Designing Real-time Teacher Augmentation to Combine Strengths of Human and AI Instruction

Kenneth Holstein

CMU-HCII-19-104
September 2019

Human–Computer Interaction Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213
kjholste@cs.cmu.edu

Thesis Committee:
Vincent Aleven (Co-Chair), HCII, CMU
Bruce M. McLaren (Co-Chair), HCII, CMU
Jodi Forlizzi, HCII, CMU
Pierre Dillenbourg, School of Computer and Communication Sciences, EPFL
Nikol Rummel, Institute of Educational Research, RUB & HCII, CMU

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

The research reported here was supported, in whole or in part, by the Institute of Education Sciences (IES) under Grants R305A180301 and R305B150008 and by the National Science Foundation (NSF) under Grant #1530726 to Carnegie Mellon University. All views expressed in this document are those of the author and do not necessarily reflect the views of the sponsoring agencies.

KEYWORDS

AI-supported classrooms, algorithmic experience, alignment, augmentation, augmented reality, authoring tools, automation, autonomy, awareness, blended learning, care work, caring profession, classroom experiments, classroom orchestration, closing the loop, complementarity, co-orchestration, co-design, dashboards, data-informed design, design methods, decision-making, decision-support, design research, educational data mining, experience prototyping, heads-up displays, human–AI hybrid, human–AI interaction, human-centered design, intelligent tutoring systems, K-12, learning analytics, learning sciences, mixed reality, multimodal learning analytics, multi-stakeholder, Open Learner Model, participatory design, pedagogical decision-making, peripheral displays, personalized classrooms, physical learning analytics, prototyping methods, rapid prototyping, real-time analytics, replay analysis, replay visualization, service design, spatial visualizations, student modeling, teacher-in-the-loop, teachers, user experience design, user modeling, visualization, wearable technologies

ABSTRACT

When used in K-12 classrooms, AI-based educational software such as intelligent tutoring systems (ITSs) allows students to work at their own pace, while also freeing up the teacher to spend more time working one-on-one with students. A common intuition is that, in many situations, human teachers may be better suited to support students than ITSs alone (e.g., by providing socio-emotional support, supporting student motivation, or flexibly providing conceptual support when further problem-solving practice may be ineffective). Yet ITSs are not typically designed to work together with teachers during a class session, to take advantage of these complementary strengths.

This dissertation explores how AI tutors might be better designed to work together with human teachers in real-time, to amplify teachers' abilities to help their students. Working together with 36 middle school math teachers, I conducted the first broad exploration in the literature of teachers' needs for real-time analytics and orchestration support in AI-supported, personalized classrooms. As part of this work, I worked with teachers to design a form of real-time, wearable teacher augmentation called *Lumilo*.

Lumilo is a set of mixed-reality smart glasses that direct teachers' attention during a class session, towards situations the tutoring software may be ill-suited to handle on its own, and support teachers in deciding how best to respond. *Lumilo* has been used by teachers and students in over 40 middle school classrooms so far. An in-vivo classroom experiment showed that teacher–AI co-orchestration, as supported by *Lumilo*, enhanced students' learning compared with an AI-supported classroom in which the teacher did not have such support.

Over the course of this research, I have also developed new design and prototyping methods to address challenges in the co-design, experience prototyping, and evaluation of data-driven AI systems. To support the use of these methods within the area of education, my collaborators and I have extended an existing technical architecture (*CTAT/TutorShop*) to facilitate rapid prototyping of data-driven educational AI applications.

In the final chapters of this dissertation, I explore how the concepts embodied by *Lumilo* might be prepared for wider use, from two angles. First, I involve students, as well as teachers, in the next phase of design to better serve the needs and respect the boundaries of both stakeholder groups. Second, through a newly-formed academic–industry partnership with Carnegie Learning (a major educational AI company) I begin to explore how real-time, wearable teacher augmentation might be generalized to work with a broader range of AI tutoring systems and curricula.

