Using Semantic Web Technologies to Analyze Learning Content

The authors demonstrate how to use Semantic Web technologies to improve the state-of-the-art in online learning environments and bridge the gap between students on the one hand, and authors or teachers on the other. The ontological framework presented here helps formalize learning object context as a complex interplay of different learning-related elements and shows how we can use semantic annotation to interrelate diverse learning artifacts. On top of this framework, the authors implemented several feedback channels for educators to improve the delivery of future Web-based courses.

Most distance and online educators would agree that Web-based teaching necessarily involves four main steps. First, educators prepare learning content in the form of reusable *learning objects*. Next, they organize these objects in accordance with a chosen instructional approach to provide a cohesive, Web-based course that includes various assessment mechanisms. As students access the course, educators receive statistics about students' interactions, although usually only in a coarse-grained tabular form. Finally, when the course ends, educators can modify it based on the observed student interactions. Currently, *learning content management systems* (LCMSs) largely support these activities (see the glossary for relevant online learning terms).

A significant issue with this workflow is that the educator is an abstract role; often many kinds of people are involved, including subject-matter experts, content authors, instructional designers, and teachers. No individual actor is fully involved in every step of the process. In many cases, subject-matter experts work with content authors (who have specific technology skills in Web content creation) to develop learning objects; instructional designers sequence, set learning outcomes, and package the content; and the teacher delivers the course to students, offering some mentorship or help. The last step in the workflow — and the one we're most interested in supporting — is content modification, which typically works best when all edu-

Jelena Jovanović and Vladan Devedžić
University of Belgrade

Dragan Gašević
Athabasca University

Marek Hatala, Timmy Eap, and Griff Richards
Simon Fraser University

Christopher Brooks
University of Saskatchewan

1089-7801/07/$25.00 © 2007 IEEE

Glossary of Relevant Online Learning Terms

- *Learning objects*: small, reusable pieces of content relevant for learning (for example, an online exercise; a coherent set of introductory readings on a specific topic; or an assessment test).
- *Learning content management system* (LCMS): a software environment that enables interactive Web-based teaching and supports administration of distance courses. Using an LCMS, instructors can distribute information to students, produce course material, prepare assignments and tests, engage in discussions, and manage distance classes. The most popular commercial LCMSs are WebCT (www.webct.com) and Blackboard (www.blackboard.com), which recently merged. The open source community offers Moodle (http://moodle.org) and iHelp Courses (http://ihelp.usask.ca/).
- *Sharable courseware object reference model* (*SCORM*): a collection of standards and specifications adapted from multiple sources to provide a comprehensive suite of e-learning capabilities that enable interoperability, accessibility, and reusability of Web-based learning content (www.adlnet.gov/scorm/index.cfm).

cational actors are involved, although this task often falls to only one person for cost reasons.

Our research over the past several years has aimed to augment e-learning environments with semantic-rich awareness information. We intend educators (in particular, content authors and instructional designers) to use this information when updating learning content for future courses. To this end, we've developed numerous heuristics that enable both the intelligent analysis and conversion of LCMS log data into Semantic Web ontologies. We've implemented these heuristics and ontologies within the iHelp Courses LCMS[1] and the popular Reload content packaging tool (www.reload.ac.uk/editor.html), and we use them to link specific learning objects to the student interactions observed within the LCMS environment. Thus, we turned Reload into a Semantic Web application that focuses on actual people involved in the learning process (learners and educators). It demonstrates how we can leverage Semantic Web technologies to provide contextual information that improves Web-based courseware.

Feedback Generation

iHelp Courses is a state-of-the-art LCMS that teachers can use to deploy their courses on the Web. Before they can deploy content, however, educators must package it; authors typically use the popular open source Reload Content Packaging Editor for this purpose. Learners then interact directly with content within the iHelp Courses system, which includes asynchronous and synchronous discussion tools, assessment mechanisms, and a shared text editor and annotation tool.

Each time a user (either a learner or an educator) accesses a learning object through iHelp Courses, the system records this interaction, including all computer-mediated communication that goes on within the iHelp environment, such as page-load requests; how many times the students watched a video or followed an external link; and assessment outcomes (results from multiple choice quizzes). Based on this captured interaction data, our LOCO-Analyst tool supplies online educators with awareness of student activities within a course. Specifically, LOCO-Analyst imports the interaction data from iHelp Courses (or any other LCMS) and transforms it into an ontological framework we've dubbed the *learning object context ontologies* (LOCO),[2] in which every user action is reformulated into RDF. Subsequently, it processes this RDF data to generate meaningful feedback for educators responsible for updating course material.

