NeuroImage 97 (2014) 163–177

Contents lists available at ScienceDirect

NeuroImage

journal homepage: www.elsevier.com/locate/ynimg

Discovering the structure of mathematical problem solving

John R. Anderson [a,1], Hee Seung Lee [b], Jon M. Fincham [a]

[a] Department of Psychology, Carnegie Mellon University, USA
[b] Department of Education, Yonsei University, Republic of Korea

ARTICLE INFO

Article history:
Accepted 8 April 2014
Available online 16 April 2014

Keywords:
fMRI
Hidden Markov model
Mathematical problem solving
Multivariate pattern analysis

ABSTRACT

The goal of this research is to discover the stages of mathematical problem solving, the factors that influence the duration of these stages, and how these stages are related to the learning of a new mathematical competence. Using a combination of multivariate pattern analysis (MVPA) and hidden Markov models (HMM), we found that participants went through 5 major phases in solving a class of problems: A Define Phase where they identified the problem to be solved, an Encode Phase where they encoded the needed information, a Compute Phase where they performed the necessary arithmetic calculations, a Transform Phase where they performed any mathematical transformations, and a Respond Phase where they entered an answer. The Define Phase is characterized by activity in visual attention and default network regions, the Encode Phase by activity in visual regions, the Compute Phase by activity in regions active in mathematical tasks, the Transform Phase by activity in mathematical and response regions, and the Respond phase by activity in motor regions. The duration of the Compute and Transform Phases were the only ones that varied with condition. Two features distinguished the mastery trials on which participants came to understand a new problem type. First, the duration of late phases of the problem solution increased. Second, there was increased activation in the rostrolateral prefrontal cortex (RLPFC) and angular gyrus (AG), regions associated with metacognition. This indicates the importance of reflection to successful learning.

Introduction

The past decade has seen a considerable growth in the understanding of the neural basis of certain aspects of mathematics. The greatest amount of research has gone into understanding the role of various parietal regions in basic arithmetic tasks and their role in normal and abnormal development (e.g., Ansari and Dhital, 2006; Arsalidou and Taylor, 2011; Butterworth et al., 2011; Castelli et al., 2006; Molko et al., 2003). Dehaene's (1997) triple-code theory identifies three regions as critical to the representation of number: the horizontal intraparietal sulcus that processes numerical quantity, the angular gyrus that is involved in the verbal processing of numbers, and the fusiform gyrus that processes number form. In addition, the prefrontal cortex is particularly involved in more advanced tasks involving topics like algebra, geometry, or calculus (e.g., Krueger et al., 2008; Qin et al., 2004). One prefrontal region of interest is the lateral inferior prefrontal cortex that is involved in retrieval of arithmetic facts and semantic facts (Danker and Anderson, 2007; Dehaene et al., 1999; Menon et al., 2000). More dorsal and more anterior prefrontal regions become engaged as the problem solving gets more complicated (Wintermute et al., 2012).

Most of this past research has looked at the execution of well-established procedures. The current research investigated how mathematical knowledge becomes "alive" and extends to solving novel problems. We taught participants a new mathematical skill (which is really just equation-solving in disguise) and then challenged them to extend what they had learned to novel transfer problems. In order to identify when the key cognitive events occurred we needed to develop new methods that deal with the variability in complex mathematical problem solving.

A complex skill like algebra problem solving involves a rich mixture of perceptual, cognitive, and motor activities. For instance, when manipulating an equation in traditional paper and pencil mode, a student has to scan past lines of equations, identify the next critical transformation, determine what the new equation will be, and then write that equation. In more modern computer interfaces and tutoring systems, handwriting can be removed but there still are the same basic steps with computer gestures replacing handwriting. This complexity and mixture of activities makes it difficult to identify when the critical cognitive events are taking place. This paper will show that it is possible to analyze individual trials and identify the critical events by combining multivariate pattern analysis (MVPA—e.g., Norman et al., 2006; Pereira et al., 2009) and Hidden Markov Model (HMM) algorithms (Rabiner, 1989). The MVPA recognizes the mental states and the HMM recognizes the sequence of states.

E-mail address: ja+@cmu.edu (J.R. Anderson).
[1] Department of Psychology, Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA 15213, USA.

http://dx.doi.org/10.1016/j.neuroimage.2014.04.031
1053-8119/

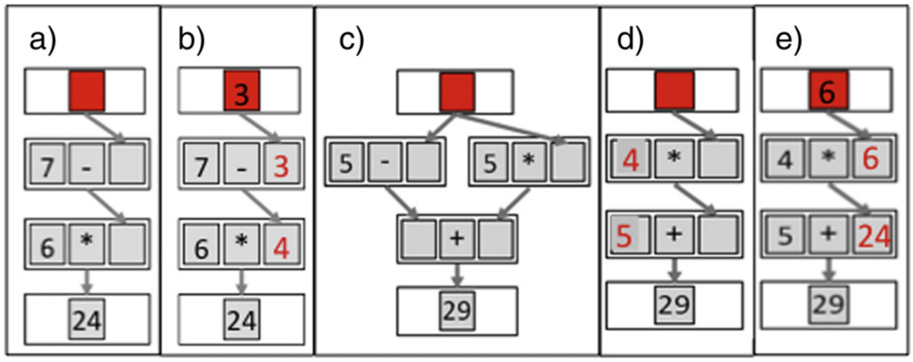

Fig. 1. Data flow graphs where an unknown number flows down from the top box. Red denotes numbers added to the diagram by the participant. (a) A simple Propagate problem equivalent to $6 * (7 - x) = 24$; (b) The solution for (a) where 4 is entered because $6 * 4 = 24$ and 3 is entered because $7 - 3 = 4$; (c) A Linearize problem, equivalent to $(5 - x) + (5 * x) = 29$, with two paths which must be converted into Propagate form; (d) The Propagate equivalent of (c) since $5 - x + 5x = 4x + 5$; (e) The solution for (d) since $5 + 24 = 29$ and $4 * 6 = 24$.

