Intelligent Learning Object Guide (iLOG): A Framework for Automatic Empirically-Based Metadata Generation

S.A. RILEY[a] , L.D. MILLER[a], L.-K. SOH[a], A. SAMAL[a], and G. NUGENT[b]
[a] *University of Nebraska—Lincoln: Department of Computer Science and Engineering*
[b] *University of Nebraska—Lincoln: Center for Research on Children, Youth, Families and Schools*

Abstract. We present a framework for the automatic annotation of learning objects (LOs) with *empirical usage metadata*. Our implementation of the Intelligent Learning Object Guide (iLOG) was used to collect interaction data of over 200 students' interactions with eight LOs. We show that iLOG successfully tracks student interaction data that can be used to automate the creation of meaningful *empirical usage metadata* that is based on real-world usage and student outcomes.

Keywords., Learning Objects, Feature Selection, Association Rule Mining, Empirical Usage Metadata, SCORM

Introduction

A high school math teacher and a college matrix theory professor each search an online learning object repository for a learning object (LO) on "matrix multiplication." The needs of their respective classes are probably quite different, but the instructors both have the same desire—to locate an LO that will provide a successful learning experience for as many students as possible. However, locating an appropriate LO is not a straightforward process: students have varied background knowledge, experience, motivation, self-efficacy, and other learning characteristics. Finally, not all LOs are created equal—some LOs will inevitably 'work' better on average than others.

Not all LOs are designed with the same type of learner in mind; furthermore, with real-world usage, unexpected patterns may emerge: an LO may be unsuitable for students with low motivation or for those without Calculus experience, or the LO might carry an inherent gender bias. The point is, *we cannot be certain what will happen when real students use an actual LO*—but this information may be critical in making an informed LO selection decision. And for an instructor tasked with locating a suitable LO, without this type of information, the selection process may be daunting enough to send them right back to the textbook and chalkboard.

One possible way to approach this problem is to tag each LO with insightful information regarding how it has been used and the impact it has had on learning; we hereafter refer to this information as *empirical usage metadata*. This multi-dimensional metadata allows LOs to be indexed and searched not only by content but also by usage history. This should precipitate a radical improvement in instructors' ability to identify high-quality LOs that match the educational, experiential, and affective backgrounds of students. For example, it would accommodate the identification of LOs by way of high-level usage statistics and rules ranked by their relative strength: (high motivation → pass .51), or (highSchoolStudent → fail .65), (averageTime = 653 seconds) would have alerted our high school math teacher that this LO is too advanced for her class.

Further, one of the main benefits of LO standards is the complete decoupling of the learning management system (LMS) from the LO itself, because this allows content to be interoperable, maintainable, and (in theory) discoverable and reusable. However, *widespread discoverability and reusability of LOs has been largely unrealized, likely due to insufficient metadata*. A recent real-world study [2] of the most widely used standard for tagging LOs, the IEEE Learning Object Metadata (LOM) [1], showed that LOM metadata are typically *incomplete, inaccurate, and not machine-interpretable*. Thus, a system that automatically tags LOs with empirical usage metadata will have a profound impact on the level of discovery and reuse of LOs. Such a system should have the following properties: (1) ***general***: based on widespread e-learning standards; (2) ***automatic***: metadata annotation should not require human intervention; and (3) ***interpretable***: metadata should be both human- and machine-readable.

We have implemented an **Intelligent Learning Object Guide** (iLOG) to automatically generate empirical usage metadata for Sharable Content Object Reference Model (SCORM)-conformant LOs. The **MetaGen** component of iLOG uses feature selection to isolate the attributes most salient to successful learning outcomes, and predictive rule mining to automatically generate empirical usage metadata in the form of rules, which are in turn used to update the LOM file associated with the SCORM LO.

Note that the use of such rich metadata is generally associated with Intelligent Tutoring Systems rather than standardized e-Learning systems; however, a recent paper [3] outlines many opportunities in merging the two approaches. The authors also outline several challenges, including the need for what they term *contextualized metadata* and the need for *metadata validation*; these are the needs which iLOG will address.