We developed LOCO-Analyst as an extension of the Reload editor to leverage the familiarity that educators already had with packaging courses. Our solution is nonintrusive, in that we don't change the overall workflow of the actors involved — rather, we just supply them with more information at critical steps.

Feedback Types

The notion of learning object context emerged as a part of our efforts to facilitate learning objects' reuse[3,4] and advanced levels of personalized learning.[2] We define learning object context as a specific learning situation, determined by the learning activity, the learning content, and the (group of) learners involved.

In July and August 2006, we conducted a small-scale survey of online educators' current practices and requirements. We included instructors and designers from three Canadian universities, as well as members of the International Forum of Educational Technology & Society (http://ifets.ieee.org) mailing list, a well-known group of developers, researchers, and instructional designers from around the world. Fifteen participants completed the survey, each providing highly informative comments. Among other important findings, all survey participants reported a lack of feedback about the learning process. We consider feedback to be information about observed learners' interactions either with learning content or with other participants in the learning process. We determined six levels of feedback based on the responses we received from the survey:

1. *Better recognition of problems at a coarse-grained level.* This feedback begins with an analysis of students' results on the quiz for a given course module. In the first step, we compare students' final scores, and then focus on students who performed poorly (that is, those with quiz scores below the average) with the objective of finding out why. We check each student's results on quizzes done in the other modules; if they performed poorly on other quizzes as well, we conclude that the problem lies in the students' motivation, working habits, or some kind of cognitive disability. However, if most students performed well on the other quizzes, it might signal to the teacher that something is wrong with the applied teaching approach (the learning design) or the respective module's learning content.
2. *Better recognition of differences between successful and unsuccessful learning trajectories.* This starts from a quiz and compares students' performance. Based on that, we divide the students into two groups: those who performed well and those who performed poorly. We then try to uncover some commonalities (patterns) in all the students' learning trajectories, reconstructing these trajectories by first identifying the activities a student performed prior to taking the quiz and subsequently sequencing those activities in the order in which they took place. We look for patterns such as "students who first took lesson on the topic *X* and then proceeded to *Y* and *Z* and who were also actively discussing the topics of *Z* while learning about it performed better that those who skipped *X* and went straight to *Y*." Such patterns are valuable for teachers because they help them rethink the applied teaching approach.
3. *Detection of content that was hard for students to comprehend.* This feedback focuses on students' readings to find lessons that were difficult for them. For each lesson, we calculate the average number of times the students revisited it, as well as the average time they spent on it during each revisit. In addition, we use some simple calculations to determine a lesson's cognitive load. In particular, we compare average time spent reading a lesson with the lesson's "size" (such as the number of words, graphs, and pictures). We assume that a lesson with a high time-spent-to-lesson-size ratio (that is, an above-average ratio) might have posed problems for students because they needed more time than usual to go through it. If a lesson with a high cognitive load also has a high number of "revisits," we classify it as difficult for students.
4. *Identification of student difficulties at a topic level.* This focuses on the quiz-question level to identify those that were most difficult for students — that is, the questions that most students answered incorrectly. We first relate each identified question to the domain topics it covers and then to the lessons discussing those topics. The link between quiz questions and lessons is possible due to semantic annotation of the learning content (that is, annotations with regard to concepts of domain ontologies, which we describe later). In addition, by exploiting the semantic annotations of the messages students exchanged in online discussions, we can verify previously drawn conclusions about difficult domain topics and give teachers more reliable information. In particular, we identify whether and how often students discussed difficult topics and use that information as an additional parameter when judging a topic's difficulty.
5. *Identification of frequently discussed topics.* This focuses on the content of messages exchanged in students' online discussions (discussion forums and chat rooms). For each message, we try to identify whether it discusses any of the domain topics (that is, those from the domain ontology discussed later) or if it was a (frequent) social "Hi, how are you?" message. If we can identify at least one domain topic, we classify the message as "course related"; otherwise, we assume it's a "social" one. We based topic recognition on various text-mining techniques applied to the messages' content. In addition, we're currently combining some simple heuristics with these techniques to identify whether a message expresses a student's complaint (not understanding the content, for example), a plea for help, or an inquiry. The feedback that these analyses produce could help teachers

