In Proceedings of the Intelligent Tutoring Systems Conference (ITS 2010)

Computational Workflows for Assessing Student Learning

Jun Ma, Erin Shaw, Jihie Kim,

Information Sciences Institute, University of Southern California
4676 Admiralty Way, Marina del Rey CA 90292, United States
{junma, jihie, shaw}@isi.edu

Abstract. The use of technology for instruction, and the enormous amount of information available for consumption, places a considerable burden on instructors who must learn to integrate appropriate student practices and learning assessment. The Pedagogical Workflows project is developing a novel workflow environment that supports efficient assessment of student learning through interactive generation and execution of various assessment workflows. We focus especially on how student discussion use can be combined with more traditional assessment data. In this paper, we present our initial assessment workflows, the initial feedback from instructors, and the user portal that is being developed for running the workflows. Inherent in the development of the workflows is an examination of what teachers think is important to learn about their students, a question that is central to every intelligent tutoring system. We anticipate that assessment workflows will become an important tool for instructors, researchers, and ITS development.

Keywords: Discourse analysis, workflow technology, student learning assessment.

1 Introduction

Educational technology for online learning is now centrally supported by many colleges and universities. The perceived mandate to use technology for instruction, in addition to the enormous amount of information available for consumption on the Web, places a considerable burden on instructors who must learn to integrate appropriate student practices and learning assessment via the new media. Discussion boards, for example, have become an essential tool for student-student and student-instructor communication beyond the walls of the classroom; however, properly integrating participation results and traditional assessments is very much a challenge.

Workflow technology has been successfully applied to scientific applications [1, 2]. Existing workflow generation and execution approaches can be useful for making educational assessment tools more accessible to instructors, and for making large scale assessment, requiring large amounts of course data, feasible. Workflow results can be used to answer questions and provide formative feedback to instructors to facilitate "just in time" instructional adaptation to students learning and needs. Our

workflow project modularizes the selection and application of both traditional and non-traditional assessment techniques in online instruction. Instructors can use workflows to perform a variety of correlations, including those that require privacy protected data such as grade information. For example, an instructor may wish to analyze student discussion participation data, the grades of a student or entire class, and additional data from earlier semesters, to answer questions such as 'What type of discussion board use correlates to better performance?', 'How long do students wait before classmates respond?', 'How do online activities this semester differ from those of previous semesters?' These questions guide instructional improvement and are traditionally difficult to answer when focused on online environments.

The goal of our project is to create a novel workflow environment that supports efficient assessment of student learning through interactive generation and execution of various assessment workflows. With respect to online activities, we focus on qualitative methods for evaluating learning by discourse [3, 4]. These methods use information retrieval and natural language processing (NLP) techniques to analyze the impact of discussion board participation on conceptual understanding and communication, both forms of cognitive assessment that inform learning. In addition, we are taking into account longitudinal student changes by electronically tracking students' learning performance across courses as they matriculate to their degree [5].

In this paper, we present our initial assessment workflows that are developed based on the Wings/Pegasus workflow system [6, 2] and initial feedback we received from two Computer Science instructors. We also show the PedWorkflow portal which was developed for the instructors to support easy access.

1.1 Assessment questions and supporting workflows

Table 1**.** Assessment categories and descriptions.

Category	Workflow Description
Analysis of online activities	Composition of discussion data processing/ Classification steps
Correlation between online activities & performance	Composition of discussion data processing steps, student profiles, and correlation analysis
Correlation between online activities & self-assessment	Composition of self-assessment survey, student activity profiles, and correlation analysis
Student profiling	Composition of student information and discussion data processing/ classification steps
Discussion profiling	Composition of discussion data processing/classification steps and relation analysis
Trend analysis	Splitting of discussion data and iterative analysis
Group comparison	Composition of discussion data processing/ classification steps, student profiles and relation analysis

Table 1 illustrates the assessment question categories that we have compiled so far and Table 2 gives examples of questions from these categories. Some of these questions come from interactions with engineering instructors who use our enhanced online discussion board. Other questions are from research in learning assessment [7,

8]. Most questions require the execution of some combination of multiple workflow steps, where often concurrent steps are desirable. Some of the jobs that perform Trend Analysis, which use data sets from different time frames, or Correlation Analyses, which analyze data from multiple semesters, may need parallel execution to improve efficiency. Some of the workflows or sub-workflows may be re-used for similar questions. Provenance of the workflow creation and execution will help the instructor keep track of student activity details and understand how the results are produced.

