2016-12-01

Task-Level Feedback in Interactive Learning Enivonments Using a Rules Based Grading Engine

John Shadrack Chapman
Brigham Young University

Task-Level Feedback in Interactive Learning Environments

Using a Rules Based Grading Engine

John Shadrack Chapman

A dissertation submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Randall Davies, Chair
Royce Kimmons
Gove Allen
Ross Larsen
Peter J. Rich

Department of Instructional Psychology and Technology

Brigham Young University

ABSTRACT

Task-Level Feedback in Interactive Learning Environments
Using a Rules Based Grading Engine

John Shadrack Chapman
Department of Instructional Psychology & Technology, BYU
Doctor of Philosophy

In order to improve the feedback an intelligent tutoring system provides, the grading engine needs to do more than simply indicate whether a student gives a correct answer or not. Good feedback must provide actionable information with diagnostic value. This means the grading system must be able to determine what knowledge gap or misconception may have caused the student to answer a question incorrectly. This research evaluated the quality of a rules-based grading engine in an automated online homework system by comparing grading engine scores with manually graded scores. The research sought to improve the grading engine by assessing student understanding using knowledge component research. Comparing both the current student scores and the new student scores with the manually graded scores led us to believe the grading engine rules were improved. By better aligning grading engine rules with requisite knowledge components and making revisions to task instructions the quality of the feedback provided would likely be enhanced.

Keywords: knowledge components, diagnostic instructional feedback, data mining

ACKNOWLEDGEMENTS

I would like to thank many people who have helped make this dissertation a reality. First on the list is my patient and caring wife, Kathryn. Her vision of degree completion was never dim nor was it far away. Thank you! And to my children (Josh, Anna, Elisa, Rebekah, Sarah, Sophia, Moriah, Sharon, and Julia) some of whom do not know a life without their father "going to school," thank you for the motivation to continue and for the many reassuring hugs and smiles after long days. To my parents, I thank you for your empathy and support to say nothing of the countless hours of conversation, hopeful encouragement, and insightful ideas. A special thank you to my siblings work group whose influence was felt far beyond the limits of our weekly meeting. A true work group without politics or agendas, dedicated solely to constructive, even innovative comments and suggestions. It was a pleasure, and a challenge to be yoked together on our different paths to similar goals. To my extended family, thank you for your understanding and support. Your hope in me and in this process has not gone unnoticed.

I would especially like to thank my committee with whom I quite enjoyed our interaction and work. Your comments and concerns were expressed professionally, freely, and productively. Your example is one I hope to emulate in my future professional work. I would like to mention Dr. Randy Davies specifically for his constant and encouraging guidance over the past few years in my research projects and dissertation. It has been an honor to have you as my chair. Dr. Gove Allen, thank you for your open door, through which I have passed many times, and your data. But thank you mostly for your generosity in time and thought. I am also grateful for the many wonderful associations with other students in classes and research projects. Thank you for sharing your lives, hopes, ambitions, consternations, and much more. Michele Bray, thank you for you! The department is lucky to have you.

TABLE OF CONTENTS

DESCRIPTION OF RESEARCH AGENDA AND THE DISSERATION STRUCTURE

Technology is playing an ever-increasing role in the design, implementation, and assessment of instruction (Davies & West, 2014). Computer Assisted Instruction (CAI), Intelligent Tutoring Systems (ITS), Adaptive Hypermedia Systems (AHS), Educational Data Mining (EDM), Learning Analytics (LA) and other forms of technology have introduced new possibilities for teaching and learning (Brusilovsky, 2012; Koedinger, Corbett, & Perfetti, 2012; Vandewaetere & Clarebout, 2014; Wenger, 1987). Part of the justification for the development of these possibilities is the belief that new technology will improve teaching and learning (Pea, 2014; U.S. Department of Education, Office of Educational Technology, 2016). One improvement that continues to receive attention is the potential for technology to help personalize a learning experience for individual students (Bloom, 1984; Brusilovsky, 2012; Shute & Zapata-Rivera, 2008, 2012; Wenger, 1987).

Personalized learning requires that students receive individualized feedback based on the learning interactions that an individual student has with the instructional activities provided. For personalized learning systems to be effective, the feedback provided needs to be more than simply indicating whether a student completed a specific tasks or whether they got a wrong answer on a specific test item. If technology is to be truly beneficial, it must provide quality feedback (i.e., actionable information with diagnostic value for the student). As technological advancements push the capabilities of educational technology, there is an increased potential for technology-enabled instructional systems to provide feedback with greater specificity, which is expected to make a positive impact in teaching and learning. In this regard, education has not yet reached the anticipated potential to improve instruction and learning through the use of technology (Woolf, 2010). With the exception of limited use of assessment data to provide general feedback, most of the current technology-enabled instructional systems provide very little personalized instruction. In fact, the ability to provide automated personalized feedback in real-time through the use of learning analytics is still at a nascent stage (Chung, 2014; Mayer, 2009). This dissertation follows an article format focusing on informing and improving the diagnostic function technology-enabled learning environments to provide accurate, specific, and actionable feedback to learners. The articles in this dissertation examine or support the use of transaction-level (or step level) log data and a knowledge component domain model in the context of an automated grading system to provide accurate and diagnostic feedback.

Article 1: A Framework for Improving the Diagnostic Function in Technology-Enabled Learning Environments

This article reviews the fundamental concepts related to the diagnostic function within technology-enabled learning environments. These foundations are used to form a framework for improving this diagnostic function. This framework is provided to address the diagnostic function within the inner loop of program adaption (VanLehn, 2006). As described above, the inner loop of intelligent tutoring systems provides diagnosis and feedback to students as they are solving a problem. It can diagnose and inform individual steps students take to solve a problem or complete a task. The framework seeks to improve the diagnostic function of inner loop technology-enabled learning environments.

Article 2: Digging Deeper: The Potential of Transaction-Level Data in an Online Instructional System to Diagnose Knowledge Gaps and Inform Diagnostic Feedback and Remediation

The second article in this dissertation examines the potential for an online homework system to use transaction-level log data to identify learning gaps and misconceptions. It stems from a completed doctoral research project. This research examines two different types of data available to diagnose student understanding including both traditional (i.e., correct and incorrect assessment data) as well as transaction-level data (i.e., process level data). By questioning the assumption that students with the same final answer understand the same underlying concepts, this study asked what diagnostic value transaction-level data might have in improving feedback. Results indicate that while the majority of students with the same final answer may understand the same underlying concepts, many students likely do not. This research identified the need to assess student understanding at a finer level than the correct-or-incorrect, traditional individual assessment level.

Article 3: Improving the Accuracy of an Automated Grading System

The third article of this dissertation builds on a completed pilot study that compared the results and feedback produced by the automated grading engine of an online course with manual scoring of four tasks assigned in an online homework system. This study completes the research by applying potential modifications to the grading engine rules based on an analysis of the final answer, which was aligned with requisite knowledge components identified as essential for the completion of the task. This research then re-assessed student submissions based on new grading-engine rules. The new scores and feedback were compared with the scores and feedback provided by the existing grading engine rules. Improvement was measured by differences in the new and old scoring compared to the baseline (i.e., manual scoring of the task). The research found that the new scoring method based on a revised grading engine rules aligned better with the manual scoring baseline than the previous grading engine rules. This result was due in part by aligning the rules more closely with knowledge components, which will facilitate better feedback to students.

References

Bloom, B. S. (1984). The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher*, *13*(6), 4–16.

Brusilovsky, P. (2012). Adaptive hypermedia for education and training. In P. J. Durlach & A. M. Lesgold (Eds.), *Adaptive technologies for training and education* (pp. 46–65). New York, NY: Cambridge University Press. Retrieved from http://d-scholarship.pitt.edu/13321/1/ARI_2012.pdf

Chung, G. (2014). Toward the relational management of educational measurement data. *Teachers College Record*, *116*(11), 1-16.

Davies, R. S., & West, R. E. (2014). Technology integration in schools. In J. M. Spector, M. D. Merrill, J. Elen, & M. J. Bishop (Eds.), *Handbook of research on educational communications and technology* (4th ed., pp. 841–853). New York, NY: Springer. Retrieved from http://link.springer.com/chapter/10.1007/978-1-4614-3185-5_68

Koedinger, K. R., Corbett, A. T., & Perfetti, C. (2012). The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive Science*, *36*(5), 757–798. https://doi.org/10.1111/j.1551-6709.2012.01245.x

Mayer, M. (2009, August). *Innovation at google: The physics of data*. [PowerPoint Slides] presented at the PARC Forum, PARC, a Xerox company. Retrieved from http://www.parc.com/event/936/innovation-at-google.html

Pea, R. (2014). *The learning analytics workgroup: A report on building the field of learning analytics for personalized learning at scale*. Stanford, CA: Stanford University. Retrieved from https://ed.stanford.edu/sites/default/files/law_report_complete_09-02-2014.pdf

Shute, V. J., & Zapata-Rivera, D. (2008). Adaptive technologies. In J. M. Spector, M. D. Merrill, J. van Merrienboer, & M. P. Driscoll (Eds.), *Handbook of research on educational communications and technology* (3rd ed., pp. 277–294). New York, NY: Lawrence Earlbaum Associates.

Shute, V. J., & Zapata-Rivera, D. (2012). Adaptive educational systems. In P. J. Durlach & A. M. Lesgold (Eds.), *Adaptive technologies for training and education* (pp. 7–27). New York, NY: Cambridge University Press.

U.S. Department of Education, Office of Educational Technology. (2016). *Future ready learning: Reimagining the role of technology in education*. Washington, DC. Retrieved from http://tech.ed.gov/files/2015/12/NETP16.pdf

Vandewaetere, M., & Clarebout, G. (2014). Advanced technologies for personalized learning, instruction, and performance. In J. M. Spector, M. D. Merrill, J. Elen, & M. J. Bishop (Eds.), *Handbook of research on educational communications and technology* (4th ed., pp. 425–437). New York, NY: Springer. Retrieved from http://link.springer.com/chapter/10.1007/978-1-4614-3185-5_34

Wenger, E. (1987). *Artificial intelligence and tutoring systems: Computational and cognitive approaches to the communication of knowledge*. Los Altos, CA: Morgan Kaufmann Publishers.

Woolf, B. P. (2010). *A roadmap for education technology* (Research Report No. hal-00588291). HAL. Retrieved from https://hal.archives-ouvertes.fr/hal-00588291/

Article 1:

A Framework for Improving the Diagnostic Function in Technology-Enabled Learning Environments

A Framework for Improving the Diagnostic Function in Technology-Enabled Learning Environments

John S. Chapman

Brigham Young University

Abstract

This article reviews the fundamental concepts of the diagnostic function within technology-enabled learning environments. A framework is presented, which combines human-identified knowledge components and transaction-level data analytics to improve the diagnostic function of intelligent tutoring systems. The inner loop of intelligent tutoring systems provides diagnosis and feedback to students as they are solving a problem or completing a task. Improving the diagnostic function will lead to improvements to student learning. The article leverages previous research regarding the domain model and the student model and then describes how human-identified knowledge components combined with transaction-level data analytics improves the diagnostic function in technology-enabled learning environments.

Keywords: diagnostic function, knowledge component, learning, transaction-level data

A Framework for Improving the Diagnostic Function in Technology-Enabled Learning Environments

Recently there has been an emerging need to combine research from instruction and learning with advances in modern technology to provide engaging, relevant, and personalized learning experiences to learners. Various developments support this initiative, including (a) advances in technology both in increased computational power and the development of advanced analytics tools (Baker & Siemens, 2014), (b) an escalation in the ability of technology-enhanced learning systems to capture data (Ferguson, 2012), (c) improvements in online assessment (Marzano, 2009), and (d) a desire to increase the access, efficiency, and cost-effectiveness of education (Fletcher, Tobias, & Wisher, 2007). In addition, Bloom's (1984) widely cited finding that one-on-one tutoring leads to better learning gains than mastery learning or traditional lecture-format methods combined with the growing belief that technology should facilitate differentiated instruction (Benjamin, 2013; Edyburn, 2004) have fueled initiatives moving toward differentiated, adaptive, personalized learning through technology-enabled instruction (Park & Lee, 2004).

Unfortunately, we are far from obtaining this goal. Technology-enabled instruction has the potential to improve the teaching and learning process but is not currently reaching that potential (Woolf, 2010). Christensen, Johnson, and Horne write, "the billions that schools have spent on computers have had little effect on how teachers teach and students learn—save possibly to increase costs and draw resources away from other school priorities" (2010, p. 72). Durlach and Lesgold describe how technology was first applied to education, "Initially, technological approaches replicated classroom methods (mass instruction) and generally provided either no tuning of instruction to individual student needs, simple branching schemes, or mastery approaches in which instruction essentially was repeated until a mastery test was passed" (2012, p. 1). While schools have more technology now than ever before, the educational technologies being used seem to have made relatively little impact on improving teaching and learning in schools (Davies & West, 2014).

The purpose of this article is to review the foundational concepts useful for forming a framework for improving the diagnostic function of technology-enabled learning environments. The foundational concepts chosen include adaptive learning and feedback, the domain model, the student model, knowledge components, and the role of transaction-level data.

Adaptive Learning and Feedback

A related concept to personalized learning is adaptive instruction or adaptive learning. According to Durlach and Tierney, "Adaptive instruction is instruction that can change to suit the needs of individual learners, with the potential to alter aspects like time on task, content, practice examples, and pedagogical strategy" (2012, p. xiii). Adaptive instruction can be enabled by technology and has the potential to produce personalized learning. Shute and Zapata-Rivera wrote, "Adaptive educational systems monitor important learner characteristics and make appropriate adjustments to the instructional milieu to support and enhance learning" (2012, p. 7). More specifically, Brusilovsky (2012) described how technology adapts to an individual. He stated that "A distinctive feature of an adaptive system is an explicit user model that represents user knowledge, goals, and interests, as well as other features that enable the system to adapt to different users with their own specific set of goals" (p. 46). Some of the instructional variables that are adapted might include the type of remediation (e.g., hints, explanations), the timing of feedback (e.g., immediate or delayed), the sequence of content, the degree of scaffolding (e.g., learner support, rewards), and the view of the content (e.g., overview, preview, review, visualization of goals and/or correct answer) (Shute & Zapata-Rivera, 2012).

Intelligent Tutoring Systems (ITS) seek to adapt learning at two levels. VanLehn (2006) calls these levels loops. The outer loop identities the problem to solve, whereas the inner loop identifies the next step a specific student should take to solve a problem. Within the outer loop VanLehn (2006) describes four common types:

1. Student-driven task selection,
2. Tutor-assigned, pre-determined tasks,
3. Mastery learning: Current task must be learned before proceeding to the next task,
4. Macroadaptive learning: Matching task traits and student traits.

The first two outer types do not make a comparison within the tutoring system. In the first type the student may or may not make any comparison between a particular task and an intended goal. In the second, there is also no comparison between a current state and a desired state. However, a comparison is made in both the third and fourth types of outer loops.

In the third, the tutoring system determines if the student's performance matches a mastery level of competence in the task. This mastery level is usually identified as part of a domain model. The domain model represents an expert knowledge performance for that task. The fourth loop the tutoring system "tracks traits, including both unchanging traits such as learning styles, and changing traits, such as correct and incorrect knowledge components. It chooses a task based on the match between the task's traits and the student's traits" (VanLehn, 2006, p. 9). A task trait refers to how the task is communicated to the student. One task can be communicated verbally and another task can be communicated visually. A student trait refers to students' learning preferences or style (VanLehn, 2006).

Domain Model

The domain model outlines the skills needed to perform the tasks for an assignment. Corbett and Anderson wrote:

> Thirty years ago three influential papers … outlined the promise of mastery learning. The core idea is that virtually all students can achieve expertise in a domain if two conditions are met: (1) the domain knowledge is appropriately analyzed into a hierarchy of component skills and (2) learning experiences are structured to ensure that students master prerequisite skills before tackling higher level skills in the hierarchy. (1994, p. 253)

Mastery learning was the name associated with this idea. The intent was to help all students achieve expertise, which means meeting a specific criterion. The two conditions assume that knowledge can be broken down into a hierarchy of component skills (Corbett & Anderson, 1994). This approach to domain knowledge is evident in courses where concepts build on one another.

Student Model

In ITS, AHS, and other systems, a student model is created or derived from the domain model. Holt, Dubs, Jones, & Greer, described how a student model is related to the domain model:

> [T]he student model is conceptualized by comparing the learner's behaviour with that of an expert. This approach assumes that all differences between the learner's behaviour and that of the expert model can be explained as the learner's lack of skill. Therefore, the knowledge of the learner is simply a subset of the expert's knowledge. (1994, p. 6)

Feedback, as Shute (2008) explains, is generally regarded as crucial to improving knowledge and skill acquisition. Feedback is most helpful for student improvement when it is explicit and focused on improving course outcomes (Tanes, Arnold, King, & Remnet, 2011). The feedback relevant to this study is task-level feedback. As opposed to summary feedback:

> Task-level feedback typically provides more specific and timely (often real-time) information to the student about a particular response to a problem or task compared to summary feedback and may additionally take into account the student's current understanding and ability level. (Shute, 2008, p. 154)

Thus task-level feedback, like adaptive learning technologies, takes into account individual student understanding and ability. In order to provide specific, individualized feedback we need diagnostic data that provide specific information about learning.

Knowledge Components

Learning objectives are a common way for educators to state the knowledge, skills, attitudes, and competencies students are expected to acquire when participating in a learning activity (Kuh, Jankowski, Ikenberry, & Kinzie, 2014). Knowledge components, sometimes called threshold concepts (Meyer & Land, 2003), are similar to learning objectives but are more specific in terms of the key aspects of the expected learning. A knowledge component is a mental structure or process that a learner uses, alone or in combination with other knowledge components, to accomplish a task or a problem (Spiro, Feltovich, Jacobson, & Coulson, 1995; VanLehn, 2006). Knowledge components often include domain knowledge (e.g., facts, concepts, principles, rules, procedures) prerequisite to the student completing advanced problems. Corbett and Anderson (1994) argue that the core idea supporting intelligent tutoring is that students can achieve expertise in a domain if two conditions are met: (a) the domain knowledge is appropriately analyzed into a hierarchy of component skills and (b) learning experiences are structured to ensure that students master prerequisite skills before tackling higher level skills in the hierarchy. In both conditions, the central idea is that knowledge can be broken into components. These components, organized into a hierarchy, can then be taught building on previously learned components.

VanLehn (2006) defines a knowledge component in the context of decomposed knowledge. A knowledge component, in this sense, is a part or a piece of a larger knowledge puzzle. When an instructor or instructional designer builds a course, the chunking or dividing up of the course into instructional units is largely done using implicit or explicit knowledge components. In the beginning of the course, the knowledge components take a relatively simple form. As a student progresses through a course of instruction the student must build from and on an increasing foundation of requisite knowledge components as they complete more complex learning tasks.