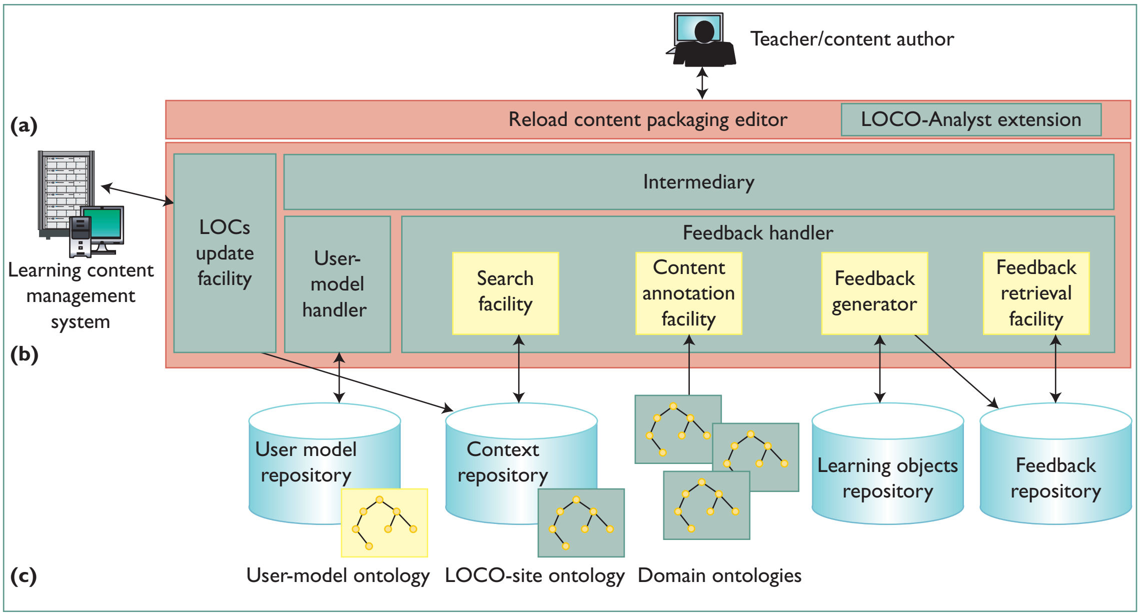

Figure 1. The LOCO-Analyst's three-tier architecture. (a) The user interface is built on the Reload editor with our extension for learning feedback. (b) The application logics is the core part of the tool responsible for semantic data integration and analysis. (c) The semantic repositories include different kinds of learning-related data stored in ontology-based data repositories.

learn more about the social network in their online classrooms.

6. *Identification of student's level of engagement in online interactions.* This feedback requires a detailed analysis of a student's online interactions. We analyze students' behavior in discussion forums and chat rooms: were they just reading messages or also sending messages to others? Were they equally active in all discussion forums and chat rooms or only in some of them? Have they ever started a thread in discussion forums or chat rooms and, if so, on which topics? How many other students have they interacted with — was it always with the same person or with several? Being informed about how active students were in social interactions, teachers can more easily decide how to alter their teaching approach to activate students or make them more focused on a lesson's relevant parts. Currently, we do this analysis for individual students, but we could apply it to study groups as well.

These findings are in accordance with two recent empirical studies that investigated instructors' needs when teaching from a distance using LCMSs.[5,6]

LOCO-Analyst

LOCO-Analyst helps educators by increasing their awareness about the quality of the learning process that occurs in the online courses for which they're responsible. This increased awareness comes from the feedback types that LOCO-Analyst generates.

Architecture

LOCO-Analyst is a classical three-tier architecture, in which the bottom layer is composed primarily of various Semantic Web data repositories (see Figure 1) that include the following:

- *Learning object repository.* A repository of learning objects stored as *shareable courseware object reference model* (SCORM) content packages.
- *Context repository.* A semantic repository of learning object context data that holds instances of the LOCO-Cite ontology, based on the popular Sesame engine (www.openrdf.org).
- *User-model repository.* Another Sesame-based semantic repository that stores user models (profiles) specific to an education-based user model ontology.
- *Feedback repository.* A repository of generated feedback, stored for quick access. Due to the nature of the analyses it performs when generating feedback, LOCO-Analyst can't instantly respond to users' requests for feedback. Instead, it performs these analyses in the background and caches the results in this repository.