Table 2. Assessment questions and their priority ratings by two instructors.

Priority Rating		Question
L	M	Which topics have been discussed in the last three semesters?
H	H	Which topics do students ask the most questions about?
H	M	Were all of the questions about topic x answered?
L	H	Which questions were unanswered?
L	M	Do students who participate more often receive better grades?
H	M	Do gender and politeness affect participation?
H		Do more motivated students perform better?
H		Do more confident students participate more?
H	M	Who are the mentors for topic x?
H	H	Which students are confused about topic x?
H	H	Is a student a mentor or help seeker?
H	L	What are his/her strengths?
H	L	Were there similar questions or answers in previous semesters?
H	M	How long did students have to wait for an answer?
H	M	How has student participation changed over time?
H	L	How are online activities in this semester different from previous semesters?

Two instructors, who are working with us to develop and evaluate assessment workflows, were asked to rate the priority of the assessment questions in Table 2 as part of a formative assessment. Their ratings are shown on the left (L=low priority, M=medium, H=high). Both instructors teach multiple undergraduate courses and both have been teaching for at least 10 years. Each had very different concerns. The first instructor (leftmost rating) actively participates in discussions and thus gleans information about topics discussion and questions answered directly, and so rated these types of assessment questions as low priority. Where as the second instructor rated them as high or medium. The second instructor was less focused on correlating personal traits and more focused on identifying students who were behind or who might deserve extra credit, as well as flagging topics that students ask many questions about. The second instructor uses his course teaching assistant to monitor discussions but said he would prefer an objective measure.

Teachers were also asked to contribute assessment questions they might use for their own classes. Both instructors mentioned the importance of having objective measures that were linked to class goals. One instructor desired concrete evidence that the non-computer science students for whom his class was required did not have the pre-requisite skills to succeed in his class. In one of the classes taught, the assignments were team projects and the instructor wished to assess individual students

relative to their teammates. Identifying patterns of student performance relative to student activity was also discussed. The results indicate a need for facilitating measurement for many types of assessment.

2 Workflow Portal for Learning Assessment

2.1 Background: Wings/ Pegasus workflow system

Wings takes the user's workflow requirements and generates a high-level workflow for execution [2]. Pegasus generates executable workflows by assigning execution resources to the computations in the workflow [1]. Pegasus also reduces the workflow execution time by eliminating unnecessary computations whose results already exist and can be reused, and reorganizing the structure of the workflow to minimize job queuing time and data movements. It then submits the workflows to the grid for execution and monitors their status.

There have been intelligent interfaces developed for supporting end-user composition of workflows [9, 10]. The approach exploits knowledge-rich descriptions of the individual components and their constraints in order to validate the composition, and uses artificial intelligence planning techniques in order to systematically verify formal properties of valid workflows. The system analyzes partial workflows created or modified by the user, determines whether they are consistent with the background knowledge that the system has, notifies the user of issues to be resolved in the current workflow, and suggests to the user what actions could be taken to correct those issues. To represent and reason about datasets, workflow components, and workflows, OWL W3C's standard (OWL 2010) [11] and Jena (Jena 2010) [12] were used.

2.2 The PedWorkflow Portal

The Wings/Pegasus workflow system is a powerful tool for semantic workflow generation and execution [2]. Its portal services ease the use of the workflow system by end users. Based on them, we created the PedWorkflow Portal for instructors. The instructors do not need to write any workflow template nor any programming codes to use the workflow system.

The PedWorkflow Portal contains two main parts: the first one is the workflow generation and execution that enables the instructors to select and run a specific assessment workflow with data of his/her interest. And the second one is its presentation of workflow runs that enables the instructors to access and understand the results using the web interface.