Diagnosis of learning at the knowledge component level leads to improved assessment, which improves personalized instruction (Chung, 2014). For simple tasks the result of the assessment (i.e., getting a question right or wrong) might adequately indicate a knowledge component has been mastered; but for complex tasks more information might be needed. Instead of indicating a correct or incorrect message, knowledge component assessment data show what parts of the problem the student did not understand or did understand. Selecting and capturing the data regarding specific knowledge components is critical to providing helpful feedback to the student. Focusing solely on final answer assessment-level data that does not provide information indicating how a student arrived at their answer is often of little value to remediate or adapt instruction.

Data Levels

Chung (2014) describes three levels of educational data. At the highest level of aggregation is system-level data. These data may be captured from a student information system (SIS) at a university or school. Examples of systems level data might include courses a student has completed, prior achievement information (i.e., summative course grades), and other basic demographic information. These kinds of data allow institutions to ask questions about student retention rates, graduation rates, and time to degree (Goldstein & Katz, 2005).

A second level of data, the individual level (often referred to as assessment data), includes educational measures comprising information about individual students on specific assignments (e.g., total score on an achievement test, scores on a performance task, or scores on individual items in a test). In general, this level has been the finest grain-size used in educational measurement. Chung takes the position that traditionally, educational measurement has used this individual-level data as the de-facto standard for measuring student understanding both at the item (or problem level) and at the aggregate test level.

Chung goes on to describe a third level of data—transaction-level data. Transaction-level data (also referred to as process-level data) is a "finer" level of data, which dramatically increases the quantity of information it provides, but more importantly it has dramatic implications for diagnosing gaps in a student's understanding or misconceptions the student may have. Chung (2014) wrote:

> [M]ore recently, there has been interest in the use of data at an even finer level of detail and made practical only in technology-based applications … Transaction-level data reflect a student's interaction with a system where the interaction may be an end in itself (e.g., the action a learner performs as part of gameplay) or a

> means to an end (e.g., the act of uploading an assignment in a learning management system). (p. 3)

In other words, transaction-level data records the steps a student takes to answer a question, perform a task, or solve a problem. While a tutor may gather transaction-level data by observing a student perform a task, math instructors might gather process-level data when they ask students to show their work; when Chung talks of transaction-level data he refers specifically to the extended set of data that is captured by a technology-enabled instructional system.

Transaction-level data is becoming more common in education, due at least in part to the advancement of learning technologies and their ability to capture these types of data (Romero, Ventura, Pechenizkiy, & Baker, 2010). But not all transaction-level data points are useful in terms of understanding what a student knows and does not know. Chung (2014) suggests that, "a fundamental issue is the technical quality of measures derived from fine-grained data. There has been little empirical research on how to establish the technical quality of such measures and only recently have psychometricians begun to address this issue" (p. 12). Chung adds,

> The development of robust measures will presumably lead to more effective instructional practices and student learning. Whether diagnostic information is culled from gameplay and reported to teachers to help them decide where to allocate instructional resources or used in adaptive technology-based systems to 'sense' when to provide immediate feedback or execute different instructional branching strategies, the availability of high-quality measures will be critical for any precise targeting of instruction. (2014, p. 13)

Chung connects the advances in "robust measures" to improved feedback and instructional branching strategies including the precise targeting of instruction. Finally, Chung makes this observation, “Historically, significant advances in scientific understanding have followed advances in measurement and observation. As the resolving power of an instrument increased, so have gains in the understanding of the phenomena being observed” (2014, p. 2). He believes that increasing the resolution of educational data to include “moment-to-moment choices” has the potential to result in significant improvements in individual learning assessment. His example of the microscope follows this pattern. The microscope made visible what had previously been invisible and because of this new perspective many new advancements were made possible. A common request a tutor makes to a student is to show their work or to verbalize what they are thinking as they complete a task. Additionally, a tutor gains understanding by observing a student’s actions, giving the tutor the advantage of identifying patterns that are invisible to the tutor without a moment-to-moment view. Once patterns are identified, the tutor can begin intervening to correct or validate the student’s actions. This same pattern parallels the moment-to-moment transaction-level data view of technology-based instruction.

Framework

This article reviewed the fundamental concepts related to the diagnostic function within technology-enabled learning environments. These foundations were used to form a framework for improving this diagnostic function. This framework is provided to address the diagnostic function within the inner loop of program adaption (VanLehn, 2006). As described above, the inner loop of intelligent tutoring systems provides diagnosis and feedback to students as they are solving a problem. It can diagnose and inform individual steps students take to solve a problem or complete a task.

The framework seeks to improve the diagnostic function of inner loop technology-enabled learning environments in a two ways. First, the domain model includes human-identified knowledge components recognizable at the transaction level. Second, the environment distinguishes between knowledge components in the final answer or leading to the final answer. In this way the learning environment can accurately assess what the knowledge components the student doesn't know. It is hoped that this combination of human-driven knowledge components combined with transaction-level analytics will inform and improve both the diagnosis of knowledge gaps as well as the accuracy of the feedback. For example, the transaction-level analytics could inform and refine the knowledge components and, the human-identified knowledge components will inform and refine the capture and analysis of the transaction-level data. This framework combines a human role and a system role in the design and improvement of the diagnostic function for technology-enabled learning environments.

The purpose of this article was to present research regarding ITS design to inform improve the design of diagnostic functions for technology-enabled learning environments. This included a review of the domain model, the student model, how these models are used in ITS and how they are central to the diagnosis function of an ITS. In addition, the article reviewed the role feedback plays in ITS designs and the role knowledge components play in that feedback. Finally, the article presented a framework for improving the diagnostic function by connecting human-identified knowledge components and transaction-level data analytics to improve the designs of ITS.

References

Baker, R., & Siemens, G. (2014). Educational data mining and learning analytics. In R. K. Sawyer (Ed.), *The Cambridge handbook of the learning sciences:* (2nd ed., pp. 253–272). New York, NY: Cambridge University Press. Retrieved from https://www.cambridge.org/core/books/the-cambridge-handbook-of-the-learning-sciences/educational-data-mining-and-learning-analytics/D6FED86BC99E3C403209251B6B44D301

Benjamin, A. (2013). *Differentiated instruction using technology: A guide for middle & high school teachers*. New York, NY: Routledge.

Bloom, B. S. (1984). The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher*, *13*(6), 4–16.

Brusilovsky, P. (2012). Adaptive hypermedia for education and training. In P. J. Durlach & A. M. Lesgold (Eds.), *Adaptive technologies for training and education* (pp. 46–65). New York, NY: Cambridge University Press. Retrieved from http://d-scholarship.pitt.edu/13321/1/ARI_2012.pdf

Christensen, C., Johnson, C. W., & Horn, M. B. (2010). *Disrupting class, expanded edition: How disruptive innovation will change the way the world learns* (2nd ed.). New York, NY: McGraw-Hill Education.

Chung, G. (2014). Toward the relational management of educational measurement data. *Teachers College Record*, *116*(11), 1-16.

Corbett, A. T., & Anderson, J. R. (1994). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, *4*(4), 253–278. https://doi.org/10.1007/BF01099821

Davies, R. S., & West, R. E. (2014). Technology integration in schools. In J. M. Spector, M. D. Merrill, J. Elen, & M. J. Bishop (Eds.), *Handbook of research on educational communications and technology* (4th ed., pp. 841–853). New York, NY: Springer. Retrieved from http://link.springer.com/chapter/10.1007/978-1-4614-3185-5_68

Durlach, P. J., & Lesgold, A. M. (2012). Introduction. In P. J. Durlach & A. M. Lesgold (Eds.), *Adaptive technologies for training and education*. New York, NY: Cambridge University Press.

Durlach, P. J., & Tierney, D. (2012). Preface. In P. J. Durlach & A. M. Lesgold (Eds.), *Adaptive technologies for training and education*. New York, NY: Cambridge University Press.

Edyburn, D. (2004). Technology supports for differentiated instruction. *Journal of Special Education Technology*, *19*(2), 60–62.

Ferguson, R. (2012). *The state of learning analytics in 2012: A review and future challenges* (Technical Report No. KMI-12-01). The Open University, UK: Knowledge Media Institute. Retrieved from http://kmi.open.ac.uk/publications/techreport/kmi-12-01

Fletcher, J. D., Tobias, S., & Wisher, R. A. (2007). Learning anytime, anywhere: Advanced distributed learning and the changing face of education. *Educational Researcher*, *36*(2), 96–102.

Goldstein, P. J., & Katz, R. N. (2005). *Academic analytics: The uses of management information and technology in higher education* (No. Volume 8, 2005). Boulder, CO: Educause Center for Applied Research. Retrieved from https://net.educause.edu/ir/library/pdf/ecar_so/ers/ers0508/EKF0508.pdf

Holt, P., Dubs, S., Jones, M., & Greer, J. (1994). The state of student modelling. In J. E. Greer & G. I. McCalla (Eds.), *Student modelling: The key to individualized knowledge-based* *instruction* (pp. 3–35). Springer Berlin Heidelberg. Retrieved from http://link.springer.com/chapter/10.1007/978-3-662-03037-0_1

Kuh, G. D., Jankowski, N., Ikenberry, S. O., & Kinzie, J. (2014). *Knowing what students know and can do: The current state of student learning outcomes assessment in US colleges and universities*. Urbana, IL: National Institute for Learning Outcomes Assessment. Retrieved from http://learningoutcomesassessment.org/documents/2013%20Survey%20Report%20Final.pdf

Marzano, R. J. (2009). Formative versus summative assessments as measures of student learning. In T. J. Kowalski & T. J. Lasley II (Eds.), *Handbook of data-based decision making in education* (pp. 261–271). New York, NY: Routledge.

Meyer, J. H., & Land, R. (2003). Threshold concepts and troublesome knowledge: Linkages to ways of thinking and practising. In C. Rust (Ed.), *Improving student learning – Ten years on*. Oxford, UK: Oxford Centre for Staff and Learning Development. Retrieved from https://www.utwente.nl/ces/vop/archief_nieuwsbrief/afleveringen%20vanaf%20okt%202005/nieuwsbrief_17/land_paper.pdf

Park, O., & Lee, J. (2004). Adaptive instructional systems. In D. H. Jonassen (Ed.), *Handbook of research on educational communications and technology* (2nd ed., pp. 651–684). Mahwah, NJ: Lawrence Earlbaum Associates. Retrieved from http://www.aect.org/edtech/25.pdf

Romero, C., Ventura, S., Pechenizkiy, M., & Baker, R. (2010). Introduction. In *Handbook of educational data mining*. Boca Raton, FL: Taylor & Francis Group.

Shute, V. J. (2008). Focus on formative feedback. *Review of Educational Research*, *78*(1), 153–189. https://doi.org/10.3102/0034654307313795

Shute, V. J., & Zapata-Rivera, D. (2012). Adaptive educational systems. In P. J. Durlach & A. M. Lesgold (Eds.), *Adaptive technologies for training and education* (pp. 7–27). New York, NY: Cambridge University Press.

Spiro, R. J., Feltovich, P. J., Jacobson, M. J., & Coulson, R. L. (1995). Cognitive flexibility, constructivism, and hypertext: Random access instruction for advanced knowledge acquisition in ill-structured domains. In L. P. Steffe & J. E. Gale (Eds.), *Constructivism in education*. Hillsdale, NJ: Lawrence Earlbaum.

Tanes, Z., Arnold, K. E., King, A. S., & Remnet, M. A. (2011). Using signals for appropriate feedback: Perceptions and practices. *Computers & Education*, *57*(4), 2414–2422. https://doi.org/10.1016/j.compedu.2011.05.016

VanLehn, K. (2006). The behavior of tutoring systems. *International Journal of Artificial Intelligence in Education*, *16*(3), 227–265.

Woolf, B. P. (2010). *A roadmap for education technology* (Research Report No. hal-00588291). HAL. Retrieved from https://hal.archives-ouvertes.fr/hal-00588291/

Article 2:

Digging Deeper: The Potential of Transaction-Level Data in an Online Instructional System to Diagnose Knowledge Gaps and Inform Diagnostic Feedback and Remediation

Digging Deeper: The Potential of Transaction-Level Data in an Online Instructional System to Diagnose Knowledge Gaps and Inform Diagnostic Feedback and Remediation

John Chapman

Brigham Young University

Abstract

As technological advances push the capabilities of educational technology, the potential for these advances to make a positive impact in teaching and learning increases; this is especially true with regards to improving the quality of feedback instructional systems provide to learners. In Intelligent Tutoring Systems and Adaptive Learning Systems, feedback is more than indicating whether a student receives a correct answer or not. The feedback must be informed by actionable information. The goal to improve the quality of feedback is contingent on diagnostic assessment. This means answering the *why* question of student learning: Why did the student answer in that way? What understanding might have caused the student to answer in that way? This research reviews two different types of data available to diagnose understanding, both traditional, correct and incorrect assessment data, as well as transaction-level data. By questioning the assumption that students with the same final answer understand the same underlying concepts, this study asked what diagnostic value transaction-level data might have in improving feedback. Results indicate that while the majority of the forty-five hundred, university-level students with the same final answer do understand the same underlying concepts, many do not. The research offers future research possibilities to address this dilemma and to improve the quality of feedback in technology-enabled instructional interfaces.

Keywords: knowledge component, instruction, learning, diagnosis, transaction-level data

Digging Deeper: The Potential of Transaction-level Data in an Online Instructional System to Diagnose Knowledge Gaps and Inform Diagnostic Feedback and Remediation

There is a widely held belief in education research and practice that personalized or differentiated instruction facilitates learning (Keller, 1974; Pea, 2014; Shute & Zapata-Rivera, 2008; Vandewaetere & Clarebout, 2014), and while teachers strive to accomplish this, they often find their ability to provide personalized instruction to be a daunting task (Barrows, 1988). Educational technology is believed to be of help and is often mandated by policy (Davies & West, 2014); however, one of the challenges of educational technology in the 21st century, which is also an opportunity, is to design and implement customized feedback obtained from technology-enabled instructional systems in a way that improves learning.

In classrooms or other face-to-face learning environments, a common way to personalize learning is to use tutors (VanLehn, 2011). The tutor might be a classroom teacher or, more formally, an individual assigned to specific students. The typical tutoring process has the goal of improving learning through personal attention to an individual student's learning needs (Bloom, 1984). To accomplish this goal, a tutor must gather and analyze information about a specific student's performance or knowledge; then, using his or her experience and expertise, the tutor must diagnose any knowledge or performance gaps, suggesting remedial action as warranted (Barrows, 1988; Fox, 1993). If technology is to facilitate or replicate a human tutor, the technology must also gather and analyze information about the student's performance, then diagnose any knowledge or performance gaps (VanLehn, 2006). Information is needed to generate this type of feedback. Sometimes getting and analyzing data is best done by humans. For example, assessing a student's emotional state or assessing whether an answer they provided might be considered creative or logical (Baker, D'Mello, Rodrigo, & Graesser, 2010). However, in many situations technology can facilitate the data gathering process and at times is better at analyzing data than a human being. Technology-enabled instructional systems get data in many ways. Computers are often faster and more efficient than human beings in this regard. The volume of data a technology-facilitated instructional system can produce is enormous (Koedinger et al., 2012). However, data is not information; and actionable information is needed for effective feedback (Bushweller, 2011).

The first step in the learning analytics process is the selection and capture of relevant data (Campbell & Oblinger, 2007). The most common type of data gathered by educators is assessment data. Teachers might also observe and record their impressions of how well students perform a task. These data are typically used to determine students' grades and to provide feedback. However, for assessment data to be useful as feedback, the assessment items must be aligned with specific intended learning outcomes. Feedback is more accurate when the assessment is carefully designed (Cizek, 2010). Unfortunately, most assessment data by itself has limited diagnostic value (Marzano, 2009). One obstacle that technology-enabled instructional systems need to overcome is the selection of data needed to inform a remedial feedback system. This means going beyond correct or incorrect assessment and getting deeper into the data. One deeper level of data is the transaction level. Sometimes referred to as process-level data (Chung, 2014), these data capture the steps a student takes to arrive at a final answer.

This research asked the following questions. What is the diagnostic value of transaction-level data compared to the traditional, final answer assessment-level data used to evaluate student understanding? And, to what degree do students with the same final answer understand the same underlying concepts?

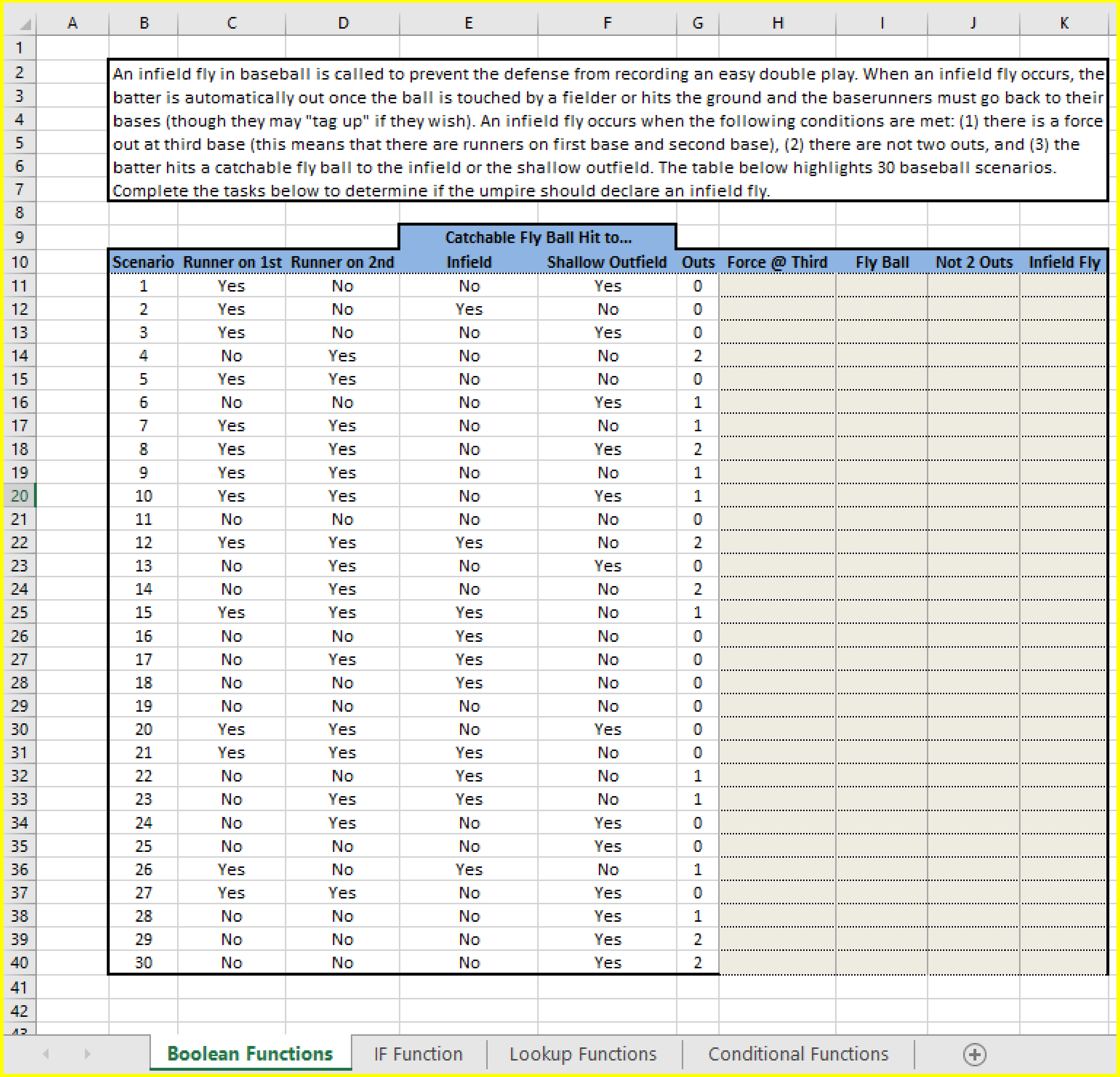
Method

This section addresses how the questions were answered including how the data were collected, how the data were sorted and sampled, and how the data were analyzed.