1. Related Work

To locate the most appropriate LO for a given situation, we must consider two related problems: (1) finding a set of LOs that match the desired topic, and (2) selecting the LO that best matches the educational, experiential, and affective needs of students.

Existing work on automatic generation of standardized LO metadata [4], [5] primarily focuses on extracting ontological and taxonomic information from raw learning content. This work will help a search engine in the retrieval of LOs that more closely match the target topic, thus satisfying need (1). We do not focus on this facet of the problem in our own work, but instead build upon it to aid in satisfying need (2).

Thus, we need contextual LO metadata to relate student profiles and interaction behaviors (log files) with learning outcomes. Log files can be defined as a record of student interactions with a learning environment. Systems that mine information from log file data have been shown to have a positive impact on the quality of both instruction and learning [8]. However, these systems have primarily focused on automatic sequencing of content, feedback for course authors [7], and feedback for instructors [8].

We see two key differences in our research: (1) we are working strictly within existing LO metadata standards, and (2) we focus on mining empirical metadata rules exclusively from pre-existing data sources and log files for the purpose of contextual search and retrieval of LOs in learning object repositories.

2. Empirical Usage Metadata

As mentioned in the Introduction, we define *empirical usage metadata* as the set of metadata covering: (1) how the LO has been used: the actual usage data as determined by empirical evidence; (2) the impact the LO has had on learning. This information facilitates better understanding of an LO's real-world impact on learning, and also provides meaningful data for researchers, courseware developers, and instructors.

iLOG utilizes data from three key sources: (1) static LO data, describing the content of the LO, (2) static learner data, describing the learning context of a student, and (3) the student-LO session log file. These data are used to generate empirical usage metadata (see examples in Table 1). The use of these three data sources makes it possible to compute more precise usage metadata. iLOG's empirical usage metadata are further organized by the unit of "study", where each study is self-contained and often represents the deployment of an LO to a single class. A study is composed of statistics and association rules that relate student characteristics to learning outcomes. We organize the data by study (while also maintaining an LO lifetime metadata summary) primarily to mitigate the potential for an individual study to skew the metadata.

Table 1. Examples of static and dynamic parameters iLOG uses to generate empirical usage metadata.

Static Learner Data	Static LO Data	Interaction Data
Baseline motivation	Topic	Total time on tutorial
Baseline self-efficacy	Length	Total time on exercises
Gender	Degree of difficulty	Total time on assessment
Major	Level of feedback.	Min time spent on a tutorial page
GPA	Blooms' level for assessment questions	Max time spent on a tutorial page
SAT/ACT score	⁞	Avg. time per assessment question
⁞		⁞

3. The iLOG Framework

In this section, we describe the two main components in the ***iLOG framework*** (as shown in Figure 1): (1) the ***LO wrapper***, which is responsible for logging student interactions and updating the LO metadata, and (2) ***MetaGen***, which processes the log file data in order to generate empirical usage metadata that is then sent back to the LO wrapper. As discussed in the Introduction, to satisfy the complementary goals of LO reusability and discoverability, the logging and annotation process should be general, automatic, and interpretable. Thus, iLOG adheres to the SCORM and LOM learning standards; however, any combination of standards could work within this framework.

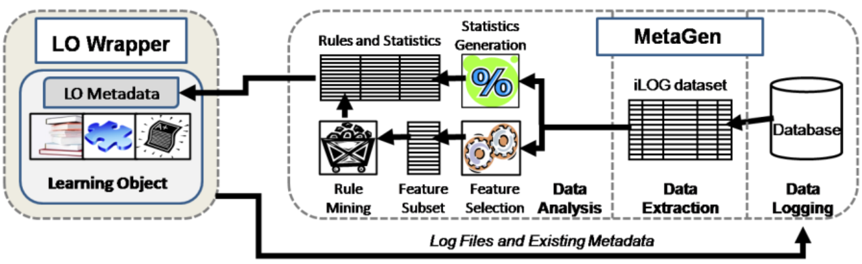

Figure 1. The Intelligent Learning Object Guide (iLOG) Framework.