- *Repository of domain ontologies.* A repository that hosts domain ontologies — that is, those formally representing the subject domain of the learning content stored in the learning object repository.

The system's heart is encapsulated in the architecture's middle layer. It includes four well-decoupled modules: *feedback handler*, *user-model handler*, *intermediary,* and the *learning object contexts* (LOCs) *update facility*.

The feedback handler, the most complex module, consists of four components:

- The *search facility* performs search and retrieval of learning object context data from the context repository.
- The *content annotation facility* semantically annotates (with the terms from domain ontologies) the content used or generated during the learning process (for example, discussion forum postings). To perform semantic annotation, this component uses two cutting-edge Java frameworks: the General Architecture for Text Engineering (GATE; http://gate.ac.uk) and the Knowledge & Information Management (KIM) Platform (www.ontotext.com/kim/).
- The *feedback generator* integrates the logic for generating feedback from learning object context data and annotated learning content.
- The *feedback retrieval facility* retrieves feedback previously generated and stored in the feedback repository.

The user-model handler deals with all requests for accessing or updating the user-model repository. The intermediary module serves as a mediator between the application layer and the user interface layer and enables decoupling between the application logic and the user interface. Finally, the LOCs update facility occasionally "refreshes" the context repository's content. In particular, it takes log data from an LCMS, transforms them into a set of instances of the LOCO-Cite ontology, and stores the generated instances into the context repository. The top-most layer handles interactions between the system and users.

Underlying Ontologies

We based LOCO-Analyst on the ontological framework we developed in our previous work[2] to enable formal representation of learning object context and its principle building blocks: different kinds of learning activities, participants in the learning process (learners, teachers, teaching assistants, and so on), and the learning content.

The LOCO-Cite ontology is the central part of this framework. It comprises several classes and properties aimed at formally representing learning object context. The ontology's core class is the `LearningObjectContext` class, which is related to the activity (`Activity`) that one or more learners (`um:User`; `um` denotes the user-model ontology's namespace) undertook while interacting with the learning content (`LearningContent`). To be consistent with our definition of the learning object context, an instance of `LearningObject-Context` must relate to exactly one `Activity` instance as well as one `LearningContent` instance. On the other hand, it can relate to more than one `um:User` instance when a collaborative activity engages more than one user (as in online discussions). We've recognized a few types of activities typical for every online learning environment and modeled them as subclasses of the `Activity` class: `Reading`, `Quizzing` (the class that models assessment activity), and `Discussing`. In addition, we distinguished between synchronous (chatting) and asynchronous interactions (such as participating in discussion forums) and modeled these by subclassing the `Discussing` class. Each recognized activity is further formally described with a set of ontological classes and properties.

LOCO-Cite refers to numerous ontologies related to online learning (domain ontology, user-model ontology, an ontology of learning design, and content structure ontology, for example). Here, we name just two types used in LOCO-Analyst.

Domain ontologies allow for defining a specific domain conceptualization. In our experiments, we focused on one specific domain — introductory computer science for nonmajors. We intended to use a larger (1,000+ concepts) ontology we developed in an earlier project,[7] but to be able to use KIM's tools for semantic annotation, we had to adjust this ontology to KIM's requirements for knowledge modeling — that is, we had to make it compliant with the Proton upper-level ontology (http://proton.semanticweb.org).

To model participants in the learning process, we refined the user-model ontology developed in the scope of a previous project.[8] The ontology represents characteristics common to all participants, such as some basic personal data and preferences,