Data Collection

The data used for this study includes end of semester extant data collected from an introductory spreadsheet course offered at three universities in the United States. These universities each use this course and each provided a sufficient representative source to analyze the data. For each assignment, students were required to complete a task by adding data and formulas to a spreadsheet workbook. As the student completes the task, the instructional system creates a detailed log of each step a student takes. The system that builds and maintains these hidden logs is called the “hidden event log for individual observation system” or HELIOS. Logged data can be aggregated into a single sheet for analysis. The tool that aggregates and manages these student logs is called the “activity record evaluation system” or ARES. These two tools, HELIOS and ARES are freely available to professors at accredited institutions of higher education for non-profit, educational use. They come with a grading mechanism that automatically scores students’ work. The score of each task is based on one or more criteria (i.e., rules) designed by the content experts.

The unobtrusive nature of the data collection avoids the potential for distracting learners from performing the task. A further justification for this method is that the possibility of any Hawthorne effect is mitigated. Additionally, learners do not need to rely on their memory to recall what steps they took. This transaction-level data is different from other research that requires third-party observers, retrospective data collection, or self-report. The following is an example of a typical assignment, including how the student completes the assignment, the structure of the collected data, and the results of the grading engine. Figure 1 shows an assignment from the introductory spreadsheet course used for this study.



An infield fly in baseball is called to prevent the defense from recording an easy double play. When an infield fly occurs, the batter is automatically out once the ball is touched by a fielder or hits the ground and the baserunners must go back to their bases (though they may "tag up" if they wish). An infield fly occurs when the following conditions are met: (1) there is a force out at third base (this means that there are runners on first base and second base), (2) there are not two outs, and (3) the batter hits a catchable fly ball to the infield or the shallow outfield. The table below highlights 30 baseball scenarios. Complete the tasks below to determine if the umpire should declare an infield fly.

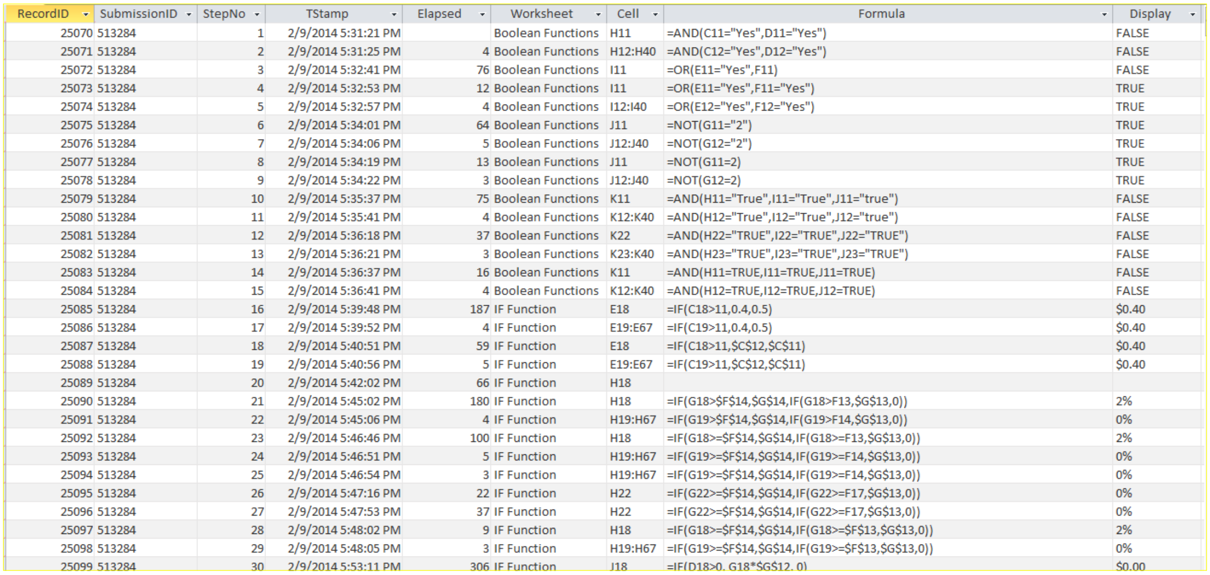
			Catchable Fly Ball Hit to...						
Scenario	Runner on 1st	Runner on 2nd	Infield	Shallow Outfield	Outs	Force @ Third	Fly Ball	Not 2 Outs	Infield Fly
1	Yes	No	No	Yes	0				
2	Yes	No	Yes	No	0				
3	Yes	No	No	Yes	0				
4	No	Yes	No	No	2				
5	Yes	Yes	No	No	0				
6	No	No	No	Yes	1				
7	Yes	Yes	No	No	1				
8	Yes	Yes	No	Yes	2				
9	Yes	Yes	No	No	1				
10	Yes	Yes	No	Yes	1				
11	No	No	No	No	0				
12	Yes	Yes	Yes	No	2				
13	No	Yes	No	Yes	0				
14	No	Yes	No	No	2				
15	Yes	Yes	Yes	No	1				
16	No	No	Yes	No	0				
17	No	Yes	Yes	No	0				
18	No	No	Yes	No	0				
19	No	No	No	No	0				
20	Yes	Yes	No	Yes	0				
21	Yes	Yes	Yes	No	0				
22	No	No	Yes	No	1				
23	No	Yes	Yes	No	1				
24	No	Yes	No	Yes	0				
25	No	No	No	Yes	0				
26	Yes	No	Yes	No	1				
27	Yes	Yes	No	Yes	0				
28	No	No	No	Yes	1				
29	No	No	No	Yes	2				
30	No	No	No	Yes	2				

Figure 1. Screenshot of a task requiring an understanding of Boolean functions. The task asks student to use the information provided to determine whether the automatic out Infield Fly Ball rule applies.

The text at the top in Figure 1 describes the problem scenario of an infield fly in baseball. The purpose of this assignment is to determine whether students can correctly use the AND, NOT, and OR boolean functions. There are three conditions that must each be “true” if the infield fly ball rule is to be called. In column H students are expected to use the AND function to determine whether there is the potential for a force play at third base. In column I students are expected to utilize the OR function to determine whether the situation includes a fly ball in the infield or shallow outfield. In column J students should use the NOT function to determine whether or not there are two outs. The output of each of these three formulas should be a boolean value (True or False). In column K the student must combine the outputs from the results of columns H, I, and J using the AND boolean function to determine whether the infield fly ball rule applies in that situation.

In order to get full points for the task, the student must identify and correctly use various knowledge components, including which data type to use (i.e., text and booleans) as well as which boolean function to use. For example, spreadsheets make a distinction between the words "true" and "false" entered as text into a cell and the Boolean values TRUE and FALSE. While the student might recognize what the correct result of the formula should be, simply typing in the text "true" or "false" into the cell would not be evaluated as correct by the grading engine.

As students enter a potential solution into a cell, the input is captured in the submission log as a unique attempt (see Figure 2). Each line in the log represents an attempt made by the student and is bounded by the enter key or navigating away from the cell. A student can make multiple attempts for each cell. Each attempt is recorded in the submission log with a unique step number. If a student accidentally hits the ENTER key before finishing the formula, this entry is marked as an attempt as well. The data include the step number (StepNo), the date and time of the action (TStamp), the name of the worksheet (Worksheet), the cell location (Cell), the input typed in by the student (Formula), and the resulting display (Display). In the example provided in Figure 2, the data show a student's first input was at cell H11. Then 4 seconds later, the student copied this formula down the column. Seventy-six seconds later, the student input a formula into cell I11 (StepNo 3), but made a correction to the formula 12 seconds later (StepNo 4). These data show the duration of time the student spent on the "Boolean Functions" Worksheet, which is the difference between StepNo 1 and the last step for this worksheet StepNo 15, in this case 5 min and 20 seconds. It also shows the total # of attempts the student made in this worksheet, which was 15. The final attempt (the Formula at StepNo 14) is what is graded by the grading engine when the assignment is submitted.



RecordID	SubmissionID	StepNo	TStamp	Elapsed	Worksheet	Cell	Formula	Display
25070	513284	1	2/9/2014 5:31:21 PM		Boolean Functions	H11	=AND(C11="Yes",D11="Yes")	FALSE
25071	513284	2	2/9/2014 5:31:25 PM	4	Boolean Functions	H12:H40	=AND(C12="Yes",D12="Yes")	FALSE
25072	513284	3	2/9/2014 5:32:41 PM	76	Boolean Functions	I11	=OR(E11="Yes",F11)	FALSE
25073	513284	4	2/9/2014 5:32:53 PM	12	Boolean Functions	I11	=OR(E11="Yes",F11="Yes")	TRUE
25074	513284	5	2/9/2014 5:32:57 PM	4	Boolean Functions	I12:I40	=OR(E12="Yes",F12="Yes")	TRUE
25075	513284	6	2/9/2014 5:34:01 PM	64	Boolean Functions	J11	=NOT(G11="2")	TRUE
25076	513284	7	2/9/2014 5:34:06 PM	5	Boolean Functions	J12:J40	=NOT(G12="2")	TRUE
25077	513284	8	2/9/2014 5:34:19 PM	13	Boolean Functions	J11	=NOT(G11=2)	TRUE
25078	513284	9	2/9/2014 5:34:22 PM	3	Boolean Functions	J12:J40	=NOT(G12=2)	TRUE
25079	513284	10	2/9/2014 5:35:37 PM	75	Boolean Functions	K11	=AND(H11="True",I11="True",J11="true")	FALSE
25080	513284	11	2/9/2014 5:35:41 PM	4	Boolean Functions	K12:K40	=AND(H12="True",I12="True",J12="true")	FALSE
25081	513284	12	2/9/2014 5:36:18 PM	37	Boolean Functions	K22	=AND(H22="TRUE",I22="TRUE",J22="TRUE")	FALSE
25082	513284	13	2/9/2014 5:36:21 PM	3	Boolean Functions	K23:K40	=AND(H23="TRUE",I23="TRUE",J23="TRUE")	FALSE
25083	513284	14	2/9/2014 5:36:37 PM	16	Boolean Functions	K11	=AND(H11=TRUE,I11=TRUE,J11=TRUE)	FALSE
25084	513284	15	2/9/2014 5:36:41 PM	4	Boolean Functions	K12:K40	=AND(H12=TRUE,I12=TRUE,J12=TRUE)	FALSE
25085	513284	16	2/9/2014 5:39:48 PM	187	IF Function	E18	=IF(C18>11,0.4,0.5)	$0.40
25086	513284	17	2/9/2014 5:39:52 PM	4	IF Function	E19:E67	=IF(C19>11,0.4,0.5)	$0.40
25087	513284	18	2/9/2014 5:40:51 PM	59	IF Function	E18	=IF(C18>11,$C$12,$C$11)	$0.40
25088	513284	19	2/9/2014 5:40:56 PM	5	IF Function	E19:E67	=IF(C19>11,$C$12,$C$11)	$0.40
25089	513284	20	2/9/2014 5:42:02 PM	66	IF Function	H18		
25090	513284	21	2/9/2014 5:45:02 PM	180	IF Function	H18	=IF(G18>$F$14,$G$14,IF(G18>F13,$G$13,0))	2%
25091	513284	22	2/9/2014 5:45:06 PM	4	IF Function	H19:H67	=IF(G19>$F$14,$G$14,IF(G19>F14,$G$13,0))	0%
25092	513284	23	2/9/2014 5:46:46 PM	100	IF Function	H18	=IF(G18>=$F$14,$G$14,IF(G18>=F13,$G$13,0))	2%
25093	513284	24	2/9/2014 5:46:51 PM	5	IF Function	H19:H67	=IF(G19>=$F$14,$G$14,IF(G19>=F14,$G$13,0))	0%
25094	513284	25	2/9/2014 5:46:54 PM	3	IF Function	H19:H67	=IF(G19>=$F$14,$G$14,IF(G19>=F14,$G$13,0))	0%
25095	513284	26	2/9/2014 5:47:16 PM	22	IF Function	H22	=IF(G22>=$F$14,$G$14,IF(G22>=F17,$G$13,0))	0%
25096	513284	27	2/9/2014 5:47:53 PM	37	IF Function	H22	=IF(G22>=$F$14,$G$14,IF(G22>=F17,$G$13,0))	0%
25097	513284	28	2/9/2014 5:48:02 PM	9	IF Function	H18	=IF(G18>=$F$14,$G$14,IF(G18>=$F$13,$G$13,0))	2%
25098	513284	29	2/9/2014 5:48:05 PM	3	IF Function	H19:H67	=IF(G19>=$F$14,$G$14,IF(G19>=$F$13,$G$13,0))	0%
25099	513284	30	2/9/2014 5:53:11 PM	306	IF Function	J18	=IF(D18>0, G18*$G$12, 0)	$0.00

Figure 2. Submission Log showing the first 30 steps a student took to complete an assignment.

The grading engine uses rules to evaluate the final attempt for each task. The rules target requisite knowledge components needed to complete the task such as the expected use of the NOT function, the correct reference cell within the function, the correct operator used to compare the reference cell, and the correct value used to evaluate the reference cell. Note that the task could be completed using a formula without the NOT function, i.e., =(G11<2), but this formula would not receive full credit. In this example, four grading engine rules are used to evaluate specific knowledge components. Rule 1 is worth 1 point. It checks to see if the NOT function is used. Rule 2 is worth 2 points. It changes the value of the reference cell, G11, to 2 and re-executes the student-inputted formula given the new value, comparing the new display with the grading engine's expected display. In this case, the student receives the full 2 points for this rule because the student's display from the final attempt (=NOT(G11=2)) agrees with the grading engine's expected display. The student receives 2 points for Rule 3, which changes the reference cell value to 1 and receives 2 points for Rule 4, which changes the reference cell to 0. In the scenario presented in Figure 2, the student received a perfect 7 out of 7 rule points for this task.

It is important to note that while the grading engine is comparing the displayed value, resulting from what the student entered into the content of the cell, with the displayed value once the grading engine makes changes to the requisite input values, it is doing so to test four different knowledge components. This is very different from simply assessing the student's understanding by comparing the final attempt display (in this case, TRUE) to an instructor's answer key. The difference is that this grading engine can evaluate individual knowledge components embedded within the student's final attempt display, not just the students' final display (i.e., final result).

Data Sorting and Sampling

As described above, the submission log contains students' step-by-step and final attempt data. This submission log exists for all assignments across the spreadsheet course. The course contains up to 10 spreadsheet-based assignments. The data used in this analysis focuses on the fundamental concepts found in assignments 2 through 4, and assignment 6. This analysis excludes the first assignment and the fifth, because these assignments do not contain formula-specific, transaction-level data. The first assignment assesses the following skills: deleting columns, formatting of cells and worksheets, basic navigation and file management. The fifth assignment focuses on the creation and management of charts and graphs. The assessment of knowledge components in these assignments is outside the scope of this study but warrants meaningful contemplation at a later time. Lessons 2, 3, 4, and 6 constitute the basis of this research.

For each assignment, a student can submit a solution up to 2 times. When the student submits the solution the first time, the grading engine assesses the student's submission and provides feedback including how many points the student received for rules that indicated correct work, and a basic description of the error if a rule indicated incorrect work. If desired, the student can make changes to their assignment and re-submit the assignment for grading. The student's final grade for the assignment is the average of the two submissions, if submitted twice. If a second submission is not provided, the first submission's grade becomes the student's final assignment grade. This research includes only the first submissions in order to focus on the students' understanding of the concepts before receiving feedback from the grading engine. No student submissions were excluded. However, unique final attempts (i.e., situations where only one unique answer was submitted by students) were omitted from the analysis because standard deviations for these attempts could not be computed as there was no variance. The final data set includes 12,572 submissions, 4,496 students, 4 assignments, 56 tasks, and 164,626 attempts.

Data Analysis

For this analysis we compare individual-level data to transaction-level data. Individual-level data comes from the displayed value of the final attempt for each task and is equivalent to traditional educational measurements used to assess a grade for a task in that all students with the same display value typically receive the same grade for the task. The display value is the resulting value from a formula input into the spreadsheet. Transaction-level data for this study refers to the analytic grading of the content of the cell students submitted as their answer to complete a task. This is based on a multi-rule rubric targeting specific knowledge components.

Individual-level assessment assumes that students with the same display value are at the same level of understanding or in other words by getting a correct final result they understand the same knowledge components. Comparing individual-level assessment with transaction-level assessment provided evidence as to whether this assumption is correct. If it is true that students with the same individual-level assessment understand the same knowledge components, then students with the same display value should receive the same rule score from transaction-level assessment. If individual-level assessments are not the same as the transaction-level rules-based scores, then this suggests that transaction-level assessment is more sensitive in diagnosing student understanding of knowledge components than individual-level assessment data alone.

In order to calculate the number of students with full marks based solely on the display value, the number of submissions with the correct display for the identified task was summed by task. Because the display values were not in the grading engine data, they were extracted from the submission log along with the cell location. These data were then compared for correct display values and totaled. The number of students who received full points on the rule-based grading was identified by comparing each rule score with the number of points possible for each rule. The rule scores with the same number of points as the points possible were totaled by task. Standard deviation of the scores by similar display value were compared to 0.

For example, Table 1 presents counts of the various display values obtained from the Force Out at Third task. The total points possible for this task was 7. This table shows the average rule score per submission for each display and the associated standard deviation. As mentioned above, if students with the same individual-level assessment understand the same knowledge components, then students with the same display value should receive the same rule score from transaction-level assessment. Therefore, the variation or standard deviation of rule scores for students with the same Display should be equal to 0. Any standard deviation above 0 suggests that students with the same display value do not understand the same knowledge components.

Table 1

Counts of Displayed Results Obtained from the Force Out at Third Task

Task Name	Display	Count	Rule Points Possible	Average Score	Standard Deviation
Force Out at Third	TRUE	2684	7	6.89	0.57
Force Out at Third	FALSE	87	7	2.28	1.42
Force Out at Third	Yes	15	7	0.00	0.00
Force Out at Third		9	7	5.44	3.09
Force Out at Third	0	3	7	0.67	0.58
Force Out at Third	#NAME?	2	7	1.00	0.00

Results

To determine the degree to which students with the same displayed answer might be assumed to understand the same underlying knowledge components, we compared scores based on the displayed values alone (i.e., full points for correct display) with the rule based scores of the final solution. The analysis produced 1,286 display values (with more than 1 unique submission) across 56 different tasks. The mean number of possible points for any task evaluated using rule-based, transaction-level assessment was 3.7.