The LO wrapper (Section 3.1) automatically logs a student's LO interactions and at the end of each student session this log file is forwarded to a database in the MetaGen module (Section 3.2). Next, MetaGen processes the raw interactions and runs feature selection and rule mining algorithms (Section 3.3) to generate contextual metadata. Finally, this metadata is sent to the LO wrapper, which updates the metadata file.

3.1. LO Wrapper

Most methods for tracking LO interaction data are done by altering the LMS or LO player to collect and store log file data. However, we find that it makes more sense for the tracking capabilities to reside with the LO itself. Thus, our LO wrapper can be integrated into any existing LO. The LO wrapper is responsible for (1) collecting log file data and transmitting it to MetaGen, and (2) updating the LO metadata with empirical usage metadata when it is returned by MetaGen. The main function of the LO wrapper is to provide a bridge between the LO and the MetaGen component.

First, the wrapper automatically logs each student's interactions with the LO using a web scripting language such as JavaScript to listen for web events. These include items such as: number of clicks on each page, time spent per page/question/exercise, and interactions with exercises (see table 1). It then calls the LMS's API to retrieve assessment scores, converts log and assessment data to the MetaGen database format, and again uses web scripting to transmit these to MetaGen. Once all student sessions from a study have been received, MetaGen sends a request to the LO wrapper for existing contextualized metadata. In this way, MetaGen can generate study metadata (using only data from the current deployment) and also maintain lifetime LO usage metadata.

After MetaGen completes the metadata generation process, it returns both the study and lifetime contextualized metadata to the LO wrapper in batch format. Finally, the LO wrapper converts the usage metadata to the metadata format used by the LO.

3.2. MetaGen Modules

The MetaGen component of the iLOG system generates the empirical usage metadata to automatically tag the LOs and is composed of three separate modules: (1) data logging, (2) data extraction, and (3) data analysis. The *data logging module* of MetaGen integrates data from three sources: static LO data, static student data, and log file data (from the LO wrapper). Next, the *data extraction module* extracts the iLOG dataset from the database; each dataset instance represents a single student-LO session. The *data analysis module* is a multi-step process: (1) feature selection to identify the salient features in the iLOG dataset, (2) predictive association rule mining on the iLOG data-subset, and (3) deriving usage statistics from the iLOG dataset. The data analysis step serves two important functions: *isolating salient features* associated with learning outcomes for each LO, and *generating empirical usage metadata*.

3.3. Feature Selection and Data Mining in MetaGen

As alluded to earlier, iLOG utilizes feature selection and data mining to isolate the characteristics associated with learning outcomes and subsequently mine empirical metadata using these salient features. MetaGen uses two main types of feature selection algorithms from Weka [11]: (1) those that evaluate features independently and produce a ranked list, and (2) those that pair a selection metric with a search heuristic in order to explore the space of possible feature subsets. In MetaGen, we are not simply interested in finding a *good subset*, we are also interested in *isolating as many salient features as possible* while limiting the number of irrelevant features in the subset. Table 2 shows the selection criteria and search heuristics used in MetaGen. We used more than a dozen combinations of selection metric and search heuristics as each has different strengths [9]. Then, MetaGen counts the total number of times that each feature in the iLOG dataset is selected by any of the feature selection methods, and ultimately forms the feature subset as the attributes that are most frequently selected, thresholded by majority vote. This reduction in the number of attributes not only drastically speeds up the subsequent predictive rule mining step, but it also gives us stronger rules.