When an instructor logs onto the PedWorkflow Portal, he can view the assessment questions that are supported by the portal. For each question, there is a corresponding workflow template in the workflow template library. Once the specific question (i.e. workflow template) has been selected, he can select/upload the inputs for the workflow. The input can be files, database entries or parameters. Once the inputs of the workflow template have been identified, the PedWorkflow Portal binds the data

with the workflow template and uses the Wings engine to instantiate the workflow template with the inputs. The portal then executes the workflow instance. During the execution, each step run is recorded and the instructor can monitor the generation and running of the workflow from the portal interface. When the workflow execution is finished, the instructor can view all the output data and intermediate data using the portal in the result page. The procedure is illustrated in Figure 1: the left window shows selection of assessment question (workflow template) and dataset binding. Once the instructor click run, then the top right window pop up showing the workflow generation and execution detail. In viewing the details, there are links to all data corresponding to this run of the workflow, including the final output graphs, which are shown in the bottom right window.

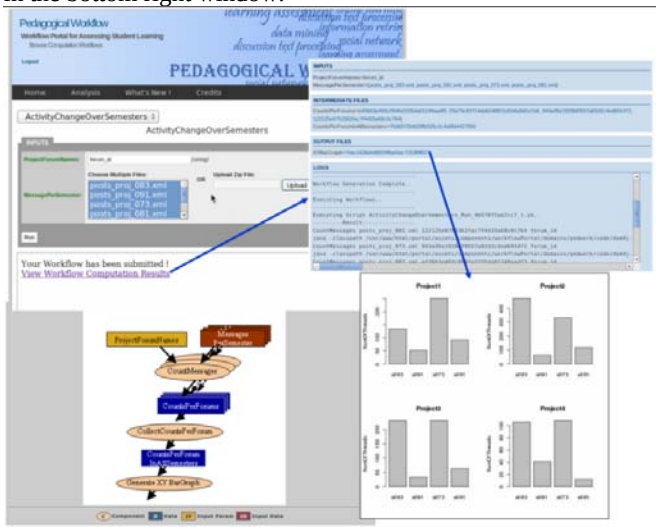


Fig. 1. The PedWorkflow Portal

3 Learning Assessment Workflows

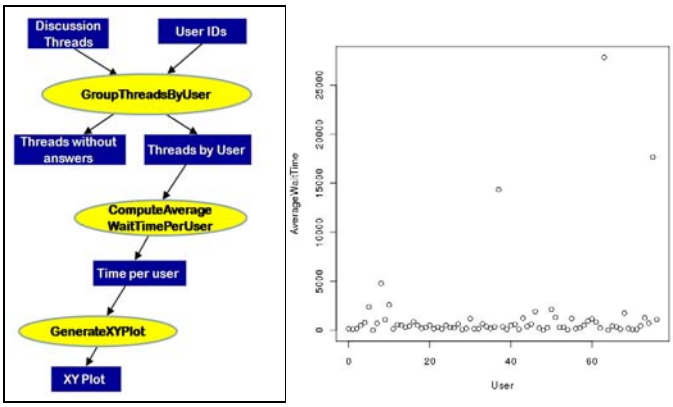
This section describes several categories of assessment workflows that we have developed so far: 1) collective analysis of online student activities that covers all the students in the class; 2) correlation analysis with multiple measures: student online activity vs. performance; and 3) comparative analysis of data from multiple classes. We illustrate how our workflow framework supports instructional assessment.

3.1 Analysis of Online Activities

This type of analysis collects data from all the participating students, and combines the dataset with respect to a certain measure, to provide insight on student collective behaviors. The following is an example query:

How long did students have to wait for an answer?

Each discussion thread consists of question and answer exchanges among students, instructors and teach assistants. Thus the average time a student needs to wait for a question is of great importance to evaluate the effectiveness of the online discussion board.



(a) (b)

Fig. 2**.** The workflow diagram for StudentWaitTimeAnalysis and an output example.