Difference between Scores Based on Display and Rule-Based Scoring

The frequency of non-zero standard deviations is shown in Figure 3. This includes display values that match the expected answer and those that did not. Of the 1,286 display values for 56 tasks, 719 (55.9%) had standard deviations of the rule scores equal to zero. The remaining 567 (44.1%) instances had standard deviations of the rule scores greater than zero. This means that students with the same display value scored differently on the rule scores more than 40% of the time, which indicates that students with the same display value likely do not understand the requisite knowledge components to the same degree.

Every task had at least one display value other than the expected display value, and in each case there was at least one instance where the standard deviation was greater than 0. In other words, it was often the case (with correct and incorrect display values) where it could not be assumed that having the same display values meant that students have the same underlying understanding of the knowledge components needed to accomplish a task.

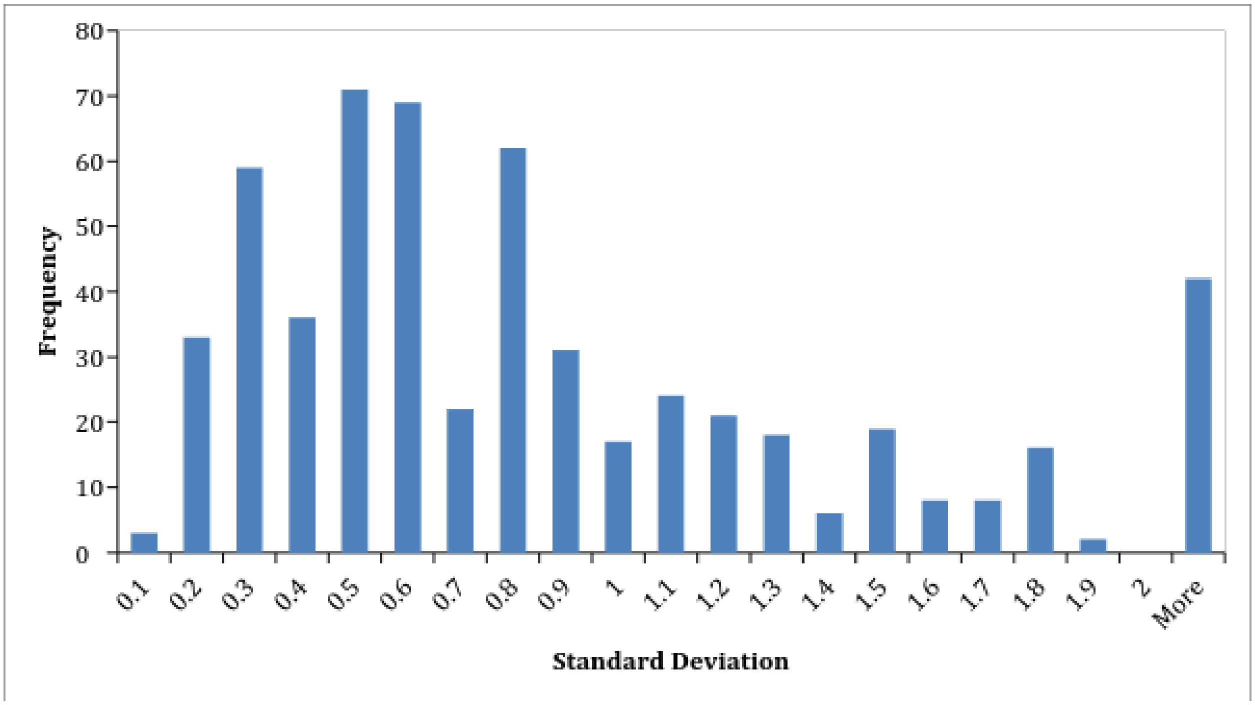


Figure 3. A histogram containing the frequency of non-0 standard deviations from rule based scores.

Results Disaggregated by Task for Correct Displays

On average, submissions within any specific task earned 64.7% of the available points possible. Table 2 presents results disaggregated by task for display values that matched the expected display value as well as submission counts where all rules were correct. In addition, the table contains the number of rules, the average score, and the standard deviation for each of the 56 tasks. It is worth noting that the percentage of submissions with correct displays, in all cases, is greater than the percentage of submissions with all rules correct. This reaffirms the diagnostic potential of rule-based transaction-level scoring over traditional individual-level assessment for all tasks.

Table 3 shows the correlations between select variables from Table 2. Of specific interest is the correlation between the Number of Rules and the Standard Deviation of Task Score, which is 0.50. This indicates that as the number of rules increases, the standard deviation is likely to increase as well. Interestingly, the correlation between the % Correct Display and the Number of Rules is 0.30. This small correlation suggests a mild relationship. And a -0.11 correlation between the Number of Rules and % Correct Rules suggests that as the number of rules in a task increase the % of submissions with all rules correct decreases slightly.

Table 3

Correlation of a Selection of Variables from Table 2

	% Correct Display	*% Correct Rules*	*Number of Rules*	*Rule Points Possible*	*Average of TaskScore*	*StdDev of Task Score*
% Correct Display	1.00					
% Correct Rules	0.59	1.00				
Number of Rules	0.30	-0.11	1.00			
Rule Points Possible	0.18	0.01	0.54	1.00		
Average of TaskScore	0.21	0.16	0.48	0.98	1.00	
StdDev of TaskScore	-0.04	-0.42	0.50	0.71	0.58	1.00

Discussion and Conclusions

Based on the results of 1,286 display values (correct and incorrect), there is convincing evidence that students with the same display value do not understand the same set of requisite knowledge components needed for a specific task. Therefore, it cannot be assumed (as is the case with traditional assessment practices) that students understand the same knowledge components if they produce the same final answer (for this study this means the resulting value displayed given the solution entered into the cell). The result of this research suggests that over 40% of the submissions with the same display value received different rule scores focusing on different knowledge components needed to answer a question correctly. This suggests that students did not have the same understanding of requisite knowledge components. The results support the idea that assessing students using transaction-level data that targets specific knowledge components is a more sensitive assessment of student understanding than individual-level assessment. Significantly, a reason why transaction-level assessment may be more helpful than individual-level assessment data is because knowledge components targeted by the transaction-level assessment can differentiate between specific knowledge components individually rather than as a presumed whole.

Using transaction-level data (rather than the final answer alone) allows the instructor to diagnose specific knowledge gaps and misconceptions. This would be augmented if the system could score the successive attempts students make as they arrive at the final solution they submit. An instructor who can see the different attempts a student makes can dramatically improve and adapt the instruction. Similar to a tutoring session where the tutor sees the process a student uses to complete a task; any issues are more easily identified with transaction-level data than assessment-level data provided by the final answer alone.

In addition, we found a moderate correlation between the number of rules targeted and the standard deviation for the rule based scoring. This suggests a potential to use the Number of Rules as a proxy for task difficulty. Additional research should be considered to further investigate this relationship in part because not every rule aligns cleanly with a single knowledge component.

The use of transaction-level data depends on the instructional system's ability to capture these data and the care instructors take to identify important knowledge components pertinent to the solving of a specific problem or completing a specific task. In this study this translates to the instructional designer creating rules for the grading engine that align with and can be used to evaluate all the important knowledge components associated with each task. For example, in the Force Out at Third task described above, the grading engine has four rules. It does not explicitly evaluate if the student includes an equals sign at the beginning of the function. Yet the equals sign is critical to the proper execution of the function. Without the equals sign the function is not executed; the display shows the function, not the expected result of the function.

In defense of the decision to exclude a rule that targets this specific knowledge component, while this specific issue (i.e., not including an equals sign) is essential to the successful completion of the task, the spreadsheet program in this case displays the content of the cell (the function) as text rather than the resulting value. In other cases the program gives feedback to students who make simple mistakes by issuing an error message in the cell (e.g., “#VALUE!”, or “#NAME?,” error). In almost every case, the missing equals sign mistake is identified promptly by the student and is corrected quickly with no need for instructional intervention. Thus, the relationship between rules and knowledge components is not necessarily one-to-one and can be quite complex. An important design decision associated with the capture of transaction-level is to determine which knowledge components to target. Design decisions related to which knowledge components to target affect the value of the data captured and will depend on aligning the captured data with specific knowledge components deemed essential to complete the task.

Further Research

What is unique about this kind of an assessment is that the assessment digs deeper into the student’s understanding. While the system currently captures transaction-level data at both the process level and the final submitted answer stage, the system does not utilize the data to adapt the system or customize the feedback it gives. The assessment could be more interactive if it was capable of reporting specific knowledge gaps and misconceptions students may have. The design of more interactive assessments that utilize transaction-level data is needed. Such a system would require research into how to implement information reporting and instructional adaptations that would improve the utility and effectiveness of the system. Further research is needed that informs the design of an instructional system that uses student assessment to make the system more intelligent. This would involve identifying the required knowledge components needed to complete each task, carefully aligning rules that target specific knowledge components, reporting knowledge gaps and possible misconceptions, then taking action based on these data to adapt or differentiate the instruction for individual students.

Table 2

Disaggregated Results for Correct Display Results by Task

Task Number	Count of Submissions with correct Display	% Submissions with correct Display	Count of Submissions with all Rules Correct	% Submissions with all Rules Correct	Number of Rules	Average Score	Standard Deviation
1	2431	82.6%	1356	46.1%	6	5.37	2.11
2	2521	95.3%	2298	86.8%	5	2.90	0.40
3	2515	95.8%	2281	86.9%	5	2.90	0.38
4	2296	86.8%	1795	67.9%	5	4.40	1.07
5	2519	93.4%	2450	90.8%	5	4.79	0.84
6	2750	97.8%	2514	89.4%	5	8.69	1.17
7	2709	96.8%	2197	78.5%	5	2.85	0.39
8	2864	83.7%	1679	49.1%	5	3.90	1.36
9	2027	76.5%	1697	64.0%	5	4.86	1.81
10	2473	90.1%	1387	50.5%	5	3.59	1.63
11	2648	96.4%	2562	93.3%	5	5.82	0.82
12	2608	97.3%	2342	87.4%	5	2.88	0.40
13	2595	97.9%	2330	87.9%	5	2.90	0.37
14	2673	98.6%	2297	84.8%	4	2.90	0.30
15	2637	98.1%	2282	84.9%	4	2.90	0.32
16	2090	79.6%	1892	72.0%	4	1.87	0.40
17	2685	95.8%	2573	91.8%	4	6.69	1.17
18	2560	90.4%	2338	82.5%	4	9.01	2.42
19	3424	99.6%	3385	98.5%	4	4.98	0.23
20	3272	95.1%	2294	66.6%	4	5.25	1.21
21	2554	74.3%	2511	73.0%	4	5.70	0.58
22	3365	98.1%	3338	97.3%	4	4.94	0.42
23	1917	66.2%	1041	35.9%	4	3.85	0.81
24	2671	95.4%	2427	86.6%	4	6.57	1.29
25	2436	86.9%	1712	61.1%	4	4.05	1.46
26	3222	94.4%	2867	84.0%	4	5.61	0.96
27	3099	97.9%	2557	80.8%	4	6.65	0.95
28	3410	99.4%	3383	98.6%	4	4.97	0.29
29	2526	96.2%	2323	88.5%	4	2.78	0.68
30	2198	85.4%	2019	78.4%	4	1.81	0.52

(table continues)

Task Number	Count of Submissions with correct Display	% Submissions with correct Display	Count of Submissions with all Rules Correct	% Submissions with all Rules Correct	Number of Rules	Average Score	Standard Deviation
31	2903	91.5%	2839	89.4%	3	3.94	0.36
32	3418	98.8%	3409	98.6%	3	4.96	0.33
33	3373	97.7%	3365	97.5%	3	4.96	0.36
34	2778	93.1%	1067	35.8%	3	2.64	0.89
35	2674	89.6%	1255	42.1%	3	2.31	0.70
36	3272	95.1%	2877	83.6%	3	5.66	1.00
37	2909	84.9%	2890	84.4%	3	5.68	0.80
38	3042	96.2%	3019	95.4%	3	5.92	0.50
39	2575	74.5%	2542	73.5%	3	4.84	0.71
40	2432	70.3%	2412	69.8%	3	4.93	0.44
41	2429	70.3%	2414	69.8%	3	4.96	0.33
42	2249	75.1%	2193	73.2%	2	2.46	0.21
43	2265	77.9%	2212	76.1%	2	2.25	0.68
44	2041	70.1%	1953	67.1%	2	2.39	0.41
45	2812	94.2%	2406	80.6%	2	2.22	0.65
46	2951	98.7%	2841	95.0%	2	2.44	0.31
47	2397	81.9%	2313	79.0%	2	2.33	0.58
48	2471	82.8%	2281	76.5%	2	2.39	0.35
49	2364	80.8%	2275	77.8%	2	2.33	0.58
50	2312	79.0%	2181	74.5%	2	2.36	0.46
51	2229	74.8%	2222	74.6%	2	2.48	0.20
52	2419	80.8%	2403	80.3%	2	2.48	0.20
53	1587	54.6%	1583	54.5%	2	2.24	0.70
54	2311	97.0%	2290	96.1%	1	0.99	0.11
55	2141	88.8%	2109	87.4%	1	0.92	0.27
56	2202	93.8%	2200	93.7%	1	1.99	0.14

References

Baker, R., D'Mello, S. K., Rodrigo, M. M. T., & Graesser, A. C. (2010). Better to be frustrated than bored: The incidence, persistence, and impact of learners' cognitive–affective states during interactions with three different computer-based learning environments. *International Journal of Human-Computer Studies*, *68*(4), 223–241. https://doi.org/10.1016/j.ijhcs.2009.12.003

Barrows, H. S. (1988). *The tutorial process* (Rev. ed.). Springfield, IL: Southern Illinois University.

Bloom, B. S. (1984). The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher*, *13*(6), 4–16.

Bushweller, K. (2011, October 17). Education week: Feedback loops for better schools. Retrieved from http://www.edweek.org/dd/articles/2011/10/19/01editorsnote.h05.html

Campbell, J. P., & Oblinger, D. G. (2007). Top-ten teaching and learning issues, 2007. *Educause Quarterly*, *30*(3), 15–22.

Chung, G. (2014). Toward the relational management of educational measurement data. *Teachers College Record*, *116*(11), 1–16.

Cizek, G. J. (2010). *Translating standards into assessments: The opportunities and challenges of a common core*. Chapel Hill, NC: University of North Carolina at Chapel Hill. Retrieved from https://www.brookings.edu/wp-content/uploads/2012/04/1028_race_to_the_top_cizek_paper.pdf

Davies, R. S., & West, R. E. (2014). Technology integration in schools. In J. M. Spector, M. D. Merrill, J. Elen, & M. J. Bishop (Eds.), *Handbook of research on educational* *communications and technology* (4th ed., pp. 841–853). New York, NY: Springer. Retrieved from http://link.springer.com/chapter/10.1007/978-1-4614-3185-5_68

Fox, B. A. (1993). *The human tutorial dialogue project: Issues in the design of instructional systems*. Mahwah, NJ: Lawrence Erlbaum Associates Inc.

Keller, F. S. (1974). Ten years of personalized instruction. *Teaching of Psychology*, *1*(1), 4–9. https://doi.org/10.1177/009862837400100102

Koedinger, K. R., Corbett, A. T., & Perfetti, C. (2012). The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive Science*, *36*(5), 757–798. https://doi.org/10.1111/j.1551-6709.2012.01245.x

Marzano, R. J. (2009). Formative versus summative assessments as measures of student learning. In T. J. Kowalski & T. J. Lasley II (Eds.), *Handbook of data-based decision making in education* (pp. 261–271). New York, NY: Routledge.

Pea, R. (2014). *The learning analytics workgroup: A report on building the field of learning analytics for personalized learning at scale*. Stanford, CA: Stanford University. Retrieved from https://ed.stanford.edu/sites/default/files/law_report_complete_09-02-2014.pdf

Shute, V. J., & Zapata-Rivera, D. (2008). Adaptive technologies. In J. M. Spector, M. D. Merrill, J. van Merrienboer, & M. P. Driscoll (Eds.), *Handbook of research on educational communications and technology* (3rd ed., pp. 277–294). New York, NY: Lawrence Earlbaum Associates.

Vandewaetere, M., & Clarebout, G. (2014). Advanced technologies for personalized learning, instruction, and performance. In J. M. Spector, M. D. Merrill, J. Elen, & M. J. Bishop (Eds.), *Handbook of research on educational communications and technology* (4th ed., pp. 425–437). New York, NY: Springer. Retrieved from http://link.springer.com/chapter/10.1007/978-1-4614-3185-5_34

VanLehn, K. (2006). The behavior of tutoring systems. *International Journal of Artificial Intelligence in Education*, *16*(3), 227–265.

VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist*, *46*(4), 197–221. https://doi.org/10.1080/00461520.2011.611369

Article 3:

Improving the Accuracy of an Automated Grading System

Improving the Accuracy of an Automated Grading System

John Chapman

Brigham Young University

Abstract

In order to improve the feedback an intelligent tutoring system provides, the grading engine needs to do more than simply indicate whether a student gives a correct answer or not. Good feedback must provide actionable information with diagnostic value. This means the grading system must be able to determine what knowledge gap or misconception may have caused the student to answer a question incorrectly. This research evaluated the quality of a rules-based grading engine in an automated online homework system by comparing grading engine scores with manually graded scores. The research sought to improve the grading engine by assessing student understanding using knowledge component research. Comparing both the current student scores and the new student scores with the manually graded scores led us to believe the grading engine rules were improved. By better aligning grading engine rules with requisite knowledge components and making revisions to task instructions the quality of the feedback provided would likely be enhanced. Common errors were identified that provided evidence of student knowledge gaps. The current grading engine functioned relatively well but the revised grading engine was more accurate enabling better diagnostic feedback.

Keywords: knowledge components, diagnostic instructional feedback, data mining

Improving the Accuracy of an Automated Grading System

Traditional classroom environments continue to transcend common didactic instructional boundaries by incorporating online exercises, assignments, simulations, and projects (Christensen, Johnson, & Horn, 2010). Intelligent Tutoring Systems (ITS), Adaptive Hypermedia Systems (AHS), among other technology-enabled instructional systems have pushed the boundaries of what is possible (Shute & Zapata-Rivera, 2008; VanLehn, 2006). These systems are providing greater visibility into the learning process for learning scientists and instructional researchers interested in improving instruction and learning (Chung, 2014). One technology-driven instructional advancement is personalized learning (Benyon & Murray, 1993; Bloom, 1984; Lewis & Pask, 1965).

Personalized learning is based on specific, individualized feedback to help cater learning experiences to individual learners. In ITS and AHS, feedback is more than indicating whether a student receives a correct answer or not. The feedback must be informed by actionable information. The goal to improve the quality of feedback is contingent on diagnostic assessment. As technological advances push the capabilities of educational technology, the potential for these advances in providing feedback with greater specificity to make a positive impact in teaching and learning increases.