Table 2. Selection Metrics and Search Heuristics used by MetaGen to rank features (from Weka [11])

Selection Metric	**Characteristics**	**Search Heuristics**	**Characteristics**
Cfs Subset	Correlation-based	Best-first	Hillclimb/ Backtrack
Chi Squared*	Correlation-based	Greedy-stepwise	Hillclimb
Classifier Subset	Classifier Accuracy	Genetic search	Genetic Algorithm
InfoGain*	Entropy-based	Random search	Random subsets
GainRatio*	Entropy-based		

*Denotes a standalone feature selection method that does not need to be paired with a search heuristic

Association rule mining is the process of generating rules that describe the co-occurrence of attributes in a dataset. The algorithm used by MetaGen is Tertius [10], which ranks the usefulness of first-order logical rules by their degree of confirmation and the relative frequency of counterexamples. For iLOG, we use LO assessment pass/fail as the target label, and use: *ruleStrength* = *confirmation* * (1 - *counterExampleFreq*) to rank the rules according to their strength. Each rule takes the form of (a set of <*att-val*> pairs) → (<*outcome*> <*ruleStrength*>).

After data analysis, MetaGen sends three types of metadata back to the LO wrapper: (1) the list of salient features; (2) the LO lifetime metadata, and (3) the rules and statistics mined from the study in question. The LO wrapper then translates these metadata into the format contained in the LO and updates the LO metadata file.

4. Implementation

First, we created eight SCORM-compliant LOs on basic computer science (CS) concepts such as conditionals, logic, arrays, looping, and functions. Each included (1) a tutorial covering the topic, (2) a set of ungraded interactive exercises, and (3) a set of assessment questions. These LOs were deployed to students using the LMS from the Blackboard Academic Suite [http://www.blackboard.com/].

Second, the LO wrapper was designed as a simple HTML document that uses Javascript to record and timestamp student interactions with the LO (e.g., page navigation, clicks on a page, etc.). The wrapper also uses a modification of the Easy SCO Adapter [http://www.ostyn.com/standards/demos/SCORM/wraps/easyscoadapterdoc.htm#license] to use the SCORM API to access student assessment results on the LMS. Then, the wrapper uses JavaScript to transmit the interactions to MetaGen on a remote site.

Finally, we implemented the three modules of MetaGen. The data logging module uses PHP to store the student interactions into a MySQL database. The data extraction module uses Java to query the database and process the data into the iLOG dataset.

The data analysis module uses the Weka [11] implementations of several feature selection algorithms to generate the iLOG data-subset, and then uses the Tertius [10] predictive rule mining algorithm to generate empirical usage metadata.

5. Experiments and Results

We deployed the iLOG system in four introductory computer science (CS) courses to over 200 students during the fall of 2008. These courses included students from a wide variety of backgrounds (e.g. non-majors, majors, and honors students). These students took an initial demographic survey and a baseline motivation/self-efficacy survey. iLOG then logged their interactions with the LOs, which each included a motivation/self-efficacy pre-survey, a tutorial, exercises, an assessment, and an evaluation post-survey. Then the MetaGen module combined the student static data, static LO data, and log file data to generate the iLOG dataset. After removing instances that were missing critical information (although the final dataset still had many missing data values), we were left with a total of 623 data instances, with 323 attributes each. For the purposes of this experiment, a *successful student outcome* is defined as a passing assessment score (70-100%), and others as unsuccessful outcomes.

5.1. Results of Feature Selection

When we ran the feature selection algorithms, the attributes in the iLOG dataset that were most often selected as associated with student outcomes varied widely across LOs. Two examples are shown in Table 3, but this variability was common to all cases. For instance, the attributes most often selected for the Searching LO (one of the more difficult LOs) were GPA and time spent on the LO, whereas the most common attributes selected for the Logic 2 LO had to do with background in Calculus, gender, and the student's opinion of the LO quality. This seems to affirm the need for empirical metadata, as it was not obvious that these particular attributes would be strongly associated with learner outcomes, or that salient attributes would be varied across LOs.

Table 3. The selected features (only top 5 shown) salient to learner outcomes vary widely among these LOs. Only two shown, but this was observed across all eight LOs.