ACKNOWLEDGEMENTS

I love reading PhD thesis acknowledgements sections. Although it's commonly said that "no one is ever going to read your thesis," I find myself browsing theses relevant to my research interests fairly regularly. One of the sections I read most closely are the acknowledgements. It is not always common in academia to hear honest expressions of deep gratitude and appreciation. So, acknowledgements sections provide a rare excuse to be sentimental in public. That said, maybe we should all strive to make this feeling less rare – working day to day to make academia feel more like the inside of an acknowledgements section.

First, I would like to thank my PhD advisors and committee co-chairs, **Vincent Aleven** and **Bruce McLaren**, without whom this research would not have been possible. Through their guidance and feedback during my PhD, I have had opportunities to learn so much more beyond "conducting research" – including grant writing, project planning and management, budgeting, balancing/trading-off across competing academic responsibilities, research mentorship, and a host of other cognitive and metacognitive skills that are core to a career as an academic researcher. In addition to advising me directly, Vincent and Bruce were also responsible for advising several of my favorite researchers – now colleagues in research communities such as CHI, AIED, and LAK, among others – several of whom have in turn given me valuable, informal advice throughout my PhD. I would also like to thank my committee members, **Jodi Forlizzi**, **Pierre Dillenbourg**, and **Nikol Rummel** for their comments, probing questions, and guidance both before and during the dissertation period, which have played a major role in strengthening this research. Beyond our interactions, much of their prior and ongoing research has helped to inspire and inform the work presented here. I hope to have opportunities to collaborate with each of you at some point in the future!

I would also like to thank the incredible undergraduate and masters research students I have had the pleasure of mentoring and working with during my PhD, who have contributed to the work presented in this dissertation. These include **Gena Hong, Peter Schaldenbrand, Mera Tegene,** and **Jasper Tom**, and **Zac Yu**. I also thank **all collaborating K-12 teachers, students, and schools** (who will remain anonymous) who have contributed continuously to this research over the last several years.

No HCII PhD acknowledgements section is complete without an enormous thank you to **Queenie Kravitz**, who has been an incredible source of support during my PhD (as she has been to so many others in our department and our field over the years). I would also like to thank **Sharon Carver**, **David Klahr**, **Audrey Russo,** and the entire **PIER Steering Committee** for

their support and feedback throughout my PhD. I thank **Anind Dey, Jeff Bigham, Jodi Forlizzi,** and **Scott Hudson** for their support in their roles as HCII Director and PhD Program Director during my time at the HCII, as well as the **entire HCII faculty, staff, and student body** for co-creating such a supportive and inspiring environment.

I would like to thank my research mentors in the FATE (Fairness, Accountability, Transparency, and Ethics in AI) research group at Microsoft Research: **Hal Daumé III, Miro Dudík, Hanna Wallach,** and **Jennifer Wortman Vaughan.** My experiences working with them during my PhD have been incredibly impactful in shaping my perspectives on research, mentorship, cross-disciplinary collaborations, the role of HCI/design in AI and machine learning, and what problems are important to work on.

I have had the privilege of working with several other amazing research mentors prior to this dissertation work: **Emma Brunskill**, **Charles Kemp, Chris Carroll, Soniya Gadgil-Sharma, Chris Lucas,** and **Tim Nokes-Malach**. Their influence continues to shape my approaches to research and mentorship. For example, my years working with undergraduate mentors Chris Lucas, Charles Kemp, and Chris Carroll on computational cognitive science research inspired a (thus far) lifelong interest in understanding and combining complementary strengths of human and machine intelligence. This interest has driven much of my dissertation work, and continues to be a core theme in most of my other research projects.

Thank you to my wife, **Mary Beth Kery**, who I met through the PhD program. Her talent and excitement for HCI research, design, art, and teaching – and her passion for making the world a kinder place, for both human and non-human animals – inspires me each and every day. She has helped me persevere through many challenges during the PhD, and there is simply no way that this document could exist in its current form were it not for her support.

Thank you to the many researchers who reached out to make me feel welcome at some of the first research conferences I attended (and ever since), including **Yoav Bergner, Ran Liu, Roberto Martinez-Maldonado, Inge Molenaar, Luis Prieto, Maria Rodríguez−Triana, Ido Roll,** and **Joseph Jay Williams**. For a first-time conference attendee, this means the world. I try to do the same for younger researchers at conferences and other events, whenever I have the opportunity.