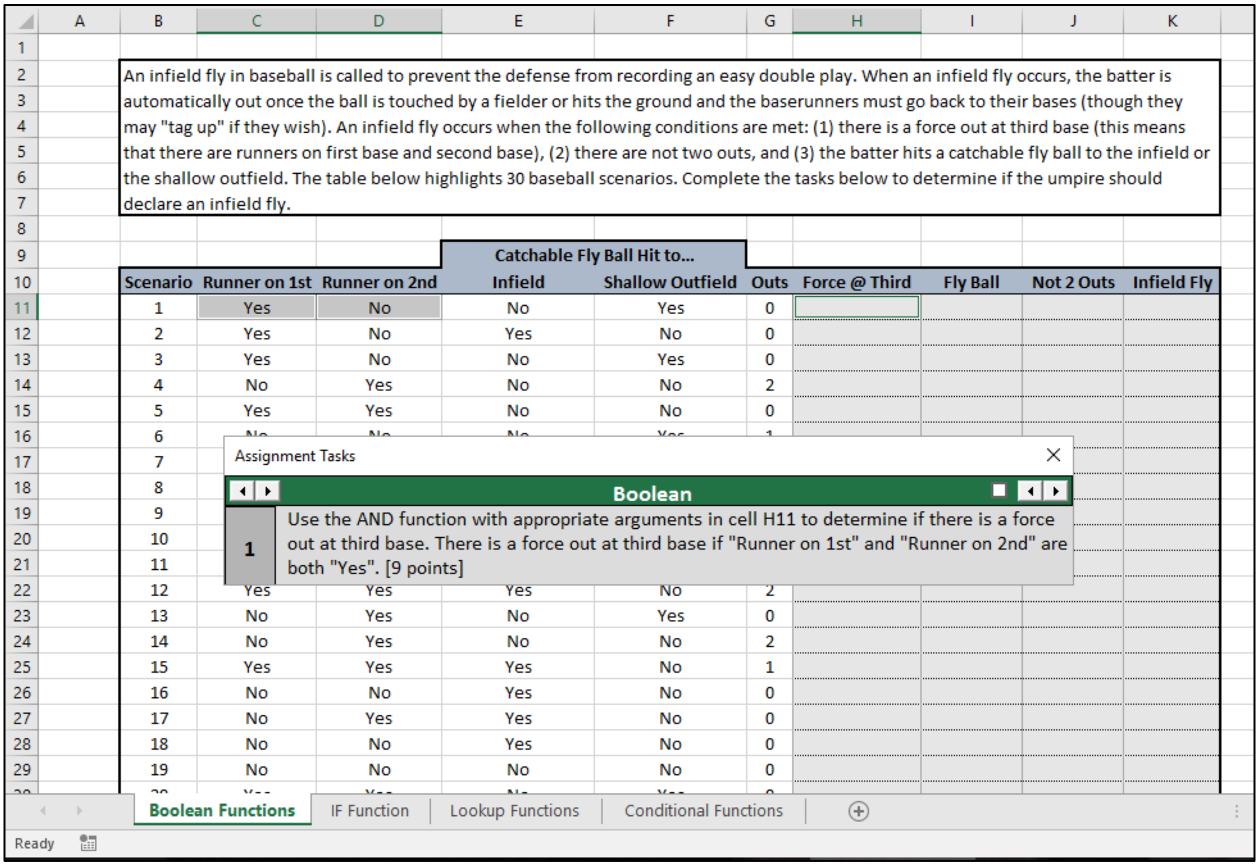
In order to provide specific and individualized feedback to learners, specific and meaningful assessment structures are needed (Harlow, 1959). This study explores the alignment of assessment structures to mirror the fundamental units of skill learners need to solve the required learning activities. Because these fundamental units have often been loosely connected to current automatic grading engines, the feedback received by students has also been loosely connected to their inputs, as described further below. By making the connection between units of knowledge more explicit with assessment structures, feedback is likely to improve (i.e., more specific and personalized).

This study explores the issues of validating and improving the scoring of an automated grading engine used in an online homework system (OHW). This research was conducted to determine the accuracy of the scoring and the quality of the feedback provided. Evaluating performance not only informs the pedagogical theory of the instruction, it can inform and optimize the learning provided (Brooks, Greer, & Gutwin, 2014; White & Larusson, 2010). This study asked the following questions:

1. What common errors are made by students? What variants to the correct and incorrect solution do students provide?
2. To what degree does the current grading engine correctly identify common unique errors (aligned with specific KCs) when compared to manual scoring?
3. To what degree do revised rules for the grading engine more accurately identify errors?

Methods

The data used for this study includes end of semester extant data collected from an introductory online spreadsheet course. The third lesson of the course covers logical arguments. A part of this lesson includes the Boolean Functions assignment, which has four tasks for students to complete—2 AND functions, 1 OR function, and 1 NOT function (see Figure 1). In the first task for the Boolean Functions assignment, the student is asked to use the AND function, in cell H11. The AND function returns the Boolean value TRUE when every argument inside the AND function evaluates to TRUE. If any single argument is FALSE, then the AND function returns FALSE. In cell I11, the student is asked to use the OR function. The OR function returns TRUE if any of the arguments are TRUE. It returns FALSE if all of the arguments are evaluated to FALSE. In cell J11, the student is asked to use the NOT function. The NOT function returns the opposite Boolean value. In cell K11, the student is asked to use the results from H11, I11, and J11 as inputs to the K11 (AND) function. Significantly, not only do the AND, OR, and NOT functions produce output values that are Boolean, but they also require arguments that return Boolean values.



An infield fly in baseball is called to prevent the defense from recording an easy double play. When an infield fly occurs, the batter is automatically out once the ball is touched by a fielder or hits the ground and the baserunners must go back to their bases (though they may "tag up" if they wish). An infield fly occurs when the following conditions are met: (1) there is a force out at third base (this means that there are runners on first base and second base), (2) there are not two outs, and (3) the batter hits a catchable fly ball to the infield or the shallow outfield. The table below highlights 30 baseball scenarios. Complete the tasks below to determine if the umpire should declare an infield fly.

				Catchable Fly Ball Hit to...						
10	Scenario	Runner on 1st	Runner on 2nd	Infield	Shallow Outfield	Outs	Force @ Third	Fly Ball	Not 2 Outs	Infield Fly
11	1	Yes	No	No	Yes	0				
12	2	Yes	No	Yes	No	0				
13	3	Yes	No	No	Yes	0				
14	4	No	Yes	No	No	2				
15	5	Yes	Yes	No	No	0				
16	6	No	No	No	Yes	1				
17	7									
18	8									
19	9									
20	10									
21	11									
22	12	Yes	Yes	Yes	No	2				
23	13	No	Yes	No	Yes	0				
24	14	No	Yes	No	No	2				
25	15	Yes	Yes	Yes	No	1				
26	16	No	No	Yes	No	0				
27	17	No	Yes	Yes	No	0				
28	18	No	No	Yes	No	0				
29	19	No	No	No	No	0				
30	20	Yes	Yes	No	Yes	0				

Assignment Tasks

Boolean

1 Use the AND function with appropriate arguments in cell H11 to determine if there is a force out at third base. There is a force out at third base if "Runner on 1st" and "Runner on 2nd" are both "Yes". [9 points]

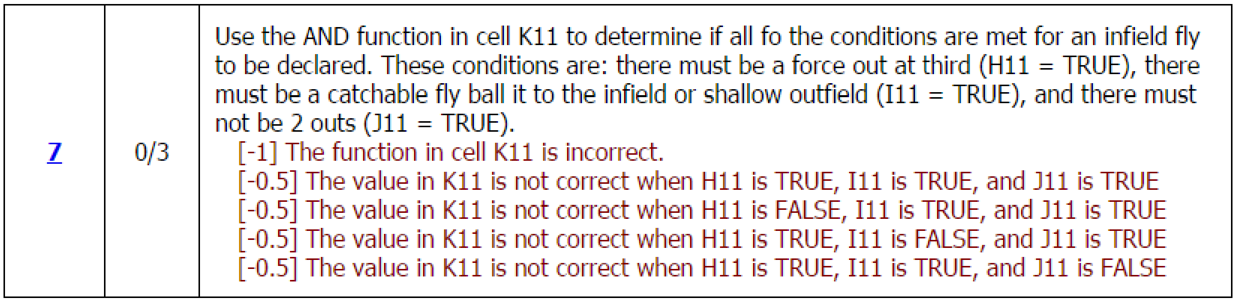
Figure 1. Screenshot of the Force Out at Third task in Microsoft Excel. This is the first task for the Boolean Functions assignment of lesson 3. Students are guided through the task using a pop-up “task guide”.

As shown in Figure 1, cell H11 is selected and the input cells C11 and D11 are highlighted. At the top of the worksheet inside a text box is a description of the context for the four tasks on this assignment. The results of the first three tasks must be completed in order to complete the fourth task, which is to determine if each scenario listed in the table (out of 30 total) should be considered an “Infield Fly.” The first three tasks each correspond to a specific requirement of an Infield Fly. In order for the scenario to truly be an Infield Fly, each of the first three tasks must evaluate to TRUE. If all three tasks evaluate to true, then and only then, is the scenario considered an Infield Fly.

The data in the white columns constitute the inputs to the formulas that will be built in the gray columns. The box in the middle of Figure 1, titled “Assignment Tasks” is a task guide for the student. It outlines the specific instructions for each task including which cell to place the formula and which cells provide inputs to the task. The task shown in the task guide in Figure 1 is the first task suggesting the cell to place the formula (H11) and the cells that should be included in the formula (Runner on 1st and Runner on 2nd). These three cells (H11, C11, and D11) are highlighted by the task guide.

As the student completes a task, the instructional system creates a detailed log of each step. It is common for the student to build the initial solution in the top cell of the column and then copy and paste, or fill, the solution from the first cells into the cells down the column. Because the input cells are not the same for each row, the results column does not contain the same result in each individual calculation. In cell H11, for example, the inputs to the function refer to cells C11 and D11. When the function is copied down the column, the spreadsheet program changes the cell references automatically based on the new row. Instead of using C11 and D11 as inputs to every formula in column H, the inputs change automatically to the corresponding row; that is as long as absolute references were not used. The use of an absolute reference would be an indication the student has a misconception about when to use this feature. After copying the solution down the column, the student can check to see if the resulting display value matches what the student expects to see given the inputs for that row. The problem is designed to allow the student the opportunity to perform a manual check of the function so that the student can compare the actual result with an expected result. If, for example, the actual result of the cell in Column H is TRUE, it means that both cells in columns C and D should also evaluate to TRUE. If another cell in column H is FALSE, it means that at least one input, either C or D, should result to FALSE. When the actual result does not match the expected result, then the student can (a) try to understand why it doesn't work as expected and (b) make changes to the function in order to match the actual and expected values. While not all students check for a match, it is common for a student to make 3 or even 4 iterations of changes before settling on a final attempt and moving on to the next task.

It is also common for a student to use the feedback from the grading engine to understand why the solution does not execute as expected. When a student finishes all tasks in the assignment, the student submits the assignment. The grading engine grades each task, records the scores, and presents to the student an assignment report. The report indicates the score and provides feedback. Figure 2 shows an example of the feedback for 5 incorrect rules in the assignment report. The point values for each rule and the feedback the student receives when the rule is incorrect are then displayed. In this example, the solution the student submitted for cell K11 received 0 out of 3 points because each of the rules failed to detect a satisfactory result. Notice the feedback does not describe how to fix the error or what the student did wrong, only how the error was diagnosed and that the result was not satisfactory.



7	0/3	Use the AND function in cell K11 to determine if all fo the conditions are met for an infield fly to be declared. These conditions are: there must be a force out at third (H11 = TRUE), there must be a catchable fly ball it to the infield or shallow outfield (I11 = TRUE), and there must not be 2 outs (J11 = TRUE). [-1] The function in cell K11 is incorrect. [-0.5] The value in K11 is not correct when H11 is TRUE, I11 is TRUE, and J11 is TRUE [-0.5] The value in K11 is not correct when H11 is FALSE, I11 is TRUE, and J11 is TRUE [-0.5] The value in K11 is not correct when H11 is TRUE, I11 is FALSE, and J11 is TRUE [-0.5] The value in K11 is not correct when H11 is TRUE, I11 is TRUE, and J11 is FALSE

Figure 2. Screenshot of a portion of the Assignment Score after the assignment has been submitted. The text next to the negative digits are examples of the messages provided to students for incorrect rules.

Each rule in the grading engine is designed to test the correctness of the formula. In some rules the grading engine changes the value of the referring cells and examines the output of the formula given the changed inputs. Other rules check for the existence of specific text within the cell. For example, the task in Figure 2 asks the student to use the AND function. The first line of red text is the message associated with the rule that checks for the existence of the AND function within the solution. In this case the grading engine did not find it.

It is important to note that the grading engine is going beyond just testing the final display value of the cell. For example, if instead of using the AND function (as described in Figure 2), the student typed in the Boolean value FALSE into the cell, the display value would be correct. It is not a coincidence that the student can perform the functional equivalent mentally for each individual solution. The challenge is not to perform the function just to put in the correct answer. The challenge is to learn to build the solution using the AND function, and then to use that solution to fill the rest of the column. Instructionally, because the solutions can be solved mentally, students can check the actual output of the solution with their own mental version of the solution. This helps the student identify potential problems in their logic, syntax, etc. before submitting the assignment to be graded by the grading engine. It is useful for instructional designers to consider including this type of functionality into a learning interface.

A unique feature of the spreadsheet platform and the grading engine is the ability to evaluate a solution not just by the resulting display value. Typically, assessment of learning is constrained to the result of the final action of the student. For example, grading the student's performance based on the outcome of the solution not the solution itself. In this scenario, if the student typed FALSE into H11, then the student would receive full credit. However, advanced technological instructional systems have progressed beyond the final display to grade the submitted solution. In this instance, the formula for cell H11 includes the cell references to C11 and D11. The grading engine in this instructional system runs rules to check the correctness of the solution. For this particular task the grading engine runs 5 rules. If the student's solution results in H11 agree with a calculated, expected result then, the student receives full points for that rule. If the results do not agree, then the student does not receive full points for that rule and feedback is presented (after the full assignment has been submitted) to the student to correct any mistake and the student has the option to re-submit the assignment. The final grade for the submission is the sum of the correct rule scores. The final grade for the assignment is the average of the two submissions or if only one submission is made, the assignment grade is grade of the first submission.

From the system logs, a list of all the unique student solutions was extracted and manually scored using 6 specific knowledge components as the basis for the grading (see Table 1). The manual scoring served as a baseline to determine the accuracy of the grading engine. The new scores were then compared to the current grading engine scores and to the manually graded scores. The assumption is that the manual scores are the gold standard against which grading engine scores will be measured. Improvement will be reached if the new grading engine scores are closer to the manual scores.

Table 1

Knowledge Components used in Manual Scoring

Component	Description
KC1	Selects the Correct Boolean Function
KC2	Understands Data Type and when to use them
KC3	Uses Correct Function Syntax
KC4	Constructs a Correct Condition
KC5	Uses Correct Condition Syntax
KC6	No extra syntax

Results

The results presented in this section follow the three research questions outlined above. First, common errors made by students were identified to better understand variants of the correct and incorrect solutions. The second research question examined the degree to which the current grading engine correctly identify common unique errors (aligned with specific knowledge components) when compared to manual scoring. The answer to this question is based on the knowledge components identified as essential to answer the item correctly (see Table 1) and the degree of alignment the current rules have to these components. Also, a comparison is made between the manual grading scores (i.e., the baseline and presumed accurate scoring) and the grading engine scores. The third research question considered the degree to which revised rules for the grading engine more accurately identify errors compared with the existing rule set. As discussed in the Methods section, the manual scoring was used as a baseline. This question will be answered by comparing the correlation between the grading engine scores and the manual scores with the correlation between the revised rules scores and the manually-graded scores.

To summarize the results, the correlation of the revised rules and the manual graded scores was .84 for all unique solutions as compared to the correlation of the current rules and the manual scores, which was .32. However, the p-value statistic comparing the difference between the correlations was insignificant at 0.365. Further detail is provided below.

RQ1: Common Errors Made by Students

Using a spreadsheet tool as an instructional platform is powerful but not without challenges. Spreadsheets are powerful because they allow an individual to accomplish a single task in a variety of ways. The challenge comes when the variety of inputs and controls make it more difficult for the grading system to identify correct or incorrect solutions. In order for a grading engine to be more successful when attempting to identify correct and erroneous solutions, it needs to understand what solutions (both correct and otherwise) students tend to submit. To this end, common errors made by students as well as variations of a correct solution need to be identified.

Incorrect solutions on first submission. Table 2 presents the common unique errors for the first part (i.e., cell H11) of the Force Out at Third assignment, which tests student understanding of Boolean Functions for the Logic and Reference section of the course. Table 2 aggregates the unique incorrect student solutions from a single course. In this data 922 submissions were made from 706 students. As mentioned previously, students are limited to two submissions per assignment. But they are not required to make a second submission. For this assignment 216 students made a second submission. All data in this research consists of first submission data only because second submissions are cumulative and include the unchanged solutions from the first submission. Thus, using only first submission data eliminates duplicate solution submissions. Before reviewing the results of this table, a careful observer will notice there are no spaces or $ (absolute reference signs) in any of the solutions presented in this table. These characters were removed from the solution to afford analysis by comparison. Had these characters been included, the comparison and results would have included functionally duplicate solutions.

Table 2

Unique Incorrect Solutions for the Force Out at Third Task, cell H11, on First Submission

Solution	Task Score (out of 9)	Count	Percent of Total
=AND(C11=D11)	7	14	22.2%
=AND(C11,D11="yes")	7	9	14.3%
=IF(AND(C11="yes",D11="yes"),"Yes","No")	0	6	9.5%
=AND(C11=C11,D11=C11)	7	5	7.9%
=IF(AND(C11="YES",D11="yes"),TRUE,FALSE)	8	3	4.8%
=AND(C11=Yes,D11=Yes)	7	2	3.2%
=AND(C11="Yes",D11="No")	5	2	3.2%
=IF(C11="yes",D11="yes")	8	2	3.2%
=AND(C11="Yes",D1="Yes")	7	1	1.6%
=C11=D11	6	1	1.6%
=AND(C12="Yes",D12="Yes")	7	1	1.6%
=AND(C11="YES",D12="YES")	7	1	1.6%
=IF(AND(C11="Yes",D11="Yes"),"True","False")	0	1	1.6%
=AND(C11="yes",D14="yes")	7	1	1.6%
=AND(C11,D11,G11)	7	1	1.6%
=AND(C11=1,D11=0)	7	1	1.6%
=AND(TRUE,FALSE)	7	1	1.6%
=AND(C11=C11,C11=D11)	7	1	1.6%
=IF(AND(C11="YES",D11="YES"),"TRUE")	6	1	1.6%
=+AND(C11="Yes",D11="Yes")	8	1	1.6%
=AND(C10:C40,D10:D40="yes")	7	1	1.6%
=IF(C11="Yes",AND(D11="Yes",TRUE))	8	1	1.6%
=IF(AND(D11="Yes",E11="Yes"),"TRUE","FALSE")	0	1	1.6%
no	0	1	1.6%
=AND(C1="Yes",D1="Yes")	7	1	1.6%
=AND(C1="Yes",D11="Yes")	7	1	1.6%
=AND(C10="Yes",D10="Yes")	7	1	1.6%
=AND(C11=TRUE,D11=TRUE)	7	1	1.6%

Table 2 includes only incorrect solutions, which is defined as any solution that did not receive a full 9 points from the grading engine. These solutions came from the “Force Out at

Third" task in cell H11. The first column in Table 2 is the solution the student typed in. The second column is the grading engine score for that solution. The third column is the number of students who submitted this solution and the forth column is the percent of the number of solutions submitted. Interestingly, the total number of incorrect solutions for the Force Out at Third task was 64 out of 706 students or 9.1%. This means that 90.9% of the students received a perfect score of 9 out of 9 for this task on the first submission. The bulk of this research is focused on identifying the common errors for the 9.1% of students.

