of "forum post id" is chosen. To see the result (a single value representing the overall number of forum posts posted by students during the given timespan and in the given course), the *Analysis* can be executed by clicking on

the "Perform Analysis" button. As a last step, the *Analysis* should be named and saved to access it at a later point.

The second example illustrates how to find the number of forum posts from students and teachers in a set of courses (i.e., all 1-level COMP courses including COMP101, COMP102, COMP103, COMP104 and COMP105). As output a list with each person's first and last name, user id, and the number of postings of that person is expected.

To answer this question, again, the first step is to create a new *Project*, select the respective LMS and create a new *DataSet* with the respective courses to be investigated (i.e., all 1-level COMP courses). In the next step, a new *Pattern* is built by selecting "Student", "Teacher", and "Post" as *Concepts*. Then, *Attributes* are selected to be "forum post id" from the POST grouping and "first name", "last name" and "user id" from the USER grouping. No limits are needed for this example. To finalize the *Pattern*, it needs to be named and saved. As a next step, to count the number of postings, another *Pattern* is built using the "Analyse" button. The *Pattern* just created is chosen from the list of available options as basis for the *Analysis* and an *Analysis* with a column as result is selected. Next, the COUNT of "forum post id" for every "User" is selected to retrieve results on how many posts each user posted. To see the result (depicted in Figure 1[*]), the *Analysis* can be executed through the "Perform Analysis" button. To access the results at a later time, the *Analysis* should be named and saved. To export the results, the *Analysis* pattern just created can be selected, then details are shown about the pattern and when clicking on "Run Project" the output and export options are shown.

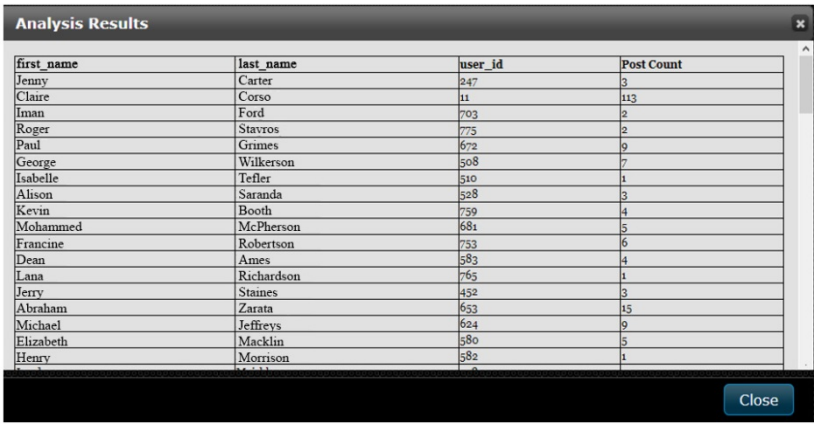

Fig. 1. Output of use case 2 – A list of users and their number of forum posts.

The third example aims at finding the average grade on an assignment (i.e., Assignment 1) in different revisions of a course (i.e., COMP201 Revision 1, COMP201 Revision 2, COMP201 Revision 3, COMP201 Revision 4). As output, a list with each course's name and the average grade on Assignment 1 is expected.

[*] The figure shows simulated data rather than real student/course data

To answer this query, again, a new *Project* is created, the respective LMS is selected and a new *DataSet* with the respective courses is chosen (i.e., all versions of COMP201). In the next step, a new *Pattern* is built by selecting "Course", "Student" and "Assignment" as the *Concepts*. Then, the *Attributes* are selected as "assignment name" and "assignment submissions grade" from the ASSIGNMENT grouping, and "course name" from the COURSE grouping. To include only Assignment 1 in the investigation, a limit is set on "assignment name" to be equal "Assignment 1". Next, the *Pattern* is named and saved. To calculate the average of the assignment grades, another *Pattern* is built using the "Analyse" button. The *Pattern* just created is chosen from the list of available options as basis for the *Analysis* and an *Analysis* is created with a column as result. In the *Analysis*, the AVERAGE of "assignment submission grade" for every "Course" is selected to show the average grade in each course. To see the result of the analysis (a list of courses with each course's average grade), the *Analysis* can be executed through the "Perform Analysis" button. To access the results at a later time, the *Analysis* should be named and saved. To export the results, the *Analysis* pattern just created can be selected, then details are shown about the pattern and when clicking on "Run Project" the output and export options are shown.

4 Conclusions

This paper presents the Academic Analytics Tool (AAT) and some hands-on examples on how AAT works. AAT empowers educators and learning designers to directly access data from learning management systems about how students interact with their courses, so they can analyse educational outcomes and the impact of changes to courses. Allowing educators and learning designers to conduct their own investigations increases opportunities to monitor the effectiveness of online courses and inform the development of course revisions and new resources. Future work will continue to increase usability for users without computer science skills and offer advanced analytical functions.

References

[1] Wayman, J.C.: Involving Teachers in Data-Driven Decision Making: Using Computer Data Systems to Support Teacher Inquiry and Reflection. Journal of Education for Students Placed at Risk, 10(3), 295-308 (2009)

[2] Baker, R. S. J. D.: Data Mining for Education. International Encyclopedia of Education (3rd ed.). Elsevier, Oxford, UK (2010)

[3] Merceron, A.: Analyzing Users Data Captured in Learning Management Systems. In Proceedings of the Workshop on Data Analysis and Interpretation for Learning Environments. Villard-de-Lans, France (2013)

[4] Fayyad, U., Piatetsky-Shapiro, G., Smyth, P.: From Data Mining to Knowledge Discovery in Databases. AI Magazine, 17(3), 37-54 (1996)

[5] Graf, S., Ives, C., Rahman, N., Ferri, A.: AAT – A Tool for Accessing and Analysing Students' Behaviour Data in Learning Systems. In Proceedings of the International Conference on Learning Analytics and Knowledge, ACM, 174-179 (2011)