The computation of analyzing wait time can be split into several steps. Each computation step is a workflow component in component library. In this example, first we need to have all discussion thread data and all student profile data as inputs (DiscussionThreads and StudentIDs in the workflow diagram Fig 2(a)). Then we use a grouping component GroupThreadsByUser to group the threads by users. Moreover, we locate each user's participation in different discussion threads, and compute the difference between the initial post (i.e. the initial question by a student) and the following response to the question. The output file ThreadsByUser for this grouping component is an xml file recording the all the sessions' waiting time of each user. We also export all the threads that are not replied with an answer as the output file ThreadsWithoutAnswers.

In the second step, we pick a general average computing component called ComputeAverageWaitTimePerUser component and calculate the average wait time for each student. The output of this component is a raw data table TimePerUser which indicates the average wait time for each user. And this output can be further fed into general plot components for result visualization, like the GenerateXYPlot component in this example, which is actually an R function in component library that plot, the input 2D data table into an image figure file XYPlot.

Once we have chosen a component from the component library to build up our workflow, we import it into workflow and link the corresponding input/output files to connect different components, as shown in Fig 2(a). The result is called workflow template, which is represented by OWL. We cannot directly run workflow templates since they are not fully instantiated yet. An executable workflow needs bindings of input/output files and parameters. In case we need multiple runs of the same component, the system needs to know how many runs are needed for each component based on the input file bindings. The PedWorkflow Portal system has an interface to

generate the instance of workflow templates dynamically. The interface enables the user to upload/select the input files and parameters, and then generate the corresponding workflow instance. Finally, the workflow is executed by either local PedWorkflow Portal or the Wings/Pegasus grid system. The result is returned to Portal once the running of the workflow instance is finished.

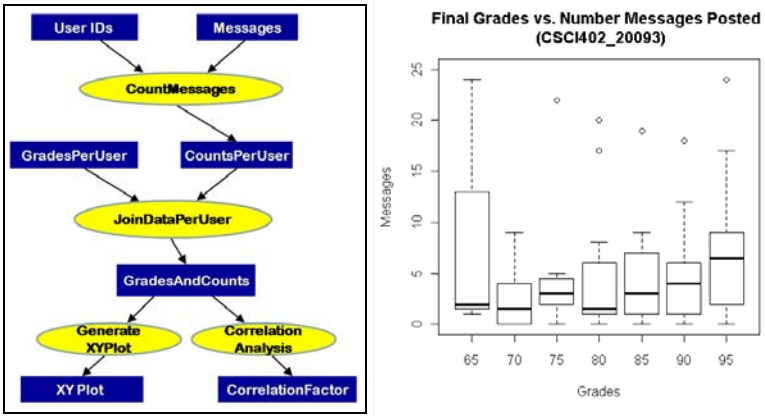
Figure 2(b) shows the output of the StudentWaitTimeAnalysis workflow with the input discussion threads of CSCI402 Fall2007 at USC. We can see from the graph that the discussion response is highly active: most threads can be replied within 24 hours (1440 minutes). We can also see from the result that there are many students participating in the online discussion: 75 online users plotted in the graph, among 120 students enrolled in the class.

3.2 Correlation Between Online Activities and Performance

Instructors may be interested in many types of correlation analyses. Here is an example question:

Do students who participate more often receive better grades?

In this case, we want to find the relationship between the student grades and the student activity in online discussion. Instructors want to know whether students who participate more often receive higher course grades. This workflow needs student grades data as input. The workflow running instance should keep the grades data private during the running of the workflow. We are currently using dedicated file system in a protected local machine but plan to explore more secure approaches.