There are three important aspects of the most common errors data provided in Table 2. First, there exist common incorrect solutions across students. The most common incorrect solution (displayed below in Figure 3) represents over a fifth of the total incorrect solutions (22.2%). The primary misconception here is in the logic of the condition. This solution will produce an error when C11 and D11 have a value of "no". There also may be a misconception in that only one parameter is used. At a minimum the AND function should have two parameters but works with one, albeit a redundant solution in that case.

=AND(C11=D11)

Figure 3. Example of a common error made by students on their first attempt. The error involves a logic error and possibly a misconception about function parameters.

There is only one variation of this solution in the table, shown below in Figure 4. In this solution the student did not use the AND function. The solution would have worked if the solution had been =C11=D11="yes" but the task asked students to use the AND function. As in the example above, the misconception seems to be an issue of incomplete logic. The solution functions correctly when both cells are "yes" but produces an incorrect result when both are "no."

=C11=D11

Figure 4. A variation of the most common error made by students on their first attempt. The error involves a logic error but also fails to utilize the AND function as instructed.

The accurate identification of this single error would account for almost a quarter of the unique incorrect solutions, and the potential for misdiagnosing this error is low because there is only one closely related solution.

There are relatively few unique incorrect solutions that more than one student submitted. These are expressed as the first eight solutions in Table 2. Across these eight, three of them begin with the IF function. The use of the IF function indicates another common misconception or knowledge gap. The IF function is often found in the list with a total of 15 solutions containing some variation of the IF function being used. Another common attribute across various incorrect solutions is the use of TRUE or "TRUE," FALSE or "FALSE," which is also related to the IF function but also is indicative of students misunderstanding data types (i.e., text and Booleans). Of the eight most common incorrect solutions (solutions with a frequency more than one), five contain two equals signs, which indicates the correct understanding of the need to include two comparisons inside the AND function. The presence of the IF function and the corresponding TRUE, FALSE and "Yes," "No" outside the AND function, in solutions with more than one student submission, suggests another common error.

Another item of interest in Table 2 is the long list of unique incorrect solutions where only one student submitted the solution. They represent 20 of the 64 or 31.3%. It is difficult to effectively group these solutions into similar error groups. For example, the solution displayed in Figure 5 includes a cell reference error (D1 instead of D11) as well as a syntax error (period instead of a comma). These might be typing errors (i.e., rushed work) and not actual student misconceptions.

=AND(C11=”Yes”.D1=”Yes”)

Figure 5. Example of a likely typo or simple syntax error made by students on their first attempt.

Another example of a solution that incorrectly compares cell references to Boolean values instead of text is provide below in Figure 6. Here the student seems confused about how to construct the condition. The content of cells C11 and D11 are text not Boolean values.

=AND(C11=TRUE,D11=TRUE)

Figure 6. Example of an error made by students on their first attempt that indicates a misunderstanding of cell contents and possibly different data types.

Another solution is the word “no.” Some students will type in the word “no” instead of a formula. This solution is less common than the other solutions and provides evidence that the student may not understand various knowledge component and types in the result in an attempt to scam the system. Ironically they should have typed in TRUE or FALSE not the text “yes” or “no”. The fact that this is not a common solution suggests that most student realize this is not a viable solution; still, it is important to identify this particular error because a student that commits this error may need remedial help or targeted feedback.

Variation of correct solutions. Table 3 provides additional insight into the first research question by describing the variation within the unique *correct* solutions for this task. Significantly, the most common correct solution (displayed below in Figure 7) accounts for 93.5% of the 643 total correct solutions submitted. This means that 93.5% of the students used the same correct solution for this task on the first submission of the assignment. However, not all the solutions that received full points would be considered completely correct and some are more elegant than others.

=AND(C11="yes",D11="yes")

Figure 7. This solution is the intended correct answer.

Table 3

Variations of a Correct Solution provided for the Force Out at Third Task, H11, on First Submission

Solution	Task Score	Count	Percent
=AND(C11="yes",D11="yes")	9	601	93.5%
=AND(C11:C40="Yes",D11:D40="Yes")	9	24	3.7%
=AND((C11)="Yes",(D11)="Yes")	9	4	0.6%
=AND("Yes"=C11,"Yes"=D11)	9	2	0.3%
=AND(D11="yes",C11="yes")	9	2	0.3%
=AND((C11="Yes"),(D11="Yes"))	9	2	0.3%
=AND(C11="YES",(D11="YES"))	9	2	0.3%
=AND(C11="Yes",D11="Yes",TRUE)	9	1	0.2%
=AND('BooleanFunctions'!C11="yes",D11="yes")	9	1	0.2%
=AND(C11=D11,C11="Yes")	9	1	0.2%
=AND(C10:C40="Yes",D10:D40="Yes")	9	1	0.2%
=AND((C11:C30)="Yes",(D11:D30)="Yes")	9	1	0.2%
=AND(C11="Yes",D11="Yes")=TRUE	9	1	0.2%
Grand Total		643	100.0%

There were 26 students (4%) who submitted a solution that generates a correct result but provides evidence of a potential misconception. For example, the use of a range is unnecessary (see Figure 8) but due to the way the Excel program analyzes the equation, the solution only uses one cell in the range when parsing the function. While this solution receives full marks, remedial feedback might be in order.

=AND(C11:C40="Yes",D11:D40="Yes")

Figure 8. Example of a solution with a range in the formula.

Several of the solutions are simple variations of the intended correct answer with the only difference being the order of the parameters, minor changes to the conditions used, or added parenthesis. One example of a challenging correct solution in terms of identification is provided in Figure 9. The logic is correct but not intuitive. The solution should get full credit.

=AND(C11=D11,C11="Yes")

Figure 9. Example of a correct solution that is not identical to the intended solution. The logic is correct but not intuitive in terms of straightforward simple logic.

Second submission errors. Table 4 presents common errors in the context of first and second submissions. As described above, after making the first submission, the student receives feedback for each incorrect task. Thus, Table 4 captures the solutions submitted in the first attempt followed by the solution of the second submission and the score for the second submission. The count column indicates the number of individuals who submitted a solution then corrected it in a particular way. The total number of first submissions that were corrected in the second submission is 26. Table 4 does not include incorrect first submissions that (a) did not make a second submission, or (b) incorrect first submissions that made a second submission but did not change the solution in cell H11. Also, the second submission has been adjusted by removing spaces and absolute references for the purposes of comparison.

It is interesting to note the large variety of unique first submission solutions that result in only 2 unique second submission solutions across 26 students. The 26 students represent 86.7% of the total 30 students who a received a less than full score on the first submission and attempted a correction on the second submission. This means there were only students who were unsuccessful at fixing an error from the first submission in the second submission. Of note is the fact that one solution obtained full points but did so in an unorthodox manner. They used a cell reference in the condition, apparently to be used as a placeholder for the text “yes.”

Table 4

Common First Submission Errors Corrected in Second Submission

1st Submission Solution	1st Sub Score (out of 9)	2nd Submission Solution	2nd Sub Score (out of 9)	Count
=AND(C11=D11)	7	=AND(C11="yes",D11="yes")	9	6
=IF(AND(C11="yes",D11="yes"),"Yes","No")	0	=AND(C11="yes",D11="yes")	9	4
=AND(C11,D11="yes")	7	=AND(C11="yes",D11="yes")	9	3
=IF(AND(C11="YES",D11="yes"),TRUE,FALSE)	8	=AND(C11="yes",D11="yes")	9	2
=AND(C11="Yes",D1="Yes")	7	=AND(C11="yes",D11="yes")	9	1
=AND(TRUE,FALSE)	7	=AND(C11="yes",D11="yes")	9	1
=AND(C10:C40,D10:D40="yes")	7	=AND(C11="yes",D11="yes")	9	1
=AND(C10="Yes",D10="Yes")	7	=AND(C11="yes",D11="yes")	9	1
=C11=D11	6	=AND(C11="yes",D11="yes")	9	1
=IF(AND(D11="Yes",E11="Yes"),"TRUE","FALSE")	0	=AND(C11="yes",D11="yes")	9	1
=AND(C1="Yes",D11="Yes")	7	=AND(C11="yes",D11="yes")	9	1
=AND(C11="Yes",D11="No")	5	=AND(C11="yes",D11="yes")	9	1
=IF(C11="yes",D11="yes")	8	=AND(C11="yes",D11="yes")	9	1
=AND(C11=1,D11=0)	7	=AND(C11="yes",D11="yes")	9	1
=AND(C11=C11,D11=C11)	7	=AND(C11=F11,D11=F11)	9	1

Table 5 shows the students who received less than full marks on the first submission and less than full marks on the second submission. The first submissions in Table 5 are the potential errors that are not corrected in a second submission. The first row in Table 5 shows the change the student made from the 1st submission to the 2nd submission, which was to change the output of the IF function from “Yes, “No” text to “True,” “False” text. The quotes around the True and False indicate a knowledge gap. And the score did not change, but stayed at 0. Ironically, the correct solution existed embedded in the IF function. However, this first submission solution is the same first submission solution on the second row in Table 4. Because 4 students were able to correct this first submission, this error does not seem to qualify as a “sticky” error. The second first submission solution in Table 5 contains 2 errors, (a) the AND function is embedded inside an IF function, and (b) the results are text values not Boolean values. The second submission solution is correct except for the extra pair of parentheses. The second submission on the third row (in Table 5) seems to be a slip where a double quote mark is in the wrong position. Finally, the forth row in Table 5 contains errors that are also resolved by other students as shown in Table 4 (AND function embedded in an IF function, and text instead of booloan values). To summarize, while there are 4 students who did not completely resolve the knowledge gaps from the first submission, the errors that remain are resolvable by other students. Thus, the likelihood of “sticky” errors, or in other words, errors that resist feedback and are present in both the first and second submission for this task is low. Additional research in other courses with new students could shed more light on this topic.

Table 5

Common Errors Not Corrected in Second Submission

	Solution	1st Sub Score *	2nd Sub Score
1st Submission	=AND(C11=D11)	7	
2nd Submission	=AND(C11="Yes,D11=""Yes")		7
1st Submission	=IF(AND(C11="YES",D11="YES"),"TRUE")	6	
2nd Submission	=IF(AND(C11="YES",D11="YES"),"Yes","No")		0
1st Submission	=IF(AND(C11="Yes",D11="Yes"),"True","False")	0	
2nd Submission	=(AND(C11="yes",D11="yes"))		8
1st Submission	=IF(AND(C11="yes",D11="yes"),"Yes","No")	0	
2nd Submission	=IF(AND(C11="Yes",D11="Yes"),"True","False")		0

Sub scores were out of 9 points

Summary of RQ1 results. This section has described the common post-feedback errors of students. The presence of errors in both the first submission and the second submission suggest there is room for improvement in diagnosing knowledge gaps and providing better feedback. While these data do not indicate the present of a “sticky” error, it is possible that other tasks such as tasks involving the VLOOKUP function or embedded IF functions, may prove to contain sticky errors. Examining errors across multiple submissions may provide additional perspective into the nature and duration of knowledge gaps than examining first submission solution errors only.

Comparing Table 3 to Table 2, the variety of answers among the correct submissions is much less than the variety among incorrect submissions. Intuitively, this makes sense. There are more ways to build the incorrect solution than the correct solution. In addition, the variation between unique solutions in Table 3 seems to be confined to small number of iterations adding extra parenthesis (e.g., (C11)) or changing the order of a condition (e.g., “Yes”=C11). There were however a few seemingly correct solutions that might indicate a misconception on the part of the students. The comparison also shows that within the collection of unique incorrect solutions, there are patterns of errors. These patterns express themselves across multiple student submissions.

Another perspective into the common errors students make is to examine the instances of errors that students make after receiving feedback and making a second submission. For example, as mentioned above, from the 706 students who made a first submission, 216 students made a second submission. The most common persistent errors after a second student submission could be named a “sticky” error. One in which the error is not resolved after the student receives grading-engine feedback and makes a second submission. The architecture in the present system provides for a first and second submission, supports this additional perspective into common errors, and is uniquely positioned to identify potential “sticky” common errors. The opportunity to observe post-feedback errors does not shed light on how an error was resolved. It could have been as a direct result of the feedback or of some other resource including the student's own resourcefulness. However, the errors, when seen at this level provide another layer of insight regarding the nature of the knowledge gaps and misconceptions because it accounts for post-feedback errors that are commonly after feedback and a second submission regardless of how they are resolved. In other words, this perspective begins to distinguish errors not just by how many students made the error, but how many students continued to make the error after feedback was provided.

RQ2: Grading Engine Alignment with Knowledge Components

The second research question regarding how well aligned the current grading engine rules are to individual knowledge components will be the focus of this section. Currently, the majority of grading engine rules do not align well with individual knowledge components above. In the "Force Out at Third" cell H11 task, there are five rules. The first rule returns correct if the AND function is used. This rule is worth 1 point. This rule is the best-aligned rule to a knowledge component in this task. It is fully aligned with the first knowledge component in Table 1, KC1, which asks if the solution used the correct boolean function. The next four rules do not align to any individual knowledge component. In the second rule the grading engine changes the values of the two input cells to "Yes" and at the same moment builds its own solution. The grading engine compares the resulting value of its solution with the resulting value of the student's solution. If they are equal, the student receives full credit for the rule, which is 2 points for each of the four rules. The four rules check four different sets of input values to the student solution. If the student's solution answers all five rules correctly the student receives 9 points for the task. If the student answers the first four rules correctly but misses the last rule, the student receives 7 points for the task. However, these rules do not align with individual knowledge components. This is best evidenced by the feedback given to students when one of these rules is incorrect. An example of the feedback message provided to the student is, “Value in H11 is incorrect when C11 is Yes and D11 is Yes.” This message follows directly from the grading engine rule, which changes the value of C11 and D11 to Yes and checks the resulting value. While one rule out of the five in the Force Out at Third task aligns with an individual knowledge component (i.e. checking to see if the student uses the correct boolean function), the other four align to multiple knowledge components including the correct data type, the correct formula syntax, the correct logic, and the correct condition.

The set of 6 knowledge components used in this research, as shown in Table 1, are (a) the correct function, (b) the correct data type, (c) the correct formula syntax, (d) the correct logic, (e) the correct condition (or comparison), and (f) no additional syntax or functions. While not all tasks in the course are formula-based, such as tasks involving the construction of charts or using advanced data analysis functions, these six knowledge components are used to diagnose student understanding for formula-based tasks. The first five knowledge components identify specific parts of a formula. The sixth seeks to capture what is not captured in the other five.

This research performed manual grading on each unique formula from the H11 cell using the six knowledge components above. The results highlight solutions where the grading engine score does not agree with the manual score. There are the only two cases where the solution was scored full marks by the grading engine but less than full marks from the manual grading (as shown in Figure 10 below).

=AND(C11="Yes",D11="Yes",TRUE)

=AND(C11="Yes",D11="Yes") = TRUE

Figure 10. Examples of solutions receiving full marks by the grading engine but less than full marks from the manual grading.

In both cases, the solutions contain extra, not necessary arguments. But the extra arguments do not negatively impact the outcome of the formula. In the first solution an extra TRUE argument is included inside the AND function. While this argument does not negatively affect the outcome of the solution, it represents an extra syntax in the AND function. The second solution contains an extra TRUE argument outside of the AND function. Again, this extra argument does not change the outcome of the solution and like the prior solution points are not deducted by the grading engine, yet it represents a knowledge gap about the AND function. To the credit of the grading engine, these solutions were the only 2 solutions out of 643 total solutions (0.3%) where the grading engine produced a false positive (when the solution receives more points than it should receive). A false positive indicates an undiagnosed knowledge gap. In this case, the inclusion of an extra TRUE argument is not diagnosed. A false negative occurs when the grading engine incorrectly diagnoses a knowledge gap when none is present. Out of the 63 incorrect solutions (totaling 28 unique solutions, see Table 2), there were 0 false negatives, or solutions with scores that should have received full marks but did not.

This section has described the six knowledge components used in this research, and how the current grading engine rules for the “Force Out at Third” task align to these knowledge components. We have reviewed the false positive and false negative unique solutions for the task and observed examples of the manually grading by knowledge component.

Because the knowledge components, which were generated by research experts, were the rubric for the manual grading process and the revised grading, it is appropriate to review how the knowledge components came about and were used in the manual grading.

Because the purpose of the research was a proof-of-concept regarding the idea of combining knowledge components with transaction-level data, the priority was on identifying distinguishable knowledge components not on identifying the best or most accurate knowledge components. It is our hope that future research might continue to refine and improve the nature of knowledge components to improve even further the diagnostic power of knowledge components at the transaction level. A detailed description of each knowledge component in included below. This section describes the assumption that the manual grading process was the "gold standard" to which the revised rules were compared. To this point, the manual grading process serves as the gold standard because of the expert diagnosis in identifying knowledge gaps and misconception. While VanLehn (2011) argues that human tutors may not be the gold standard the field has traditionally considered them to be, we justify this position on the grounds that the research experts had sufficient skill and experience in the subject matter to recognize knowledge gaps. The research acknowledges the lack of a perfect set, or the best set of knowledge components by suggesting that there is a balance between functionality and efficiency. In other words, there are some solutions that are less efficiently correct, but more do not sacrifice functionality. For example, the "= TRUE' in Figure 10 is not functionally needed and could qualify for a knowledge gap in the creation of the formula if the student believes it is required. But, the "= TRUE" is not needed to achieve a correct outcome. And yet, there is an argument to be made that by including it, the formula is easier to troubleshoot and constitutes in programming language a way to provide documentation in the code. Thus, this solution could be less efficient, but more functional for the student in the long run. These considerations stem from the complexity and diversity of the spreadsheet functions. The wide breadth of complex formula arrangements allows for a wide variety of uses. This complexity is, in part, a reason for the creation of the sixth knowledge component, which tries to capture extraneous, not needed arguments or characters in the solutions.

Thus, this research contends that the knowledge components identified in this research are sufficient to distinguish knowledge gaps and flexible enough to allow for some flexibility in the grading algorithm.