Beyond my advisors and my committee members, discussions with many other researchers have helped to shape the research presented in this dissertation. These include **Ryan Baker, Simon Buckingham-Shum, Mutlu Cukurova, Laura Dabbish, Shayan Doroudi, Neil Heffernan,**

Judy Kay, Niki Kittur, Ken Koedinger, Chinmay Kulkarni, Austin Lee, Roberto Martinez-Maldonado, Manolis Mavrikis, Inge Molenaar, Amy Ogan, Kaska Porayska-Pomsta, Luis Prieto, Maria Rodríguez−Triana, Ido Roll, Carolyn Rosé, Peter Scupelli, Daniel Spikol, Kurt VanLehn, Roger Schank, Joseph Jay Williams, David Williamson Shaffer, Kalina Yacef, and **John Zimmerman**, among others. I am looking forward to many more inspiring conversations in the future!

Thank you to my **friends and members of my extended PhD cohort**. These include **Lea Albaugh, Julia Cambre, Judeth Oden Choi, Steven Dang, Sauvik Das, Nick Diana, Jonathan Dinu, David Gerritsen, Samantha Finkelstein, Rebecca Gulotta, Erik Harpstead, Amber Horvath, Iris Howley, Sung-A Jang, Haojian Jin, Anna Kasunic, Yasmine Kotturi, Gierad Laput, Fannie Liu, Yanjin Long, Michal Luria, Michael Madaio, Joselyn McDonald, Tomo Nagashima, Felicia Ng, Huy Anh Nguyen, Jenny Olsen, Julian Ramos Rojas, Michael Rivera, Kelly Rivers, Alex Sciuto, Joseph Seering, Amy Shannon, Dan Tasse, Alexandra To, Judith Uchidiuno, Xu Wang, Kristin Williams, Françeska Xhakaj, Qian Yang, Nesra Yannier, Yang Zhang, Siyan Zhao,** among so many others.

Thanks also to my **collaborators** new and old, including **Pengcheng An, Saskia Bakker, Susan Berman, Henriette Cramer, Shayan Doroudi, Steve Fancsali, Jean Garcia-Gathright, Erik Harpstead, Wayne Holmes, Petr Johanes, Mary Beth Kery, Matt Larson, Michael Madaio, Jenny Olsen, Octav Popescu, Kaska Porayska-Pomsta, Sravana Reddy, Steve Ritter, Nikol Rummel, Michael Sandbothe, Jonathan Sewall, Joseph Jay Williams,** and **Bev Woolf**, among others.

Thank you to the many amazing **undergraduate and masters students** who I have had the pleasure of mentoring or co-mentoring during my time as a PhD student (outside of my dissertation research) including **Nupur Chatterji, Nawon Choi, Tianxin Chu, David Contreras, Kellyn Dassler, Kailin Dong, Weiqi Fang, Malcolm Guya, Alex Hadi, Casey Hicks, Yanzun Huang, Christina Jin, Bo Kim, Andrew Kim, Karen Kim, Thomas Li, Dongyang Lu, Patrick McLaren, Likang Sun, Jingyu Wang, Kexin Yang, Yuchen Yao, Chuankai Zhang,** and **Bofei Zhu**.

An enormous thank you to my **family** for their love and support. Special thanks to my mother and father, **Adora** and **Steve Holstein**, my brother **Dan Holstein**; my mother and father in-law **Jo-Ann** and **Sean Kery**, and my sisters and brother in-law, **Caroline Kery, Rachel Kery, Katie Gofreddi,** and **John Kery**. Thanks also to my **fluffy family**: the many tiny animals who have stayed at my house and kept me company at various points during my PhD, including **Eevee Horvath, Pichu Horvath, Luna Gonzalez Kery-Holstein, Ollie Kery-Holstein, Taro**

Kery-Holstein, Teru Liu, Juneau McDonald, Toffee Myers, Mallow Seering, Miyuki the Terrace Cat (who sadly passed away recently), and **Hubble Zhao-Rivera.**

Finally, thank you to everyone not explicitly listed here, for your support during, before, and (hopefully) after my PhD.