(a) (b)

Fig. 3**.** The ActivityVsGradeAnalysis workflow diagram and an output example

We also decompose our task the same way we did in the previous example to build up the workflow template: The grade and activity analysis can be represented by four computation steps. The first component (CountMessages in Fig 3(a)) computes the activity weight for each student. Here we use the number of messages posted as the

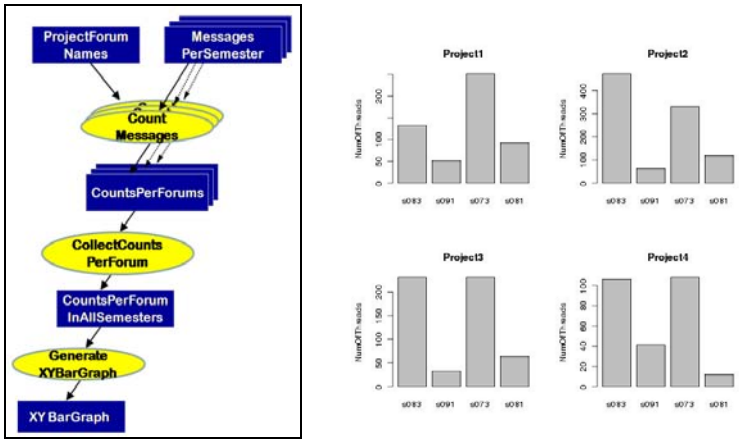
weight of a student's activity. This produces the CountsPerUser file as the output. Second, the student activity data (CountsPerUser) is *joined* with student grades (GradesPerUser) using the JoinDataPerUser component. The grade information is provided to the workflow running instance directly by the instructor. The data is stored in a dedicated local file system as described above. Once we have both grades and activities we join the data into a data table (GradesAndCounts), which indicates each student's grade and corresponding activity weight. The system then uses the data table to compute the correlation using the CorrelationAnalysis component. Finally, a graph is generated through the GenerateXYPlot component, which was also used in the previous example.

Fig 3(b) is a project grade vs. project activity box-and-whisker plot with the same data used in Fig 2. For each grade level, five number summaries are presented: minimum, first quartile, median, third quartile and maximum. The output shows that there is a trend towards higher grades for higher discussion participation, as shown by medians, but the trend is weak and the grade-activity relation is not strong.

3.3 Group Comparison: Compare student activities over semesters

How are online activities in this semester different from previous semesters?

Finally, we might want to compare online participation across different semesters. For example, we may wish to compare the number of discussion threads that correspond to each course project for four past semesters, for a specific course. In this case we need *component collections*; we need to combine the result of many duplicates of a component with different data inputs.



(a) (b)

Fig. 4. ActivityChangeOverSemesters workflow diagram and its output (A comparison of course project 1~4 between 4 semesters:07Fall, 08Spring, 08Fall, 09Spring).

As illustrated in Fig 4(a). We have the same CountMessages component as before to calculate the number of threads along all the course projects in the semester (this will be achieved by providing ProjectForumNames to the component rather than

providing UserIDs to it). Since the CountMessages component is used for each semester input, the workflow system automatically make copies of component runs based on the number of files in the input collections. The output file collections (CountsPerForums) will be fed into the CollectCountsPerForum component, which joins the count result of all the semesters and generates a data table for further visualization using GenerateXYBarGraph component.

An example output is shown in Fig 4(b), we instantiate the workflow with four semesters' of CSCI402 discussion data, which contain discussions about the courses' four project assignments per semester, and we compare the activity for each project's discussion across the semesters. The graph is valuable for instructors who wish to assess student learning from a historical perspective.

One major advantage of creating workflows as tools to analyze online education discussion is that the workflow representation is abstract, semantic and data-independent. Moreover, the components and workflows are reusable. For example, we can easily extend this workflow to analyze the number of threads about a particular discussion topic, over multiple semesters, by adding a topic detection component in front of the CountMessages component, without changing any code in the current system.

4 Related Work

Researchers are working on non-traditional, qualitative assessment of instructional discourse include [14, 15]. Our workflows can incorporate some of these as workflow components. Combined with traditional cognitive assessment methods such as assignment and exam grades, our workflow-based approach can be powerful tool in assessing impact of online learning. There have been interests in a *longitudinal* (repeated measures) analysis of student changes [5, 13]. We can use workflows that electronically track students' learning performance across courses as they progress to their degree completion.

