Using Machine Analysis to Make Elementary Students’ Mathematical Thinking Visible

Kimberle Koile  
MIT ODL  
77 Mass Ave, E34-370  
Cambridge, MA 02139  
+1-617-253-6037  
kkoile@mit.edu

Andee Rubin  
TERC  
2067 Mass Ave  
Cambridge, MA 02140  
+1-617-873-9720  
andeerubin@terc.edu

Steve Chapman  
StevCode  
stevcode.com  
Cambridge, MA 02139  
marlene_kliman@terc.edu

Marlene Kliman  
TERC  
2067 Mass Ave  
Cambridge, MA 02140  
+1-617-873-9654  
lily_ko@terc.edu

ABSTRACT

The INK-12: Teaching and Learning Using Interactive Ink Inscriptions in K-12 project has been developing and investigating the use of pen-based technology in elementary math classes. This paper reports on progress made on machine analysis of students’ visual representations created using digital tools developed to support learning multiplication and division. The goal of the analysis is to make student thinking visible in order to (a) better understand how students learn multiplication and division, and (b) provide feedback to teachers, e.g., about strategies students use to solve problems. Student work from a five-week trial in a third grade class provides a corpus for development and evaluation of the machine analysis routines. Preliminary findings indicate that the routines can reproduce human analyses.

Categories and Subject Descriptors
H.1.2 I.1.1 [Artificial Intelligence]: Knowledge Representation Frameworks and Methods; K.3.1 [Computers and Education]: Computer Uses in Education.

Keywords
Visual representations, pen-based computing, learning analytics, elementary education, mathematics, formative assessment

1. INTRODUCTION

Students are often asked to show their thinking when presented with a mathematical problem to solve. In the elementary grades, drawing visual representations is an important method for expressing mathematical thinking [1, 7]. Such representations give students the opportunity to be creative and to choose their own strategies. Indeed, the National Council of Teachers of Mathematics (NCTM) recommends that students “create and use representations to organize, record and communicate mathematical ideas” and “select, apply and translate among mathematical representations to solve problems” [5, p. 67]. Classroom conversations around representations provide valuable opportunities for feedback to both teachers and students.

The INK-12: Teaching and Learning Using Interactive Ink Inscriptions in K-12 project (ink-12.mit.edu) has been investigating upper elementary students’ uses of visual representations in multiplication and division [3, 4].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/authors. Copyright is held by the owner/authors.

Using our tablet-based software, called Classroom Learning Partner (CLP), students use a tablet pen to create representations in an electronic notebook and wirelessly submit their “page” of work to the teacher. The teacher can view student work in real time, identifying students who may need help and choosing pedagogically interesting examples for class discussion. But how does a teacher make these decisions, especially when presented with submissions from each of 20 or more students? One of our goals is to develop machine analysis routines that will provide a teacher with information that will enable her to make these decisions easily and quickly. The analysis routines are also aimed at increasing our understanding of how students learn multiplication and division, helping to find answers to such questions as: How do students use and modify representations? What do use patterns reveal about student thinking?

2. ENABLING MACHINE ANALYSIS

CLP’s machine analysis routines provide information about a representation and the process by which the representation was created. As noted by NCTM, “representation refers both to process and to product” [5, p. 67]. There are three key ideas that enable machine analysis of students’ visual representations: digital tools that enable students to draw, but that also produce structured objects that a machine can recognize; storage of objects and the history of a student’s interactions with the object; and a coding scheme for creation and interaction with representations (history codes) and for observations about the representations (analysis codes). Shown in Figure 1 is a third grade student’s number line representation for 48 ÷ 8, the history code for creating the representation, and an analysis code for representation correctness, i.e., whether the representation matches the math in the problem. The history code represents the actions of adding a number line of length 48 to the page, then drawing jumps of size 8 from 0 to 48 along the number line. The analysis code is the result of comparing the number line jumps—6 jumps of size 8 ending at 48—against an answer definition that specifies 6 groups of 8.