Table of Contents

List of Tables

List of Figures

Introduction and Background

To facilitate more personalized learning, AI-based educational software is increasingly being used in K-12 classrooms (Bingham, Pane, Steiner, and Hamilton, 2018; Luckin, Holmes, Griffiths, and Forcier, 2016). Intelligent tutoring systems (**ITSs**), a major class of AI-based educational software, have been shown through several meta-analyses to significantly enhance student learning when used in classrooms, compared with traditional classroom instruction and other forms of educational technology (e.g., Kulik & Fletcher, 2016; Ma, Adesope, Nesbit, & Liu, 2014; Steenbergen-Hu & Cooper, 2013; 2014; VanLehn, 2011; Xu, Wijekumar, Ramirez, Hu, & Irey, 2019). These systems provide step-by-step feedback and guidance to students – tailoring instruction to individual learners as they work through problem-solving activities at their own pace (Corbett, Koedinger, & Hadley, 2001; Ritter, Anderson, Koedinger, & Corbett, 2007; VanLehn, 2006).

In the first year of my PhD, I spent over 100 hours conducting field observations of ITS use in a diverse range of elementary and middle school classrooms around the Pittsburgh area (see row 1 of Table 1). These field observations were conducted in the context of a research project aimed at training and evaluating machine-learned instructional policies for ITSs, with the ultimate vision of creating autonomous, self-improving tutors (Doroudi, Aleven, & Brunskill, 2017; Doroudi, Holstein, Aleven, & Brunskill, 2015; 2016; O'Shea, 1982). However, as I observed more classrooms, I was struck by the highly active roles that teachers played during ITS class sessions (cf. Schofield, Eurich-Fulcer, & Britt, 1994).

While students worked with the ITS throughout a class period, teachers did not typically sit back behind their desks while the software took over the role of primary instructor. In most of the classrooms I observed, teachers were nearly constantly moving from student to student and providing one-on-one support – for example, by providing conceptual explanations beyond what the ITS was able to provide, by comforting and motivating students when they became frustrated, or by providing remedial instruction to students who lacked key prerequisite knowledge. In line with prior findings (e.g., Schofield et al., 1994), teachers reported viewing these AI-supported class sessions as opportunities for *more* one-on-one interaction with their students, not less (Holstein et al., 2017a; 2017b; 2019a).

At the same time, in speaking with teachers after these class sessions, some reported feeling "left out of the loop" in their own classrooms (Holstein et al., 2017b; Kulkarni, 2019; Segedy, Sulcer, & Biswas, 2010; Yacef, 2002). For instance, teachers found it challenging to monitor classes of students who may all be working on different activities at any one time, particularly given that unlike human teaching assistants, ITSs were unable to communicate with them (Holstein et al., 2017b; 2019a; 2019b; Holstein, 2018). The role of the teacher and the impacts of teachers' on-the-spot help-giving are not commonly considered in the design of ITSs, student modeling methods, or instructional policies for these systems (Aleven, Xhakaj, Holstein, & McLaren, 2016; Holstein et al., 2017a; 2017b; 2018b; 2019a; 2019b; Lesta & Yacef, 2002; Miller et al., 2015; Yacef, 2002).

Table 1. Overview of key sets of studies conducted during my PhD, which informed this dissertation.