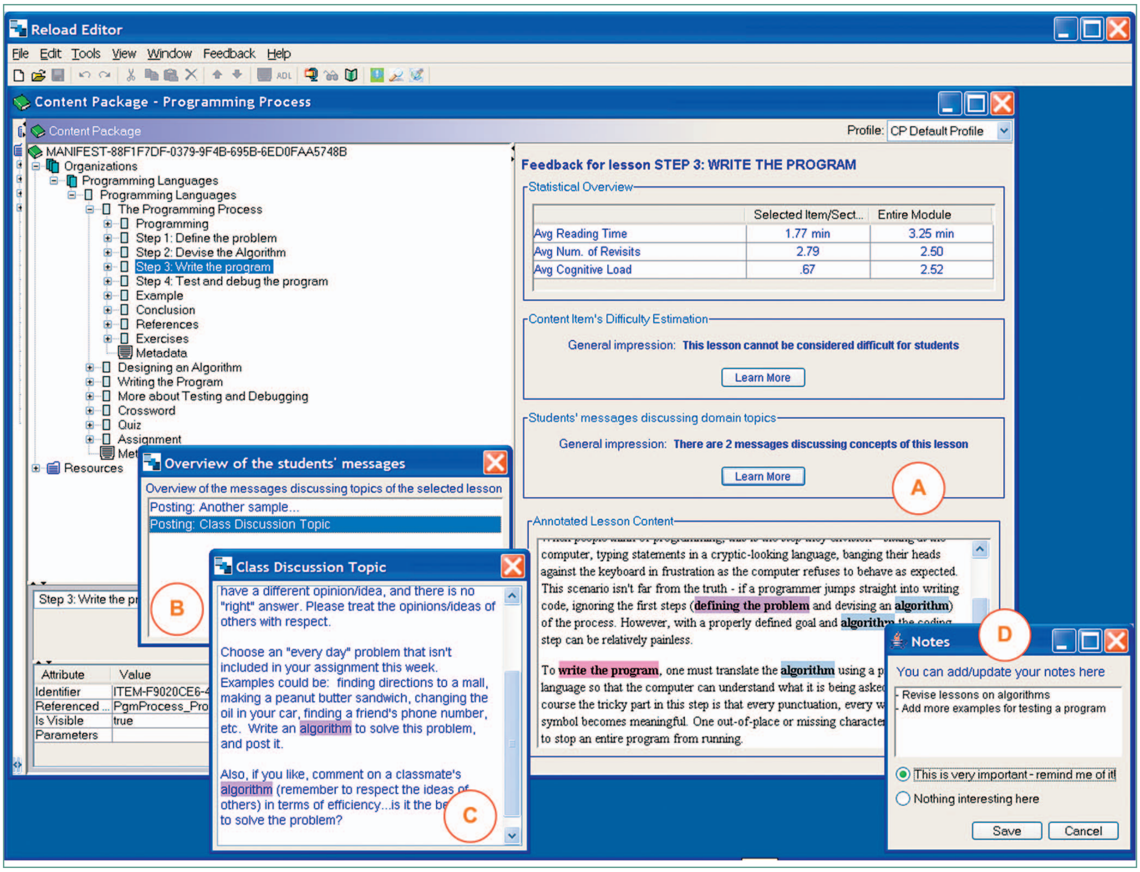

Figure 2. Feedback for the lesson "Step 3: Writing the program." The left half of the screen comes from the Reload editor and shows the lesson plan and some general content packaging data. The right half shows the generated feedback.

as well as learner-specific features, such as performance and different dimensions of learning styles. You can inspect all ontologies at http://iis.fon.bg.ac.yu/LOCO-Analyst/loco.html.

The LOCO-Analyst System

An educator using LOCO-Analyst can create new feedback requests according to preferences they've set, which include limiting the scope of the content they're interested in, identifying students they wish to get information or feedback about, and restricting other aspects (such as time) of the contexts to be examined. LOCO-Analyst uses a color-coded scheme to help educators differentiate previous requests according to the previously self-assigned priorities or relevancies. Selecting a previously fulfilled request from the list, the user can inspect its details, mark interesting parts, and take or append notes. We can view these notes, as well as the assigned relevancy level, as a kind of metadata for the feedback.

LOCO-Analyst provides feedback at diverse levels of granularity, from a single lesson to an entire module or course. Thus, educators receive layered information that can help them identify trouble areas in a course. Figure 2 presents feedback at the lesson level entitled "Step 3: Writing the program." The left half of the screen is "inherited" from the Reload editor and shows the SCORM manifest tree (the lesson plan) and some general content packaging data. The right half is intrinsic to LOCO-Analyst and shows the generated feedback.

Note that the feedback panel is divided into four subpanels, each providing a specific feedback type as described previously. The top-most panel shows a general statistical overview for the lesson, letting us compare the average reading time, number of revisits, and cognitive load of the selected lesson with the module as a whole. Because they can see the lesson and module's average values next to each other, teachers can easily identify whether the lesson required more students' engagement and consider revising its content or altering the applied teaching method.