Our past work (Anderson et al., 2010, 2012a, 2012b) trained an HMM and an MVPA classifier on the states with one set of data and then tested it with another set. This required that the states be known in advance and marked in the training set. More recently, Anderson and Fincham (2013) showed that this approach could be extended to discover what the states were solely based on the data, without needing the state structure to be specified in advance. Their demonstration involved using only imaging data, but the current paper will show that this approach becomes more powerful when the imaging data are merged with behavioral data. These methods can analyze problem-solving episodes that involve up to 60 s of mixed activities and identify the few key moments in the episodes where the most critical cognitive events are happening.

This paper is divided into four parts. Part 1 describes an imaging experiment studying how participants learn new mathematical problem solving skills. Part 2 explains the MVPA–HMM method and describes the states that it discovers for this experiment. Part 3 uses the inferred states to gain a deeper understanding of task performance at both behavioral and neural levels. Finally, Part 4 interprets our results and their implications.

A study of mathematical learning and transfer

We have developed a data-flow isomorph of school algebra that has allowed us to study college students learning algebra all over again (Brunstein et al., 2009; Lee et al., 2011). Because it is a laboratory invention and not a real mathematics topic, we have been free to explore a range of instructional variation that might not be appropriate or ethical for students learning real algebra. The data for this paper come from an experiment (Lee et al., in press[2]) that involved a contrast between learning by discovery and learning by direct instruction. Participants learned how to solve the problems in one session outside the scanner and then had to transfer this knowledge to solving new, challenging problems in the scanner. Discovery participants were somewhat slower in mastering the material in the learning session but there were no differences (behavioral or imaging) between instruction and discovery participants in the transfer session. Lee et al. analyzed the instructional effects in learning and their disappearance in transfer. Here we are interested in analyzing the common processes by which participants approached these transfer problems and will pool the data from the different instructional groups.

Fig. 1 shows examples of the data-flow structures that we used. They consist of a set of boxes containing tiles with numbers or operators. Arrows connect boxes to tiles. In data-flow representations a number flows from a top box through a set of arithmetic operations to a bottom box. If that number is unknown, the data-flow structure is equivalent to an algebraic equation with a single variable. For instance, Fig. 1a is the data-flow equivalent of $6 * (7 - x) = 24$. The task is to determine what values to fill into the empty tiles in the boxes. For a linear structure like Fig. 1a, the values can be determined by simply “propagating” the number up from the bottom and performing the arithmetic operations. The solution (as illustrated in Fig. 1b) involves placing 4 in the empty tile above the bottom box (since $6 * 4 = 24$), then placing 3 in the empty tile above it (because $7 - 3 = 4$), and finally placing 3 in the top box (equivalent to solving as $x = 3$). Most participants find solving these problems by this propagation strategy easy and intuitive (one participant described it as similar to playing Sudoku). However, when problems cannot be solved by this simple propagation strategy, participants tend to have difficulty understanding the problem structure and figuring out a procedure for solving the problem.

One class of difficult problems involves the unknown value flowing down multiple paths. Fig. 1c illustrates a simple case of such a problem, which is equivalent to solving an equation with multiple appearances of the variable. The diagram in Fig. 1c is equivalent to the algebraic expression, $(5 - x) + (5 * x) = 29$. In the diagram an unknown value flows down into the two tiles in a box below, which are summed to produce a result of 29. Because two paths converge in a single result, the propagation strategy does not work. The way to solve this problem within the rules of the system is to transform the graph in Fig. 1c into the linear form in Fig. 1d (equivalent to $4x + 5 = 29$), where this simple propagation procedure is possible again as illustrated in Fig. 1e. This transformation step, called *linearization*, is a major conceptual hurdle in this artificial curriculum. It corresponds to collection of variables and constants in regular algebra, which in combination with distribution causes some difficulty when regular algebra is taught in school.

The most difficult step in a Linearize problem is determining the values to enter into the linearized form—for instance, the 4 and the 5 in Fig. 1d. Participants in this experiment had spent the first day, outside of the scanner, mastering this linearization step on relatively simple problems like Fig. 1c. On the second day, they went into the scanner and solved Linearize problems that posed new challenges. Fig. 2 shows two examples of such challenging problems. They would see the problem on the left with the multiple boxes highlighted that had been replaced by a linear structure on the right. Their task was to enter into the two blue tiles the numbers that would make the left and right structures equivalent.

The major experimental manipulation in the transfer section involved the type of problem participants were asked to solve. The problems were either

1. Graphic problems: These involved more complex graph structures than participants had solved up until this point. Fig. 2a illustrates a

[2] This paper and entire experimental materials are available at https://www.dropbox.com/sh/bya83pfytbixzzf/OLMbG0OVX4.