Logic 2		Searching	
Attribute	**Number of Times Selected**	**Attribute**	**Number of Times Selected**
highestMath	16	GPA	14
gender	13	assessMinSecPageBelowAvg?	11
takenCalculus	13	assessmentMinScondsOnAPage	10
assessStdDevSecAboveAvg?	13	BeliveLODifficultToUnderstand	10
WasAnyPartConfusing?	13	courseLevel	9

5.2. Results of Predictive Association Rule Mining

In this section we present the preliminary results of iLOG's capability to automatically generate and rank viable contextualized metadata rules. For each LO, we mine the rules from the iLOG data-subset that was generated during the feature selection step outlined in Section 5.1. We ran Tertius on the iLOG data-subset for all students in all courses with 1, 2, and 3 literals included on the left-hand side of the rule, and we observe (see Table 4) that as we increase the number of literals—thus increasing a rule's specificity, we see an increase in ruleStrength. However, there is an inherent tradeoff: highly specific rules may simply capture noise in the data and overfit, these rules may be harder to apply, and computation is more time-intensive. The obvious next step is to try and automatically determine the optimal number of attributes to include in the rules.

Table 4. Some rules with one, two, and three attribute-value pairs for the Logic 2 LO

Specificity **ruleStrength**

Logic 2
takenCalculus? = no → fail .27 highestMath = precalculus → fail .25 assessmentStdDevSeconds = high →fail .24 wasAnyPartConfusing = yes → fail .23 gender = female → fail .22
takenCalculus? = yes AND assessmentMaxSecOnAPageAboveAvg? = yes → pass .35 gender = female AND materialInLODifficultToUnderstand = indifferent → fail .34 takenCalculus? = yes AND wasAnyPartConfusing? = no → pass .31 baselineStdDevMotivation = low AND assessmentMaxSecOnAPageAboveAvg? → fail .31
takenCalculus? = yes AND assessmentMaxSecOnAPageAboveAvg? = no AND wasAnyPartConfusing? = no → pass .40 takenCalculus? = yes AND believeLONeedsMoreDetails = no AND wasAnyPartConfusing? = no → pass .37 gender = male AND takenCalculus? = yes AND wasAnyPartConfusing ? = no → pass .37

When we examine the basic rules and usage statistics generated on a course by course basis, we observe that the rules generated from an identical feature subset vary by course (Table 5). For non-majors, it seems that students who fail the LO tend to spend an inordinately long time on at least one assessment question and have not taken Calculus. For CS majors (excluding Honors), low motivation and no Calculus experience were correlated with failing scores. Finally, for Honors CS, failing grades tended to be correlated with poor evaluation of the LO. Unfortunately, in two of the courses, we observe that females had a higher tendency for failure. The metadata could be used to avoid giving this LO to students who have not had Calculus and to note the possible gender bias in the LO. From the usage statistics, we also see that the Honors students spent almost twice as much time completing the LO than did the non-majors, yet achieved only slightly better results than the non-majors.

Table 5. Some metadata and statistics generated by iLOG, Logic2 LO for each class

Logic 2—Intro CS for non-majors	
successRate = 51%	**Usage Statistics**
assessmentStdDevSecondsAboveAvg? = yes → fail .35 assessmentMaxSecondsOnAQuestion = high → fail .33 highestMath = precalculus →fail .28 gender = female → fail .24	successRate = 51% averageTime = 433 seconds averageStudentRating = 4.3/5.0
Logic 2--Intro CS for majors	
Contextualized Metadata	**Usage Statistics**
baselineStdDevMotivation = low → fail .72 takenCalculus? = no →fail .52 currentTotalMotivationAboveAvg? = no →fail .52	successRate = 38% averageTime = 688 seconds averageStudentRating = 4.16/5.0
Logic 2—Honors Intro CS for majors	
Contextualized Metadata	**Usage Statistics**
OpinionOfLOUsability = negative → fail .59 BelieveLOAnAidToUnderstanding = yes → pass .49 BelieveLONeedsMoreDetail = yes → fail .43 gender = female → fail .36	successRate = 55% averageTime = 799 seconds averageStudentRating = 3.43/5.0