The second panel from the top shows the

Related Work in Semantic Rich Educational Feedback

Riichiro Mizogushi and Jacqueline Bourdeau wrote the pioneering work on using Semantic Web technologies for e-learning.[1] Afterward, many authors proposed using ontologies in different aspects of learning, such as adaptive educational hypermedia systems, adaptive content authoring, personalized search and content composition, user-model sharing, and context capturing. Several international journals,[2–5] along with the Applications of Semantic Web Technologies for E-Learning workshop series (www.win.tue.nl/SW-EL/), nicely illustrate the state-of-the-art in this area. We improve on this research by demonstrating how Semantic Web technologies enable a generic implementation of feedback for content authors and teachers based on students' actual activities in specific learning contexts.

Our work is closely related to Web usage mining in the e-learning domain. For example, Osmar R. Zaïane and Jun Luo applied advanced data mining techniques on an LMCS's Web access logs to help educators assess the learning process, track students actions, and measure Web course structure effectiveness.[6] The Tool for Advanced Data Analysis in Education (TADA-Ed) is another data mining platform aimed at discovering pedagogically relevant patterns in students' online exercises.[7]

Some researchers have applied information visualization techniques to help teachers understand what's happening in their classes. CourseViz works with the WebCT LCMS to produce various graphical representations of student tracking data,[8] and the Graphical Interactive Student Monitoring System (GISMO) takes a similar approach to the Moodle LCMS.[9]

References

1. R. Mizoguchi and J. Bourdeau, "Using Ontological Engineering to Overcome AI-ED Problems," *Int'l J. Artificial Intelligence in Education*, vol. 11, no. 2, 2000, pp. 107–121.
2. D.G. Sampson et al., eds., *Educational Technology & Society*, special issue on ontologies and the Semantic Web for e-learning, vol. 7, no. 4, 2004.
3. D. Dicheva and L. Aroyo, eds., *Int'l J. Continuing Engineering Education and Life-Long Learning*, special issue on application of Semantic Web technologies in e-learning, vol. 16, nos. 1/2, 2006.
4. D. Dicheva and L. Aroyo, eds., *Int'l J. Continuing Engineering Education and Life-Long Learning*, special issue on concepts and ontologies in Web-based educational systems," vol. 14, no. 3, 2004.
5. A. Naeve et al., eds., *British J. Educational Technology*, special issue on Advances of the Semantic Web for E-Learning: Expanding Learning Frontiers, vol. 37, no. 3, 2006.
6. O.R. Zaïane, and J. Luo, "Towards Evaluating Learners' Behavior in a Web-Based Distance Learning Environment," *Proc. 1st IEEE Int'l Conf. Advanced Learning Technologies*, IEEE Press, 2001, pp. 357–360.
7. A. Merceron and K. Yacef, "TADA-Ed for Educational Data Mining," *Interactive Multimedia Electronic J. Computer-Enhanced Learning*, vol. 7, no. 1, 2005; http://imej.wfu.edu/articles/2005/1/03/index.asp.
8. R. Mazza and V. Dimitrova, "Visualising Student Tracking Data to Support Instructors in Web-Based Distance Education," *Proc. 13th Int'l World Wide Web Conf.*, ACM Press 2004, pp.154–161.
9. R. Mazza, and C. Milani, "Exploring Usage Analysis in Learning Systems: Gaining Insights From Visualisations," *Proc. Workshop on Usage Analysis in Learning Systems*, 2005, pp. 65–72; http://hcs.science.uva.nl/AIED2005/W1proc.pdf.

LOCO-Analyst's conclusion regarding the lesson's difficulty for the students (feedback 3 from the list). The system gives its general conclusion about the lesson's difficulty as a statement: "this lesson can/cannot be considered difficult for students." In addition, the teacher can learn more about the deduction LOCO-Analyst makes by clicking on the Learn More button, which brings up a dialog box listing the module's difficult lessons, sorted in descending order based on their estimated difficulty. The currently reviewed lesson is colored red, so the teacher can easily spot its position in the list.

The third panel (A) is devoted to online discussions that LOCO-Analyst finds relevant for the currently observed lesson. We consider a message from a chat room or a discussion forum as relevant for a certain lesson if it mentions domain topics that the lesson at least partially covers. Again, the system gives a general statement — "there are X messages discussing concepts of this lesson"— and a Learn More button. After clicking on this button, the teacher receives a new dialog box that lists student messages related to the lesson (B). The educator can double-click on any of these messages to see their content, which the system has semantically annotated. Terms representing domain concepts are highlighted using different colors (box C), letting teachers easily spot which topics the message discusses and in what tone — is the message a complaint from a student unable to understand the respective topic, an inquiry, or something else?