The following section will describe the outcome of 706 first submissions using new grading engine rules more closely aligned to individual knowledge components.

Revised Rules

This section will respond to the third research question, “To what degree do revised rules for the grading engine more accurately identify errors?” The revised rules are divided into 2 groups. The first group searches for evidence of correct solutions. When correct, these rules add points to the total points for the task. The second group searches for evidence of incorrect solutions. The evidence for these searches came primarily from incorrect or inefficient solutions from more than one student. Thus, the new rules were designed at the cross hairs of common student errors, which come from research question 1, and the specified knowledge components from research question 2. This section will review each new rule by knowledge component, the results of the regrading process, and a description of the scoring results from the current rules scoring, the manual scoring and the new rules scoring results for the “Force Out at Third” task in cell H11.

Knowledge Component 1 –Boolean Function

The first revised rule improved the diagnosis of the correct function used in the solution. Previous research found that when the function did not immediately follow the equals sign (=), the grading engine would not correctly grade the function. For example, when the function included a parenthesis between the equals sign and the beginning of the function "=(AND," the grading engine would not recognize the function. This revised rule was created to more accurately identify the function notwithstanding extra characters between the equals sign the function name. With the revised rule, the solutions that begin with "=(AND" will be given full points for including the correct boolean function in the solution. The results of this rule after the batch regrading process identified only 4 of the 706 first submission H11 cells without the AND function. These solutions included the 3 unique solutions in Figure 11.

=IF(C11="yes",D11="yes")

no

=C11=D11

Figure 11. Three examples of solutions without the AND function.

The first solution uses the IF function instead of the AND function. Two students submitted this solution. The second solution is the word "no." This student did not enter a formula, but typed in an answer manually. The third solution does not contain any function name within the solution. These 3 unique solutions account for 4 total first submissions.

Further, the current rule looking for the AND function does not give points to 14 submissions whose functions include AND, but the function is not immediately following the equals sign. The 14 instances are false negatives. They are false negatives because they are marked as incorrect by the grading engine but should be marked correct. The revised rule, then, reduced the false negatives from 14 to 0.

Only 8 out of 706 total first submissions did not contain the AND function. This is not entirely unexpected. Upon closer examination in the log files, the great majority of students self-diagnosed the lack of the AND function and correct it relatively quickly. The more difficult task, as evidenced by the more incorrect attempts was to use the correct syntax and logic. But before moving to these knowledge components, there is one additional revised rule for this knowledge component.

The second revised rule for KC1 (correct Boolean function) searched the solution for a specific knowledge gap—the presence of the AND function embedded in an IF function (i.e., "=IF(AND …"). We found a high frequency of this specific knowledge gap within the incorrect functions. For example, 15 out of the 18 solutions with incorrect functions embedded the AND function inside of an IF function as displayed in Figure 12.

```
=IF(AND(C11="yes",D11="yes"),"Yes","No")
```

Figure 12. Example of the AND function embedded within an IF function.

These 15 solutions represent the presence of a knowledge gap about the use of the correct function, AND. For some students the result of the IF function for these solutions is a Boolean value. The student does not understand that the result of the AND function is a Boolean value, thus the IF function unnecessarily replicates the functionality of the AND function. The correct solution is incorrectly embedded in the IF function (as shown in Figure 13).

```
=IF(AND(C11="YES",D11="yes"),TRUE,FALSE)
```

Figure 13. Example of a solution containing a correct AND function embedded in an IF function.

This rule accurately identifies a knowledge gap about the use of the correct Boolean function.

Knowledge Component 2 – Data Types

One of the most common challenges in the Boolean section of the assignment is the correct use of data types. It is all too common for students to confuse the text "TRUE" with the Boolean TRUE. When displayed in a cell, the two can be deceiving similar. For example, a cell containing the text value of "TRUE" will not be centered (horizontally) in the cell. While the Boolean value TRUE is centered in the cell. This is the only significant visible difference to the user, yet the functional difference is much larger. In the "Force Out at Third" task, the student makes a comparison between 2 input cells and the text "Yes" inside the AND function. If both input cells are "Yes" then the result of the AND function is the Boolean value TRUE. If either cell is not "Yes" then the AND function returns the Boolean value FALSE. The grading engine will correctly distinguish between the text "TRUE" and the Boolean value TRUE. This knowledge component determines if the student uses the correct data type for this task.

In order to improve the diagnosis capability of the grading engine, the first revised rule for this knowledge component (and the third revised rule overall) searches for the word Yes inside the solution. Specifically, the rule searches for the expression Yes with the equals sign and the quotes around the Yes (=”Yes”). The quotes are needed because they define the text string within the solution and to distinguish from 2 other solutions that contain Yes without quotes. These Yes refer to a custom, named range within the spreadsheet built by the students. Because this approach is outside the instructional guidelines, this response is considered incorrect. Out of 706 first submissions, 678 instances contain the search string text (=”Yes”). This means that the 678 instances all correctly compare the text value "Yes" inside the AND function. Thus, there are 28 instances without this comparison. The grading engine feedback (The solution in cell H11 does not include "yes" = or = "yes") is shown to these 28 students.

Knowledge Component 3 – Function Syntax

Two rules were created based on this knowledge component. The first searches the solution for the text, TRUE). The ending parenthesis indicates the position of the TRUE word inside the solution. A common knowledge gap among student submissions is to include the Boolean value TRUE inside the AND function. This rule identifies this trend for 3 out of 706 first submissions. The grading engine feedback to the student is, “The formula in H11 should not include the boolean value TRUE.”

The second revised rule for KC3 is to check for the text “(C11=D11).” This text represents a specific knowledge gap regarding the syntax of the AND function. This knowledge gap represents another common mistake found in research question 1 of this article. The parentheses are included in the text search for this rule because there are other knowledge gaps with the C11=D11 text that are different than this knowledge gap. The C11=D11 knowledge gap (without parentheses) is the reduced form of the actual formula $C$11=D11. This formula, while much less common than the (C11=D11) formula, contains a confounding knowledge gap. Thus, in order to separate knowledge gaps, the parentheses are included in this rule. Results show 14 first submissions contain this knowledge gap. This knowledge gap represents the most common unique solution among the 63 solutions (out of 706, or 11.2%) receiving less than the full 9 points for this task. In other words, out of the 63 solutions with errors, 14 contained this knowledge gap (22%) as shown in the first solution in Table 1. The grading engine feedback for this rule is “The formula in H11 should not compare cells C11 and D11 to each other. Instead they should be compared to the value ‘Yes.’”

Knowledge Component 4 – Correct Condition

The next rule searches for “C11=” or “=C11” to diagnose if the student has a knowledge gap related to the forth knowledge component, the logic of condition (KC4 in Table 1). The condition in this task is to compare C11 and D11, which are the input cells with the string “Yes”. The challenge in designing this rule was that there were many students whose answers were correct, but did not use the string “C11=” or “=C11.” It was not uncommon for students to use a variety of variations including: "C11:C40=", "C11:C30=", "C10:C40=", and "C10:C30=." The inclusion of these variations resulted in the correct diagnosis of 688 solutions out of 706 containing this text and the correct diagnosis of 18 solutions where the text was not found. In these 18 solutions, this knowledge gap was correctly identified. One interesting example where the knowledge gap was found is shown in Figure 14.

=AND(C10:C40,D10:D40="yes")

Figure 14. Example of a solution validating the correct diagnosis of lacking and equals sign in the first argument of the AND function.

While his solution includes the C10:C40 range, it does not compare this range to the text “yes” and therefore the rule correctly identifies the knowledge gap in this case. The feedback message for incorrect conditions is, “The formula in cell H11 does not compare cell C11 to text.”

A similar rule was created for the second comparison in the formula, “D11=”. This rule was also expanded to account for the variety of cells and ranges, including: "=D11", "D10:D40=", "D11:D30=", "D11:D40=", "D10:D30=". Thus, if the solution does not contain one of these text strings, then the student does not receive full points for this task and a knowledge gap is identified. The total number of solutions with one of these texts was 697, is slightly higher than those from the C11= rule (688). It is interesting to note that out of the 6 different search strings used in this rule 4 of them include ranges, which again notes the variety of inputs inherent in the spreadsheet platform. However, solutions with range references (e.g., D11:D40) instead of cell references (e.g., D11) account for just 27 of the 698, or 3.9% of the total solutions submitted.

This rule is aligned with knowledge component 4, the logic of condition (KC4) and the grading engine feedback message, displayed when the solution contains a logic of condition error as described above is, “The formula in cell H11 does not compare cell D11 to text.”

The final revised rule aligned with KC4 is the YesNo rule. This rule is a more specific version of the IF(AND rule above. This rule diagnoses the solutions where the output of the IF function is a Yes or No. The rule searches students’ solutions for the Yes and No text at the end of an IF function with an embedded AND function (see Figure 15).

```
=IF(AND(C11="yes",D11="yes"),"Yes","No")
```

Figure 15. Example of a solution with a “Yes” and “No” at the end of an IF function.

Specifically, the rule searches for the “Yes”,”No” text. This rule found 6 instances of this text across 64 of the incorrect solutions, or 9.4%. This error stems from an incorrect understanding of the instructions of the task. This knowledge gap would most likely be corrected in the second submission with appropriate grading engine feedback following the first submission.

Knowledge Component 5 – Condition Syntax

A new rule was created to measure the correct syntax of the AND function in this task. It looks for 2 equals signs inside the AND function. The 2 equals signs correspond to the two comparisons, one for C11 and one for D11. The correct syntax for this rule includes many elements such as open and closing parentheses, correctly positioned commas, and two arguments inside the AND function, etc. This rule counts the number of equals signs to diagnose if two comparisons were made inside the AND function. The rule correctly identified 678 solutions with 2 equals signs. And did not find 2 equals signs in 28 solutions. The feedback message for this rule reads, “The cell in H11 does not contain 2 equal signs.”

Knowledge Component 6 – Extra Syntax

The revised rules related to this KC focus on three specific knowledge gaps. These are (a) Extra Parentheses, (b) Range references instead of cell references, and (c) Extra plus signs. These rules carry a 0-point deduction because in all submissions these potential knowledge gaps do not negatively affect the outcome of the solution. However, they could cause additional problems or knowledge gaps in later tasks. Thus, the purpose of these rules is to offer additional instruction without penalizing the score.

The first rule diagnoses the presence of extra parentheses in the solution. While extra parentheses, if formatted correctly, do not negatively affect the outcome of the solution, it can be more difficult to troubleshoot problems in more complex solutions. Reducing the number of parentheses is one way to reduce the potential complexity of future solutions.

The second rule seeks to flag solutions using cell ranges instead of individual cell references. This rule found 27 out of 706 (3.8%) solutions that meet this criterion. Interestingly, only 1 of the 27 did not received full points from the grading engine; the other 26 solutions all received the full 9 points for the task. Also, 24 of 27 solutions are the same when extra spaces and the absolute reference symbols ($) are removed from the solutions.

The third and final rule for this KC diagnoses a knowledge gap where students include the plus sign before the function name. Anecdotally, the plus sign comes in part from users familiar with the old Lotus spreadsheet system. Again, while this character does not change the functional outcome of the formula, it could be distracting in later, more complex formulations. Within this student data set of 706 submissions, only 1 student used the plus sign before the AND function.

Grading Engine Feedback

Table 6 presents the feedback messages for each of the revised rules. Part of the justification for new rules was the idea that the feedback would be more helpful for students to make needed adjustments to their work and make a second submission. While this research does not collect reactions of students or students' behaviors from revised feedback messages, it does present suggested feedback messages based on the revised, knowledge component-based rules.

Table 6

Revised Rules Error Messages

KC	Revised Rules Feedback Messages
1	The function in cell H11 does not contain the AND function.
1	The AND function does not need to be embedded in an IF function.
2	The formula in cell H11 does not include "yes" = or = "yes"
2	The boolean value TRUE should not be inside quotes.
3	The formula in H11 should not include the boolean value TRUE.
3	The formula in H11 should not compare cells C11 and D11 to each other.
4	The formula in cell H11 does not compare cell C11 to text
4	The formula in cell H11 does not compare cell D11 to text
4	The display value should be True or False (a Boolean value) instead of Yes or No.
5	The function in H11 does not contain 2 equal signs.
6	The function contains extra parenthesis.
6	The function contains a range instead of a single cell reference.
6	The function does not need a "+" after the equals sign.

The proposed feedback messages follow the intent of the current feedback messages, which is that the grading engine feedback identifies the presence of a specific error, but does not provide step-by-step instruction to correct the problem.

RQ3: Revised Rules Scores Compared to Manual Scores

To answer research question three (To what degree do revised rules for the grading engine more accurately identify errors?), the revised rule scores are compared with the manual scores. Because the manual scores are considered a “gold standard” a correlation between the current scores and the manual scores was compared with a correlation between the revised scores and the manual scores. The intent was to improve the accuracy of the revised scores to be closer to the manual scores. Thus, if the correlation of the revised rule scores and the manual scores was higher than the correlation between the current grading engine scores and the manual scores, then the revised rule scores would be considered a closer match to the manual scoring than the current grading engine rules thereby proving the revised rules to be more accurate than the current grading engine rules

Before the correlations are presented, an important distinction should be brought to light. While the research contained over 800 student participants, not every student submitted a unique solution. As described above in research question one, the number of unique solutions was greater for incorrect solutions (28) than for correct solutions (13). This seems logical because a correct solution is achieved within a relatively narrow range of pathways and solutions. On the other hand, there are many more incorrect pathways and incorrect solutions than there are correct pathways and correct solutions.

The amount of variety of incorrect solutions in this learning environment is different than other environments. For example, in a multiple choice test, the variety of incorrect answers is limited. Certainly, there are benefits to limiting the variety of incorrect solutions. One benefit is the speed of grading. Instead of identifying why the learner’s solution was incorrect, the limitation of incorrect solution variety affords rapid assessment of whether a solution is correct or not. Yet, the disadvantage of limited variety of incorrect solutions results in less diagnostic value in understanding why a learner made an incorrect solution. This point also touches on the essence of the challenge of designing personalized learning experiences.

Table 7 presents the three sets of scores for each correct, unique solution. The correlation between these scores is a perfect 1 because there is no deviation among them. This also suggests that no false positives exist in the manually scoring—no solution with full marks (6 out of 6) in the manual scoring received less than full marks in the current grading engine scores or the revised rules scores.

Table 7

Scores for Correct (6 on Manual Score) Unique Solutions

Unique Solutions	Current Score (out of 9)	Manual Score (out of 6)	Revised Score (out of 5)
=AND("Yes"=C11,"Yes"=D11)	9	6	5
=AND((C11)="Yes",(D11)="Yes")	9	6	5
=AND((C11:C30)="Yes",(D11:D30)="Yes")	9	6	5
=AND((C11="Yes"),(D11="Yes"))	9	6	5
=AND('BooleanFunctions'!C11="yes",D11="yes")	9	6	5
=AND(C10:C40="Yes",D10:D40="Yes")	9	6	5
=AND(C11:C40="Yes",D11:D40="Yes")	9	6	5
=AND(C11="yes",D11="yes")	9	6	5
=AND(C11=D11,C11="Yes")	9	6	5
=AND(D11="yes",C11="yes")	9	6	5

Table 8 shows the three sets of scores for each unique solution that scored less than 6 on the manual scoring. Interestingly, there are 2 solutions with less than perfect manual score (<6), but have a perfect current grading engine score (9). These solutions are shown in Figure 16.

=AND(C11="Yes",D11="Yes")=TRUE

```
=AND(C11="Yes",D11="Yes",TRUE)
```

Figure 16. Examples of solutions with less than perfect manual scores, but with a perfect grading engine score.

Each solution, even though it is marked completely correct by the grading engine contain extra syntax not needed inside or outside the function. These are false positives (incorrectly marked correct) for the current grading engine. However, with no revised rules score at a 5, this means there are no false positives for the revised rules scores.

Table 8

Scores for Incorrect (Less than 6 on Manual Score), Unique Solutions

Unique Solutions	Current Score (out of 9)	Manual Score (out of 6)	Revised Score (out of 5)
=+AND(C11="Yes",D11="Yes")	8	5	4.5
=AND(C1="Yes",D1="Yes")	7	5	3
=AND(C1="Yes",D11="Yes")	7	5	4
=AND(C10:C40,D10:D40="yes")	7	5	3
=AND(C10="Yes",D10="Yes")	7	5	3
=AND(C11,D11,G11)	7	3	1
=AND(C11,D11="yes")	7	5	3
=AND(C11="Yes",D1="Yes")	7	5	4
=AND(C11="Yes",D11="No")	5	5	4.5
=AND(C11="Yes",D11="Yes")=TRUE	9	5	4.5
=AND(C11="Yes",D11="Yes",TRUE)	9	5	4.5
=AND(C11="YES",D12="YES")	7	5	4
=AND(C11="yes",D14="yes")	7	5	4
=AND(C11=1,D11=0)	7	5	4
=AND(C11=C11,C11=D11)	7	5	4
=AND(C11=C11,D11=C11)	7	3	4
=AND(C11=D11)	7	2	2
=AND(C11=TRUE,D11=TRUE)	7	4	3.5
=AND(C11=Yes,D11=Yes)	7	5	4
=AND(C12="Yes",D12="Yes")	7	5	3
=AND(TRUE,FALSE)	7	3	1
=C11=D11	6	1	2
=IF(AND(C11="YES",D11="YES"),"TRUE")	6	5	4
=IF(AND(C11="Yes",D11="Yes"),"True","False")	0	4	4
=IF(AND(C11="yes",D11="yes"),"Yes","No")	0	5	4
=IF(AND(C11="YES",D11="yes"),TRUE,FALSE)	8	5	4.5
=IF(AND(D11="Yes",E11="Yes"),"TRUE","FALSE")	0	4	3
=IF(C11="Yes",AND(D11="Yes",TRUE))	8	4	4.5
=IF(C11="yes",D11="yes")	8	4	4
No	0	0	0

A correlation measure between the current grading engine scores (Current Scores), the manually graded scores (Manual Scores), and the scores from the revised rules (Revised Scores) for all unique solutions, was used because each score set did not contain the same amount of points. The correlation matrix was built using data from Table 8. The matrix incorporated only unique solutions in order to remove the artificial inflation that would occur with duplicate solutions in the data. For example, incorporating 603 duplicate solutions representing the correct solution would inflate the correlation measure. Thus, only unique solutions were used in the correlation calculation.

A high correlation between score sets means that a high score in one set is also likely to be high score in another set and a low score in one set is more likely to be low score in another set. A low correlation suggests that a high score in one set is less likely to be a high score in another score set and a low score in one set is less likely to be a low score in another score set. This research used the manual score set as the gold standard to which other score sets should be compared. It was the objective of this research to improve this correlation measure by designing revised rules and comparing the correlation of the revised rules scores and the manual scores with the current scores and the manual scores. If the correlation of the revised scores and manual scores is higher than the correlation between the current scores and the manual scores, then the revised scores more closely match the manual scores compared to the current scores. If the correlation between the current scores and the manual scores is higher than the correlation between the revised scores and the manual scores, then the current scores (and by extension the current grading engine rules) are more closely matched to the manual scores.