Figure 1. Number line, history and analysis codes
3. HUMAN CODING

The coding scheme that forms the basis of our machine analysis was developed over several months by the authors, two of whom are math educators, using grounded theory. The student work used in coding was the final assessment for a five-week multiplication and division unit in a third grade class and consisted of 21 students’ work on 12 problems. The coding process involved replaying interaction histories, identifying salient features, then creating terms that captured the features. The resulting scheme contains 20 history codes for five types of representations and 28 analysis codes. A subset of the scheme is shown in Table 1. The scheme was used to code all the student work, with a subset of work coded by all authors in order to ensure inter-rater reliability.

Table 1. Example history and analysis codes

<table>
<thead>
<tr>
<th>Type</th>
<th>Meaning</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>History</td>
<td>Create a number line and jump by # from # to #</td>
<td>NL [24: 4, 0-24]</td>
</tr>
<tr>
<td>History</td>
<td>Create an array of dimension # x #</td>
<td>ARR [6x7]</td>
</tr>
<tr>
<td>History</td>
<td>Draw a group of bins</td>
<td>BINS [8]</td>
</tr>
<tr>
<td>Analysis</td>
<td>Fill in answer before creating representation</td>
<td>ABR</td>
</tr>
</tbody>
</table>

History codes correspond only to actions deemed relevant to mathematical thinking; e.g., there is no code for resizing a representation. Analysis codes correspond to observations about a representation’s product or process, e.g., filling in an answer before creating a representation, which may imply the representation to check or explain work, rather than in problem solving.

4. MACHINE ANALYSIS

Our machine analysis routines aim to automate human encoding of representations. They take as input an interaction history—a sequence of low-level actions such as adding an ink stroke or an object to a “page”—and make multiple passes over the history: identifying characteristics of objects, clustering ink strokes via machine learning techniques, using handwriting recognition and heuristics to add semantics to ink clusters, and combining actions into more abstract actions that resemble human history codes. The result is a sequence of semantic events, similar to TD indicators in [2] and semantic events in [6]. The sequence describes the process of creating a representation and, as a last step, is analyzed in order to recognize salient use patterns and apply relevant analysis codes.

The interaction history for the number line in Figure 1 contains 67 low-level actions, starting with a dot at 0 and a jump from 0-8, then dot at 16, jump 8-16, dot at 24, jump 16-24, 10 ink strokes that are identified as arithmetic, then dot at 32 and jump 24-32. The arithmetic and dot-jump pattern continues until 40, indicating that the student knows her 8 times table up to 24, but must use arithmetic for larger numbers. Her last jump from 40-48 indicates that she knows the jumps must end on the dividend. Two jumps of sizes 7 and 9 are erased and replaced with jumps of size 8, indicating that she knows that the jumps must be the same size.

The entire interaction history for this example contains 341 actions, and playback and analysis reveal something extremely interesting: The student created two different representations and erased them prior to creating the number line. In fact, she did not use the number line to compute the answer; instead she drew a series of circles representing bins and used a strategy called dealing out, in which a mark is added to each bin until the number of marks matches the total, in this case 48. She then wrote the answer, and created a 6 by 8 array with skip counting along the bottom, presumably to check her work. Her bin and array representations are shown in Figure 2. She then erased everything, rewrote the equation without an answer, created the number line, and filled in the answer again. A teacher does not have time to replay students’ work, but machine analysis routines can give her valuable information not evident in the final work, in this case which strategies the student tried before settling on a number line.

![Figure 2. Additional representations and human encoding](Image)

5. CURRENT WORK

We are continuing human analysis of our current corpus and extending our machine analysis routines in order to recognize additional patterns, especially those that involve ink strokes. In Figure 2, for example, the current routines recognize skip counting because of the alignment of ink strokes with array columns. Recognizing the equation or the bins is challenging, but clustering of strokes and shape recognition routines hold promise. Robust machine analysis routines will enable us to move beyond time-consuming human analysis so that thousands of pieces of work created during several five-week trials can be analyzed, helping to further our knowledge about how students learn and how student mathematical thinking can be made visible.

6. ACKNOWLEDGMENTS

Research funding is via NSF DRL-1020152, DRL-1019841.

7. REFERENCES