The bottom-most panel previews the lesson's content, which is semantically annotated with domain concepts, each highlighted with a different color. Using similarly organized panels, LOCO-Analyst gives educators feedback at the level of a lesson composed of two or more content items or a module as a whole. In addition, it has a specific panel for presenting feedback generated from assessment mechanisms. LOCO-Analyst can also inform educators about any student's activities and social networking (feedback 6 from the list). Finally, the educator can make notes reviewing feedback, using the dialog from box D.

*Table 1. Survey results for LOCO-Analyst's usefulness.**

Feedback	Very good (%)	Good (%)	Neutral (%)	Weak (%)	Very weak (%)	Mean	Standard deviation
Individual lesson	66.7	22.2	11.1	0.0	0.0	4.56	0.6849
A group of lessons	61.1	38.9	0.0	0.0	0.0	4.61	0.4875
A module as a whole	50.0	38.9	11.1	0.0	0.0	4.39	0.6781
Students' performance on a quiz	50.0	38.9	5.55	5.55	0.0	4.22	0.8240
Students' online interactions	50.0	33.3	0.7	0.0	0.0	4.33	0.7454
Students' way of learning	38.9	61.1	0.0	0.0	0.0	4.39	0.4875

*Out of 18 participants from three universities.

Evaluation

We've conducted an evaluation to determine whether and to what extent educators find LOCO-Analyst useful for everyday practice. The study involved 18 participants with online teaching experience from the University of Belgrade, Simon Fraser University, and the University of Saskatchewan. After giving participants basic instructions on how to install and use the tool, we asked them to test it and subsequently fill out a questionnaire.

The first part of the questionnaire captured the participants' backgrounds. It revealed that only 28 percent of the participants had previously used a tool similar to LOCO-Analyst. When asked to compare LOCO-Analyst with other tools that they'd used for feedback provision, most participants stated that LOCO-Analyst's feedback was easier to understand and that other tools were difficult to use due to the large amount of uninterpreted data and unnecessary details they provided. Participants also noted that, in addition to providing some basic statistical data, as did similar systems, LOCO-Analyst gave qualitative estimates of the content difficulty and highlighted domain-specific topics. This showed that the participants recognized the benefits of the Semantic Web technologies our system employed. However, one participant found that the system incorrectly annotated some messages or content. This occurred due to the inherent imprecision found in KIM (and every other state-of-the-art semantic annotator).

The second part of the questionnaire asked participants to evaluate the system's functionalities. As already explained, each of LOCO-Analyst's feedback panels integrates one or more feedback types to provide comprehensive feedback on a certain level of content granularity (single lesson, composite lesson, and module) and content type (lesson or quiz). We hypothesized that these aspects are more valuable together. To test this, we first asked participants to rate the feedback LOCO-Analyst provided using a Likert-like scale. Participants' opinions were heavily weighted toward the positive, as Table 1 indicates. Subsequently, we asked questions related to each individual feedback type. We wanted to know, for example, if the system gives users insight into the students' understanding of the domain topics. Whereas 60 percent of the participants answered "yes," 36 percent answered "partially," and 4 percent didn't have an opinion. We obtained similar results for other individual feedback types. On further analyses of the received answers, we deduced that the synergy effect was indeed present, and that when presented together, individual feedback types are much more effective.

For many participants, the feedback on interaction between students needs improvement. Even though most of them (77 percent) acknowledged that the system provides relevant information about students' interactions, several participants indicated that they weren't that satisfied with its ability to indicate how to improve communication. Several also acknowledged that the improvement of students' collaboration depends on many factors that are difficult to capture and measure in an LCMS and that the information the LOCO-Analyst provided at least gave them a starting point.

In the last part of the questionnaire, we asked the participants to evaluate the user interface. Their general impression was predominantly positive (roughly 90 percent found it either "very good" or "good"). Additionally, most found the interface intuitive and not overburdened with information. Because most of the free-form comments requested better visualizations of the results, we've already started working on integrating different visualization techniques into LOCO-Analyst.

Our research has demonstrated the use of Semantic Web technologies for improving the state-of-the-art in e-learning. Besides addressing the comments we obtained in the evaluation (especially those related to the visualization of interactions), we're working on reducing the overhead in domain ontology creation by using text-mining techniques, concept graph extraction, and ontology learning, which should make LOCO-Analyst easy to deploy in different domains. We also want to employ research on ontology mapping so that we can generate feedback from various learning environments that don't use the same domain ontology.