5 Summary and Discussion

We have presented a novel learning assessment approach that is empowered by computational workflow techniques. We are providing a workflow portal for instructors, to assist them in integrating discussion participation results and traditional assessments. Initial feedback from instructors indicates that the system will help track student activities efficiently and that its assessment results will potentially change their teaching strategies.

We are currently increasing the number of assessment questions that the system can handle. Since the existing workflow system cannot effectively handle dynamic data access and repeated runs of the same workflow, we are developing new extensions to the workflow architecture that will handle diverse assessment workflows. We will also improve the user interface for readability and usability.

Acknowledgement

This work is supported by the NSF, CISE Information and Intelligent Systems award (#0917328). The authors thank Instructors Micheal Crowley, Geza Bottlik and Jim Arvo for their time and support of the project. We also thank Varun Ratnakar and Gaurang Mehta for their help with software installation.

References

1. Deelman, E., Singh, G., Su, M., Blythe, J., Gil, Y., Kesselman, C., Mehta, G., Vahi, K., Berriman, G. B., Good, J., Laity, A., Jacob, J. C., and D. S. Katz (2005). Pegasus: a Framework for Mapping Complex Scientific Workflows onto Distributed Systems, Scientific Programming Journal, Vol 13(3), (2005).
2. Gil, Y., Ratnakar, V., Kim, J., Gonzales-Calero, P., Groth, P., Moody, J., Deelman, E. WINGS: Intelligent Workflow-Based Design of Computational Experiments, IEEE Intelligent Systems. (2010).
3. Kim, J., Shaw, E. Pedagogical Discourse: Connecting Students to Past Discussions and Peer Mentors within an Online Discussion Board, The 21st Innovative Applications of Artificial Intelligence Conference (2009).
4. Ravi, S., Kim, J. Profiling Student Interactions in Threaded Discussions with Speech Act Classifiers, Proceedings of the AI in Education Conference, (2007).
5. Reed-Rhoads (2008) C16 – Tools for Assessing Learning in Engineering. Presentation on Inventions and Impact 2: Building Excellence in Undergraduate Science, Technology, Engineering, and Mathematics (STEM) Education.
6. Kim, J., Deelman, E., Gil, Y., Mehta, G., Ratnakar, V. "Provenance Trails in the Wings/Pegasus Workflow System", In Concurrency and Computation: Practice and Experience, Special Issue on the First Provenance Challenge, Vol 20, Issue 5, April 2008
7. Suthers, D., & Hundhausen, C. (2002). The effects of representations on students' elaborations in collaborative inquiry, Proceedings of Computer Support for Collaborative Learning (pp. 472-280). Hillsdale NJ:Erlbaum.
8. Stahl, G. (2002). The Complexity of a Collaborative Interaction. Proc. of the International Conference of the Learning Sciences (ICLS-2002).
9. Kim, J., Gil, Y., Spraragen, M. Principles for Interactive Acquisition and Validation of Workflows, Journal of Experimental and Theoretical Artificial Intelligence, 2009.
10. Kim, J., Spraragen, M., Gil, Y. An Intelligent Assistant for Interactive Workflow Composition, Proceedings of the International Conference on Intelligent User Interfaces, 2004.
11. OWL 2010 http://www.w3.org/TR/owl-features/
12. Jena 2010 http://jena.sourceforge.net/
13. Della-Piana, C., Pimmel, R., & Watford, B (2006). Project Evaluation, Workshop for Faculty from Minority Serving Institutions, Feb. 8 –10, 2006.
14. McLaren, B. M., Scheuer, O., De Laat, M., Hever, R., De Groot, R., & Rose, C. P. (2007). Using Machine Learning Techniques to Analyze and Support Mediation of Student E-Discussions, Proceedings of the 13th International Conference on Artificial Intelligence in Education.
15. Graesser, A. C., Olney, A., Ventura, M., Jackson, G. T. (2005) AutoTutor's Coverage of Expectations during Tutorial Dialogue. Proceedings of the FLAIRS Conference 2005: 518-523