In addition to our contextualized metadata, we generate usage statistics for each study, as shown in Table 5. We observe that for the Logic 2 LO, the rates of successful student outcomes are consistently low and the evaluation ratings the students gave the

LO diminish as the students become more advanced. This could indicate that there are design flaws in the underlying LO and that the content may be too basic to keep advanced students interested. Such information could be used by course designers to correct design flaws and by course instructors to make intelligent LO selection choices.

For this study of eight LOs, iLOG identified an average of 15.0 features per LO as highly associated with learning outcomes. It also generated 3 sets of empirical usage metadata for each LO; as compared with the possible rules without iLOG, the number of rules with 1, 2, and 3 attribute-value pairs were reduced from ~1000, ~470,000, and ~152,000,000 to an average of 2.1, 27.4, and 85.3 rules per LO, respectively.

6. Conclusions and Future Work

We have described an implementation of the Intelligent Learning Object Guide (iLOG) framework for automatic generation of empirical usage metadata—predictive association rules that are both human- and machine-readable—and usage statistics. We use standard feature selection and predictive association rule mining to automatically generate these rules and are able to gain useful insights in terms of general and unique usage properties of the different LOs for different types of students. As future work, to provide higher-confidence metadata for search engines and educators at higher resolution, we will add clustering to the MetaGen data analysis step to split the data into natural clusters and mine rules accordingly. Additionally, we are working to extend the MetaGen framework to also generate contextualized metadata for other perspectives on the learning process, including the student, instructor, and course designer perspectives.

This material is based upon work supported by the National Science Foundation under Grant No. 0632642 and an NSF GAANN fellowship.

References

[1] IEEE 1484.12.1-2002 Standard for Learning Object Metadata (LOM). Retrieved January 7, 2009, from http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf

[2] N. Friesen, The International Learning Object Metadata Survey. Retrieved August 7, 2008, from http://www.irrodl.org/index.php/irrodl/article/view/195/277/

[3] C. Brooks, J. Greer, E. Melis, C. Ullrich, Combining ITS and eLearning Technologies: Opportunities and Challenges, *Proc. 8th Int. Conf. on Intelligent Tutoring Systems* (2006), 278-287.

[4] D. Roy, S Sarkar, S. Ghose, Automatic Extraction of Pedagogic Metadata from Learning Content, *Int. J. of Artificial Intelligence in Education* **18** (2008), 287-314.

[5] J. Jovanovic, D. Gasevic, V. Devedzic, Ontology-Based Automatic Annotation of Learning Content, *Int. J. on Semantic Web and Information Systems*, **2(2)** (2006), 91-119.

[6] B. Jong, T. Chan, Y. Wu, Learning Log Explorer in E-Learning Diagnosis, *IEEE Transactions on Education* **50(3)** (2007), 216-228.

[7] E. Garcia, C. Romero, S. Ventura, C. Castro, An architecture for making recommendations to courseware authors using association rule mining and collaborative filtering, *User Modeling and User-Adaptive Interaction* (to appear).

[8] E. Kobsa, V. Dimitrova, R. Boyle, Adaptive Feedback Generation to support teachers in web-based distance education, *User Modeling and User-Adapted Interaction* **17** (2007), 379-413.

[9] I. Guyon, A. Elisseeff, An Introduction to Variable and Feature Selection, *Journal of Machine Learning Research* **3** (2003), 1157-1182.

[10] P.A. Flach, N. Lachiche, Confirmation-Guided Discovery of First-Order Rules with Tertius, *Machine Learning* **42** (2001), 61-95.

[11] Ian H. Witten and Eibe Frank "Data Mining: Practical machine learning tools and techniques",2nd Edition, Morgan Kaufmann, San Francisco, 2005.