Set of studies	Dates	Participant totals	Key outcomes / publications
(1) Classroom field observations and piloting of multiple outer-loop tutoring policies (using the *Fractions Tutor*)	**Fall 2015**	**1,547** students, **58** classes, and **25** teachers from **16** schools / districts	**Key publications:** Doroudi, Holstein, Aleven, & Brunskill, 2015; 2016; Doroudi, Aleven, & Brunskill, 2017 **Key outcomes:** Classroom field observations of substantial teacher help during class sessions (e.g., to "compensate" for limitations of AI tutors) directly inspired my subsequent research focus: designing to support shared teacher–AI orchestration during class.
(2) Classroom field observations, teacher interviews, and data mining of teacher–student interactions in AI-supported K-12 classrooms (using *Lynnette*)	**Spring - Summer 2016**	**299** students, **17** classes, and **5** teachers from **2** schools / districts	**Key publications:** Holstein et al., 2016; 2017a; 2017b; 2019a; Xhakaj, Aleven, & McLaren, 2017 **Key outcomes:** ● **See:** *Chapter 1 and Chapter 2*
(3) Formative design research for teacher–AI co-orchestration tools: Love and breakup letters, directed storytelling, generative card sorting, and speed dating	**Summer - Fall 2016**	**10** teachers from **5** schools / districts	**Key publications:** Holstein et al., 2017b; 2019a **Key outcomes:** ● **See:** *Chapter 1*
(4) Iterative co- design and prototyping of a teacher–AI co-orchestration tool (*Lumilo*) in AI-supported K-12 classrooms	**Winter 2016 - Fall 2017**	**8** teachers from **8** schools and **7** school districts	**Key publications:** Holstein, Hong, et al., 2018; Holstein, 2018; Holstein et al., 2019a; Holstein, Yu, et al., 2018 **Key outcomes:** ● **See:** *Chapter 3, Chapter 4, and Chapter 5* ● This research formed the foundation for two major research grants that I co-wrote during my PhD. These grants will fund various research projects that build upon the present work. ○ **IES grant R305A180301**: Enhancing Student Learning with an Orchestration Tool for Personalized Teacher-Student Interactions in Classrooms Using Intelligent Tutoring Software Education Technology (co-written with Vincent Aleven, Bruce M. McLaren, and Carnegie Learning) ○ **NSF grant #1822861:** Human/AI Co-Orchestration of Dynamically- Differentiated Collaborative Classrooms (co-written with Vincent Aleven and Nikol Rummel)

(5) Iterative Replay Enactments studies with *Lumilo* (and *Lynnette*)	**Summer - Fall 2017**	**10** teachers from **5** schools and **3** school districts (experience prototyping using replays of **5** classes from **2** schools / districts)	**Key publications:** Holstein, Hong, et al., 2018; Holstein et al., 2019a; Zhang et al., 2019 **Key outcomes:** ● **See:** *Chapter 5 and Chapter 6*
(6) Iterative classroom piloting with *Lumilo* (and *Lynnette*)	**Fall 2017 - Spring 2018**	**355** students, **18** classes, and **6** teachers from **3** schools / districts	**Key publications:** Holstein et al., 2018a; 2019a **Key outcomes:** ● **See:** *Chapter 6 and Chapter 8*
(7) In-vivo classroom experiment with *Lumilo* (and *Lynnette*)	**Winter 2017 - Spring 2018**	**343** students, **18** classes, and **8** teachers from **4** schools / districts	**Key publications:** Holstein et al., 2018b; 2019a **Key outcomes:** ● ***See:*** *Chapter 7 and Chapter 8*
(8) Concept generation and validation studies to better understand both teachers' and students' needs for co-orchestration support	**Spring - Summer 2019**	**14** students and **10** teachers, spanning **12** schools / districts, and **2** major US cities	**Key publications:** Holstein et al., 2019b **Key outcomes:** ● **See:** *Chapter 9*
(9) Classroom piloting with *Lumilo* and Carnegie Learning's *MATHia* tutor	**Spring - Summer 2019**	**138** students, **5** classes, and **4** teachers from **1** school / district	**Key outcomes:** ● **See:** *Chapter 10*

These classroom observations inspired the overarching question driving this thesis:

> *How might AI-based educational software best be designed to work together with teachers, in real-time, to actively leverage human teachers' complementary strengths and support them in co-regulating students' learning?*

Over a decade ago, Yacef proposed a reframing of intelligent tutoring systems as "intelligent teaching assistants" (ITAs): systems designed with the joint objectives of helping human teachers teach and helping students learn, rather than only the latter of these objectives (Yacef, 2002). Other researchers have since proposed optimizing student learning by leveraging complementary strengths of human and AI instruction (e.g., Baker, 2016; Ritter, Yudelson, Fancsali, & Berman, 2016b). That is, ITSs might be more effective if they could adaptively enlist human teachers' help, in situations teachers may be better suited to handle than the ITS (cf. Alkhatib & Bernstein, 2019; Davidoff, Lee, Dey, & Zimmerman, 2007; Holstein, Lucas, & Kemp, 2014; Kamar, 2016; Lake, Ullman, Tenenbaum, & Gershman, 2017; Lubars & Tan, 2019; Ritter et al., 2016b). Yet while there has been some work on real-time teacher support tools for ITS classrooms since the vision of ITAs was introduced (e.g., Feng & Heffernan, 2007; Mavrikis et al., 2016), little work

has investigated teachers' actual needs and desires for such support (i.e., through needfinding studies), or how human and automated instruction might be most effectively combined.