References

1. C. Brooks, L. Kettel, and C. Hansen, "Building a Learning Object Content Management System," *Proc. 10th World E-Learn Conference*, Assoc. for the Advancement of Computing in Education, 2005, pp. 2836–2843.
2. J. Jovanović et al., "Learning Object Context on the Semantic Web," *Proc. 6th IEEE Int'l Conf. Advanced Learning Technologies*, IEEE Press, 2006 pp. 669–673.
3. C. Brooks and G. McCalla, "Towards Flexible Learning Object Metadata," *Int'l J. Continuing Engineering and Life-long Learning*, vol. 16, nos. 1/2, 2006, pp. 50–63.
4. C. Knight, D. Gašević, and G. Richards, "An Ontology-Based Framework for Bridging Learning Design and Learning Content," *Educational Technology & Society*, vol. 9, no. 1, 2006, pp. 23–37.
5. R. Mazza and V. Dimitrova, "Informing The Design of a Course Data Visualisator: An Empirical Study," *Proc. 5th Int'l Conf. New Educational Environments*, ACM Press, 2003, pp. 215–220.
6. C. Zinn and O. Scheuer, "Getting to Know Your Student in Distance-Learning Contexts: Innovative Approaches for Learning and Knowledge Sharing," *Proc. 1st European Conf. Technology Enhanced Learning*, Springer, 2006, pp. 437–451.
7. M. Winter et al., "Using Semantic Web Methods for Distributed Learner Modelling," *Proc. 2nd Int'l Workshop Applications of Semantic Web Technologies for E-Learning*, 2004; www.win.tue.nl/SW-EL/2004/ISWC-SWEL-Camera-ready/%236-Winter-CR.pdf.
8. J. Jovanović, D. Gašević, and V. Devedžić, "Dynamic Assembly of Personalized Learning Content on the Semantic Web," *Proc. 3rd European Semantic Web Conf.*, LNCS 4011, Springer-Verlag, 2006, pp. 545–559.

Jelena Jovanović is a PhD candidate and a teaching assistant at the Department of Software Engineering in the FON—School of Business Administration at the University of Belgrade. Her research interests include Semantic Web technologies and technology-based learning. Jovanović has a BS and an MSc in information systems from the University of Belgrade. Contact her at jeljov@gmail.com.

Dragan Gašević is an assistant professor in the School of Computing and Information Systems at Athabasca University. His research interests include the Semantic Web, model-driven software engineering, knowledge management, service-oriented architectures, and learning technologies. Gašević has a BS, an MSc, and a PhD in computer science from the University of Belgrade. Contact him at dgasevic@acm.org.

Christopher Brooks is a PhD student and a research assistant in the Advanced Research in Intelligent Educational Systems (ARIES) laboratory in the University of Saskatchewan's computer science department. His research interests include learning object technologies, distributed computing, Web service technologies, and the visualization of tracking data. Brooks has a BS and an MSc in computer science from the University of Saskatchewan. Contact him at cab938@mail.usask.ca.

Vladan Devedžić is a professor in and head of the Department of Software Engineering in the FON—School of Business Administration at the University of Belgrade. His research interests focus on the practical engineering aspects of developing intelligent educational systems for the Web, and his long-term goal is to merge ideas from intelligent systems and software engineering. Contact him at devedzic@fon.bg.ac.yu.

Marek Hatala is an associate professor and director of the Laboratory for Ontological Research at the School of Interactive Arts and Technology, Simon Fraser University, Surrey. His research interests include knowledge management, artificial intelligence, distributed systems, user modeling, e-learning, and collaborative systems. Hatala has an MSc and a PhD in computer science from the Technical University of Kosice. Contact him at mhatala@sfu.ca.

Timmy Eap is a PhD candidate at Simon Fraser University. His research interests include e-learning management, Internet security, artificial intelligence, and the Semantic Web. Eap has a BSc and an MSc from Simon Fraser University. Contact him at teap@sfu.ca.

Griff Richards is an adjunct professor at Simon Fraser University and Athabasca University. He has spent 25 years in the research and development of computer applications in education. Richards has a PhD in educational technology from Concordia University. He has managed to survive each onslaught of new technology by holding fast to sound pedagogy that promotes interaction of the learner with content, self, and others. Contact him at griff@sfu.ca.