The correlation including all unique solutions (correct and incorrect) between the current scores and the manual scores was 0.52. The correlation between the revised scores and the manual scores was 0.84. Because the revised and manual scores correlation is higher than the current and manual scores correlation, this suggests the revised scores are a closer match to the manual scores than the current scores. A p-value was calculated comparing at the difference between the correlations. The p-value of 0.365 was not significant. While this result suggests that the difference in correlations could be explained by chance, this assumes that these are population values and that these are the only questions we are ever interested in. If that is the case, then p-values don't make much sense since this is a census rather than a sample. In this case we might be better served to confine pour inference to just this class and then generalize the principles learned rather than these results. Given the small sample size this qualitative assessment of the effect has practical significance. The effect can be assumed to be real and if we had a larger pool of questions to sample from, thus increasing the sample size, the effect would persist and the p-value would decrease.

Because this was a proof-of-concept study, this result is not unexpected. Future research, incorporating solutions from another student set is going to include a different set of unique solutions than those in this research. Thus, while the p-value was not significant in this set, if the variation of unique solutions changes, it could greatly impact p-value significance.

Looking deeper into the comparison between the manual and the current scores, we see the scatter plot in Figure 17. The scatter plot shows the current scores on the horizontal axis and the manual scores on the vertical axis. The scatter plot contains 3 points with a current score of 0. These scores, one could argue, skew the data away from the main group of scores. Furthermore, the 0,0 point on the scatter plot is the solution "no" with no formula present. This student did not attempt to enter a formula, but instead directly typed in an answer without using a formula. In this case both the manual score and the current score were 0. While there exists the possibility of removing this point from the data, it was decided to include the point because even a non-formula entry should be a part of a diagnostic function. Thus, the correlations and the p-value include this point.

At this proof-of-concept research stage, what may be more significant than statistical significance is practical significance. This includes the significant grade differences for students for the same unique solution. In Table 8, there are more than one case where the current score of a solution is much greater, on a percentage basis, than the manual score. In these cases, the current score does not accurately reflect the knowledge gap demonstrated by the student. The grade the student receives could lead to changes in the student's decision to make a second submission or not.

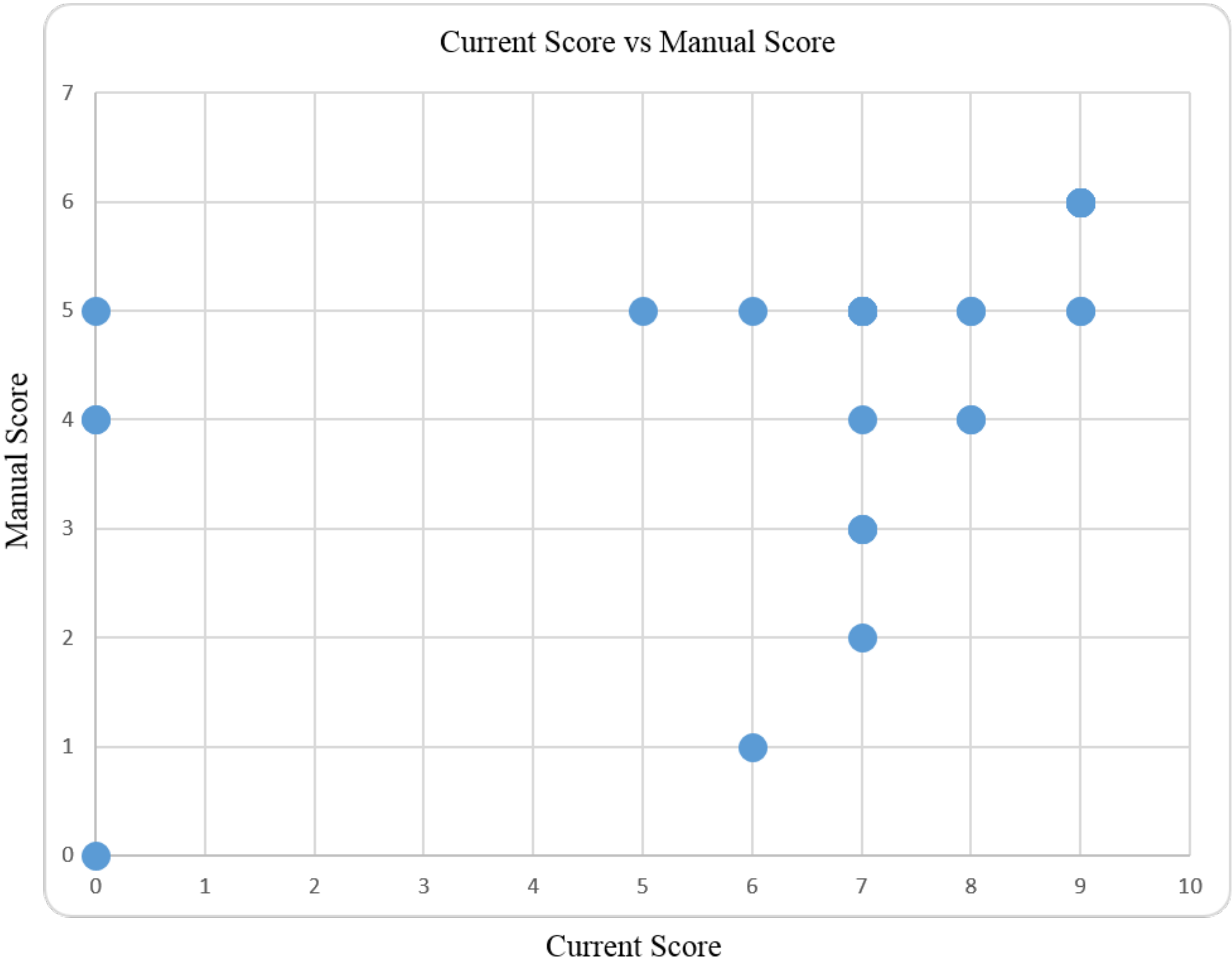


Figure 17. A scatter plot showing the two variables Manual Score and Current Score. Notice the outliers with a Current Score of 0.

As noted above, the fully correct manual scores were also fully correct for both the current scores and the revised scores. A more conservative correlation measure comes from the removal of the correct unique solutions. The correlation for the incorrect, unique solutions between the current scores and the manual scores was 0.33. The correlation between the revised scores and the manual scores was 0.76.

Discussion

There are a number of worthwhile discussions to pursue to further explore the results above. This section will discuss the generation of the revised rules, the idea of rule conflict, and the revised rule error messages compared to the current rule error messages.

The revised rules can be divided into two diagnostic groups. The first group diagnoses evidence of correct solutions. This includes evidence such as solutions resulting in correct values and formulas formatted correctly. The second group diagnoses evidence of incorrect solutions. This categorization of correct and incorrect evidence is evident in the scoring of the revised rules. Revised rules with a positive value (+1) diagnose correct evidence. Revised rules with a negative value (-0.5 or -1) diagnose incorrect evidence.

Rules diagnosing incorrect evidence more directly link to knowledge gaps better than rules diagnosing correct evidence link to knowledge gaps. Yet, the diagnosis of correct evidence is required in order to produce a score. Thus, a conflict emerges in the assessment of student submissions. On one hand, diagnosing evidence of correct solutions gives the student points, which are needed to score the assignment and the student's performance in the course. Yet, receiving 0 points for a task is not specific enough to provide information on which knowledge components are not well understood. Identifying less understood knowledge components enables personalized feedback and learning.

It is useful to consider a revised rule that was created but not included in the new set of revised rules, and why it was not included. This revised rule measured the length of the solution. If the solution was longer or shorter than the correct solution length, then the feedback to the student could provide a basis from which the student could determine what should be removed or what should be added to the solution. Based in part on the wide variety of unique incorrect solutions, this rule was created to help diagnose solutions that were significantly shorter or longer than the correct solution. The most common correct solution is shown in Figure 17.

=AND(C11=”Yes”,D11=”Yes”)

Figure 17. The most common correct solution.

One of the challenges regarding the length of a solution is the many different characters that could be included in a formula. Some of the most common extra characters include extra spaces, the absolute value symbol ($) attached to a row (C$11), or a column ($D11), or both ($C$11), and extra parentheses. In these cases, removing the spaces and the absolute reference symbols reduced the variety and improved comparability. The most common correct form of the solution without these extra characters has a length of 25 characters. Yet, there are also 32 correct solutions which are 33 characters in length. With only 5 correct solutions (out of 847) have lengths less than 25 (2 solutions) or greater than 33 (3 solutions), it would seem that a boundary of correct solution lengths would be between 25 and 33 inclusive. So, a new rule was designed to provide feedback communicating to students either the solution was too short (when less than 25 characters) or too long (when greater than 33 characters). But, the challenge of using length as a way to differentiate correct from incorrect solutions was that a number of correct solutions were outside the 25-to-33 inclusive boundary, and a number of incorrect solutions had lengths that were inside the 25-to-33 boundary. Thus the results from this rule contained too many false positives and false negatives to be included in the final set of rules. For this task, solution length does not provide convincing evidence of a correct or incorrect solution because of the overlap between correct and incorrect solutions. Nor does it provide any specific guidelines regarding the nature of the error. However, future research may profit from combining this rule with others to identify specific knowledge gaps.

One of the challenges of revising grading engine rules is the potential for conflict between rules. In one case a rule needed to remove all parentheses from the solution because extra parentheses were preventing the rule to identify the string "C11=" within the solution. However, when the parentheses were removed from the solution, the rule searching for "IF(AND" did not work because it contained a parenthesis. The design of new rules does not take place outside of other rules. It is an integrated activity. This has implications for future rule generation, in that new rule generation is content specific.

Generalizability

The ability to generalize the findings of a research study is important. Unfortunately, in most social science research the ability to identify cause and effect relationships that generalize across contexts or can even be replicated consistently is rare (Open Science Collaboration, 2015). Clearly the context within which this study was conducted precludes a direct transfer of the processes to other instructional situations. However, the basic principles put to bear in this study would be consistent if the ability to capture transaction-level data were possible. This research is a case study that seeks to inform how human-designed knowledge components combine with task-level data to improve the diagnostic function of inner-loop feedback. While the actual knowledge components used in this research will differ from other courses or curriculum, the knowledge component construct is generalizable to other contexts. The inner loop is the term used in Intelligent Tutoring Systems research referring to the steps a learner takes to solve a problem or complete a task (VanLehn, 2006). Especially for those with inner-loop feedback in technology-enabled learning environments. This research is applicable to any inner-loop functionality.

In addition, the generalizability of this study also rests on the foundation of human-designed or human-identified knowledge components. While this study did not seek to answer the question which are the best knowledge components, it helped to compare the value of human-identified knowledge components needed to improve the diagnostic function within inner-loop feedback. The human-identified knowledge components codified into the grading engine has the potential to dramatically improve inner-loop feedback.

Feedback Messages

Current Grading Engine Feedback Messages include the following:

- The function in cell H11 is incorrect,
- H11 has the wrong result when C11 is Yes and D11 is Yes,
- H11 has the wrong result when C11 is No and D11 is No,
- H11 has the wrong result when C11 is No and D11 is Yes,
- H11 has the wrong result when C11 is Yes and D11 is No.

These messages were presented to the student when a current grading engine rule was incorrect. The messages were presented together, at one time, to the student after the student submitted the assignment for grading.

One of the benefits of designing revised rules for the grading engine is that the feedback messages presented to students can be specific to the knowledge component that is incorrect.

While testing the proposed feedback messages for the revised rules was outside the scope of this study, it is useful to compare them to the current feedback messages.

The feedback message for the first revised rule is not materially different from the feedback for the current rule because the revised rule closely resembles the current rule. The feedback for the second revised rule identifies a specific problem, “The AND function does not need to be embedded in an IF function.” Another feedback message communicates another specific knowledge gap, “The boolean value TRUE should not be inside quotes.” These messages do not explicitly identify the steps to fix the knowledge gap, but it is the opinion of this research that they offer more specific feedback regarding what is incorrect. Another common incorrect feedback message is, “The solution in H11 should not compare cells C11 and D11 to each other.” Future research should consider evaluating or improving these feedback messages. An interesting measure to consider is to identify the frequency of second submission changes by feedback message to identify any correlational relationship between the nature of the feedback message and the number of second submission changes.

Conclusion

The current grading engine achieves extremely accurate and effective results regarding the scoring of correct and incorrect solutions. It offers partial credit, it accurately grades 32 unique correct solutions as correct, which account for 641 solutions out of the 706 total solutions (91%), and accurately grades 63 unique incorrect solutions. Certainly, the capabilities of the current grading engine are beyond compare in terms of the accuracy of grading across the variety of solutions. Yet, the grading engine only marginally diagnoses specific knowledge components, known or unknown, to students. The revised rules augment the remedial potential of the current grading engine by diagnosing specific knowledge components and providing knowledge component-based feedback.

The goal in this research was to align grading engine rules to individual knowledge components to improve the diagnosis of the error and to provide more actionable feedback messages to students. As has been demonstrated, knowledge component-based rules have been designed and the scores have been tested within the confines of this student data set to be a closer match to manual grading scores. Feedback messages, based on rules designed in the context of knowledge components have been proposed.

At a much higher level, this research is a substantial step toward improving how technology can diagnosis knowledge gaps for individual students. One impact improved diagnosis can have is to increase the number of attempts students make in their learning path. Currently, most learning experiences are limited to a single attempt. In many cases this limit stems from the additional resources, in time or budget or both, needed to perform the additional grading. Papers or reports or tests are usually a one-and-done experience, where the student does not have another opportunity to apply the feedback received from the grading. Using technology to improve the diagnosis of error could lead to improved learning by facilitating the application of feedback into multiple student attempts without requiring significant additional resources to assess the work. Yet challenges exist. This research has provided a step toward understanding how to overcome these challenges. One challenge is the variety of inputs in complex instructional systems. A spreadsheet application is considered a complex instructional system because of the wide variety of possibly inputs to each individual cell. Much less complex systems include a multiple choice test where the inputs are limited to 5 inputs A, B, C, D, or E for each task. The benefit, of course, of a less complex input system is the relative ease and scale of grading. Yet there, the diagnosis of error and knowledge gaps remains elusive. If variety can be appropriately managed, a spreadsheet application can combine the benefits of complex input without creating overbearing grading demands and provide specific, personalized diagnoses to students. Without constraints on the complex inputs, it becomes impossible to identify all of the unique incorrect attempts, which are needed to build new, revised rules and actionable feedback messages.

Future Research

There are a number of future research opportunities to pursue at this point. Research which improves both the combination and outcome of transaction-level data with human-identified knowledge components will continue to make an impact for future technology-enabled learning environments. Researchers with transaction-level data should incorporate human-identified knowledge component views into the data. Researchers with knowledge components should incorporate the perspective of transaction-level data. The relationship between the variety of unique incorrect attempts and the number of students is an interesting research question because it is assumed that at some point the total number of unique attempts would begin to plateau. But this assumption remains to be tested. The answer to this question helps to identify how many knowledge component-based rules should be designed.

Also, the relationship between the transaction-level data and feedback should be investigated further. How does feedback change students' attempts and how is this manifest in transaction-level data?

While the knowledge components used in this research are not research-proven or industry standards, future research should also consider improving knowledge components.

References

Benyon, D., & Murray, D. (1993). Adaptive systems: From intelligent tutoring to autonomous agents. *Knowledge-Based Systems*, *6*(4), 197–219. https://doi.org/10.1016/0950-7051(93)90012-I

Bloom, B. S. (1984). The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher*, *13*(6), 4–16.

Brooks, C., Greer, J., & Gutwin, C. (2014). The data-assisted approach to building intelligent technology-enhanced learning environments. In J. A. Larusson & B. White (Eds.), *Learning analytics* (pp. 123–156). New York, NY: Springer. Retrieved from http://link.springer.com/chapter/10.1007/978-1-4614-3305-7_7

Christensen, C., Johnson, C. W., & Horn, M. B. (2010). *Disrupting class, expanded edition: How disruptive innovation will change the way the world learns* (2nd ed.). New York, NY: McGraw-Hill Education.

Chung, G. (2014). Toward the relational management of educational measurement data. *Teachers College Record*, *116*(11), 1-16.

Harlow, H. F. (1959). Learning set and error factor theory. *Psychology: A Study of a Science*, *2*, 492–537.

Lewis, B. N., & Pask, G. (1965). The theory and practice of adaptive teaching systems. *Teaching Machines and Programmed Learning*, *2*, 213–66.

Shute, V. J., & Zapata-Rivera, D. (2008). Adaptive technologies. In J. M. Spector, M. D. Merrill, J. van Merrienboer, & M. P. Driscoll (Eds.), *Handbook of research on educational communications and technology* (3rd ed., pp. 277–294). New York, NY: Lawrence Earlbaum Associates.

VanLehn, K. (2006). The behavior of tutoring systems. *International Journal of Artificial Intelligence in Education*, *16*(3), 227–265.

White, B., & Larusson, J. A. (2010). *Strategic directives for learning management system planning* (Research Bulletin No. 19). Boulder, CO: Educause Center for Applied Research. Retrieved from http://www.educause.edu/ecar

DISSERTATION CONCLUSION

This conclusion section connects the three articles to each other and to the research agenda. A synthesis of the research is also provided.

This dissertation examined the role of transaction-level data and knowledge component domain models to improve the accuracy and diagnostic value to learners. The first article presented a framework combining both human-identified knowledge components and transaction-level data analytics to more accurately categorize and identify learner knowledge gaps. The second article found that learners with the same final solution do not understand the same knowledge components. This article suggests that transaction-level data may provide better visibility to evaluate learner understanding than final solution data. The third article leveraged and combined the first and second articles by testing human-identified knowledge components within a transaction-level-based grading engine.

The first and second article provided both the framework and justification to conduct the research of article three. The framework of the first article incorporated previous research from the field of Intelligent Tutoring Systems (ITS), including how knowledge is organized (the domain model), how the student's knowledge is tracked (the student model), and the unit of analysis of the learner's knowledge (the concept of knowledge components). The framework combined these concepts with transaction-level data only made accessible to researchers given relatively recent developments in educational technology. The second article articulated the lack of clarity that exists in evaluating learner understanding using final solution data only. In other words, because learners with the same final solution do not have the same knowledge component understanding, final solution-based diagnosis does not distinguish sufficiently to provide personalized intervention needed for learners.

The research agenda explores questions about the role of technology in teaching and learning. The area of current focus is related to the function technology can play in the diagnosis of learner knowledge gaps. Within the research field, there are efforts to allow a machine learning algorithm to systematically identify the best combinations or configurations of knowledge components to achieve the best learning outcomes. The research presented in this dissertation suggests a parallel research agenda to explore the combination of human-identified knowledge components and transaction-level data analytics. This combination is not necessarily implying that computer-based knowledge components are less effective, but that human-identified knowledge components combined with transaction-level data analytics might more accurately address the challenge to better incorporate learning theory or pedagogy into the design and assessment of technology-enabled learning environments. In this context, the research agenda and articles presented here represent a unique current and future research opportunity.