In this dissertation, I explore how AI and human teachers might best support one another, leveraging one another's complementary strengths to achieve outcomes greater than either could achieve alone (cf. Alkhatib & Bernstein, 2019; Forlizzi & Zimmerman, 2013; Kamar, 2016; Ritter et al., 2016b). I approach this work from both an *empowerment* and an *efficiency* perspective (see Kulkarni et al., 2019 for a related discussion). From an *empowerment* perspective, I seek to *support and extend teachers' abilities* to personalize instruction, and help them *fulfill the roles they aspire to play* during AI-supported class sessions (e.g., see Aiken & Epstein, 2000; Feng & Heffernan, 2007; Holstein et al., 2017b; 2019a; 2019b; Holstein, Hong et al., 2018; Mavrikis et al., 2016; Yacef, 2002). From an *efficiency* perspective, I seek to design human/AI systems that will *measurably benefit students' learning* by making more effective use of existing classroom resources, compared with human teachers or AI tutors working in a less-integrated fashion (e.g., see Baker et al., 2016; Fancsali et al., 2018; Holstein et al., 2017a; 2018a; 2018b; Holstein, 2018; Kamar, 2016; Ritter et al. 2016b).

One promising way to support effective teacher–AI partnerships is through the design of classroom analytics tools (Holstein et al., 2018a; Rodriguez-Triana et al., 2017; Tissenbaum et al., 2016; Verbert et al., 2013) and classroom orchestration systems more broadly (Dillenbourg & Jermann, 2010; Dillenbourg, Prieto, & Olsen, 2018; Prieto, Holenko, Gutierrez, Abdulwahed, & Balid, 2011; Martinez-Maldonado, 2016; van Leeuwen et al., 2018). Classroom analytics tools such as dashboards are commonly designed to enhance teachers' awareness of what goes on in their classrooms, for example by presenting teachers with real-time information about students' learning as they work with educational technologies. Classroom orchestration systems represent a broader class of technologies that may provide more comprehensive support for managing and effectively supporting a class of students as they work through instructional activities (Dillenbourg & Jermann, 2010; Dillenbourg, Prieto, et al. 2018).

In the current work, I created a real-time, wearable classroom orchestration tool to support and empower K-12 teachers who use ITSs in their classrooms. To this end, I adopted a participatory design approach, directly involving teachers at each stage, from initial needfinding to the selection and tuning of real-time analytic measures through iterative prototyping (see *Chapters 1, 4, 5,* and *6*, and Holstein, Hong, Tegene, McLaren, & Aleven, 2018; Holstein et al., 2017b; 2018a; 2019a).

The prototype that emerged from this iterative co-design process was a pair of mixed-reality smart glasses called *Lumilo* (see *Chapter 4* and Holstein, Hong, et al., 2018), which tunes teachers in to the rich analytics generated by ITSs, alerting them to situations the ITS may be ill-suited to handle and providing additional information, upon request, to support teachers in determining how to address these situations. In doing so, *Lumilo* is designed to facilitate

productive mutual support or *co-orchestration* between the teacher and the ITS (Holstein et al., 2017b; 2018b; Payne et al., 2008; Prieto, 2012; Sharples, 2013), by leveraging the complementary strengths of each (Holstein et al., 2019b). Through an in-vivo classroom experiment (see *Chapter 7*, and Holstein et al., 2018b), I found that this form of teacher–AI co-orchestration had a positive overall impact on student learning, and helped teachers better support students with lower prior domain knowledge.

Over the course of this research, I have also developed new UX design and prototyping methods to address challenges in the co-design, experience prototyping, and evaluation of data-driven AI systems. These include Replay Enactments (see *Chapters 5* and *6,* and Holstein, Hong, et al., 2018; Holstein et al., 2018a; 2019a), Causal Alignment Analysis (see *Chapter 6* and Holstein et al., 2018a), a participatory variant of the Speed Dating method (Davidoff, Lee, Dey, & Zimmerman, 2007; Zimmerman & Forlizzi, 2017) called Participatory Speed Dating (see *Chapter 9* and Holstein et al., 2019b), and the use of spatial classroom replay visualizations to inform design (see *Chapters 2* and *6*, and Holstein et al., 2017a; 2018a). To support the use of these methods in the area of AI-supported education, my colleagues and I have extended the existing *CTAT/TutorShop* architecture to facilitate rapid prototyping of data-driven, educational AI applications – including both student- and teacher-facing tools (see *Chapter 3* and Holstein, Yu, et al., 2018).

My work on *Lumilo* demonstrates promise for approaches that integrate human and machine intelligence to support student learning. However, the prototyping sessions and classroom studies mentioned above also revealed broader needs for orchestration support in AI-supported classrooms – among both teachers and students – extending beyond those addressed by *Lumilo*'s current design (see *Chapter 8* and *9,* and Holstein, Hong, et al., 2018; Holstein et al., 2017b; 2019a). For example, both teachers and students expressed needs for better mechanisms to support "private" teacher–student communication during a class session (e.g., to enable students to signal help-need during class without losing face to peers). In addition, after using *Lumilo* in live K-12 classrooms, teachers began to reveal more nuanced preferences for which classroom tasks should be handled by the AI, which should be handled by the teacher, and which should be handled by students (and under which circumstances) (cf. Davidoff et al., 2007; Lubars & Tan, 2019; Olsen, 2017; Prieto, 2012; Rummel, 2018). Similarly, students began to reveal needs for greater agency over how their personal analytics are used and interpreted than *Lumilo* (and associated ITSs) currently provides.

Building upon these and other findings, in the final chapters of this thesis (*Part Four*), I involve students, as well as teachers, in the next phase of design (Forlizzi & Zimmerman, 2013). Through iterative concept generation and validation exercises, I work with students and teachers to better understand their respective needs and boundaries (*Chapter 9*, and Holstein et al., 2019b). Drawing upon my own work, as well as prior literature on supporting self-regulated, collaborative, and teacher learning, these investigations take an initial step towards addressing

the following, *broader formulation* of the question that originally motivated this thesis (see above):

> *How might AI-based educational software best be designed to work together with teachers **and students**, in real-time, to actively leverage their complementary strengths and support them in co-regulating **both teacher and student learning**?*

Further explorations in this broader direction are left for future work (see *Conclusions, Contributions, and Future Directions* for a discussion). In particular, I plan to explore the design space of *student–teacher–AI* co-orchestration systems in a new grant-funded project with Vincent Aleven and Nikol Rummel: *Human/AI Co-Orchestration of Dynamically- Differentiated Collaborative Classrooms* (NSF grant #1822861).

In addition, through a newly-formed academic–industry partnership with Carnegie Learning (a major educational AI company) in *Chapter 10*, I begin to explore how tools like *Lumilo* might be designed for wider-spread use.

Beyond the scope of this dissertation, the explorations presented in *Part Four* will help prepare for the next phase of this research, funded by a new grant with Vincent Aleven, Bruce McLaren, and Carnegie Learning (IES R305A180301): a large-scale classroom experiment (using an updated and miniaturized version of *Lumilo*) with over 60 middle school classrooms that use Carnegie Learning's *MATHia* ITS, with the aim of better understanding the effects of teacher–AI co-orchestration on student learning and other classroom outcomes.

In sum, this dissertation makes a total of 7 main contributions to the areas of human–computer and human–AI interaction (**HCI/HAI**), design (**DES**), and learning sciences and technologies (**LS&T**). These contributions are briefly summarized in Table 2, organized by area. Each contribution is summarized in greater detail under *Conclusions, Contributions, and Future Directions*.

Following Wobbrock and Kientz's high-level taxonomy of research contribution types in HCI (Wobbrock & Kientz, 2016), contributions are categorized by each contribution's primary type (out of "Empirical", "Artifact", "Methodological", "Theoretical", "Dataset", "Survey", and "Opinion"). Secondary contribution types are also listed where applicable. I have further divided the "Empirical" category into two subcategories (which are not mutually exclusive): "Design research" and "Classroom experiments and data mining".

Table 2. Brief overview of this dissertation's seven main contributions.

Areas	Contribution	Contribution type(s)
HCI/HAI, DES, LS&T	**(1) First broad design exploration of needs for real-time teacher analytics and orchestration support:** This dissertation presents the first broad exploration in the literature of teachers' needs for real-time analytics and orchestration support in AI-supported, personalized classrooms. More broadly, the design explorations presented in this dissertation represent a case study of the design of real-time AI augmentation for workers in a "caring profession" (K-12 teaching) which may defy full automation.	**Design Research (and Theoretical)**
HCI/HAI, DES, LS&T	**(2) First design exploration and prototypes of wearable, heads-up displays to support orchestration of personalized classrooms:** This dissertation presents the first design exploration in the literature of the use of wearable, heads-up displays to support teachers in orchestrating personalized classrooms, yielding *Lumilo*, a classroom-tested prototype of mixed reality smart glasses for teachers. More broadly, the design explorations presented in this dissertation represent a case study of the use of head-mounted displays in a real-world social space (K-12 classrooms).	**Design Research (and Artifact)**
HCI/HAI, DES, LS&T	**(3) First experimental study to demonstrate student learning benefits of real-time teacher analytics:** This dissertation presents the first experimental study to demonstrate that real-time teacher analytics can measurably enhance students' pre-post learning outcomes (either within or outside the areas of AI-supported education and intelligent tutoring systems).	**Classroom Experiments and Data Mining**
HCI/HAI, DES, LS&T	**(4) Novel design and prototyping methods:** This dissertation introduces novel design and prototyping methods to support the co-design, experience prototyping, and evaluation of data-driven AI systems, and case studies exploring how these methods can be applied to involve non-technical stakeholders in the design of such systems.	**Methodological (and Design Research)**

LS&T	**(5) First investigations of relationships between teacher movement/monitoring and student behavior and learning in AI-supported classrooms:** This dissertation presents the first investigations in the literature of relationships between teachers' physical movement and classroom monitoring behaviors, and students' behaviors and learning outcomes, in AI-supported, personalized classrooms. Findings from this research indicate that, when evaluating the impacts of teacher-facing learning analytics tools, future research should take care to tease apart potential effects of a teacher's use of a monitoring tool versus teachers' use of learning analytics.	**Classroom Experiments and Data Mining**
LS&T	**(6) Causal Alignment Analysis (CAA), a framework for the data-informed, iterative design of teacher augmentation:** This dissertation introduces Causal Alignment Analysis (CAA), a framework for the data-informed, iterative design of teacher augmentation (e.g., real-time awareness and orchestration tools), which links the design of such technologies to educational goals; and a case study illustrating CAA's application and utility.	**Methodological (and Theoretical)**
LS&T	**(7) CTAT/TutorShop Analytics (CT+A), an extended technical architecture for ITS development that supports "extensible student modeling":** Finally, this dissertation presents CTAT/TutorShop Analytics (CT+A), an extended technical architecture for ITS development that supports "extensible student modeling": the sharing, re-mixing, use, and prototyping of advanced student modeling techniques.	**Artifact**

Document Organization

This dissertation is organized into four main parts, each of which consists of multiple chapters. Figure 1 provides a graphical overview of how these four parts relate to my published work and other research directions I have pursued during my PhD.

In **Part One**, I present initial needfinding studies with K-12 teachers (*Chapter 1*), exploratory classroom data analyses (*Chapter 2*), and technical groundwork (*Chapter 3*) that set a foundation for all of my subsequent research.

In **Part Two** I present an iterative prototyping process with K-12 teachers, yielding new prototyping methods and the development of a new form of real-time teacher augmentation called *Lumilo* (*Chapters 4* and *5*).

Part Three focuses on the evaluation of real-time teacher augmentation in live classroom settings. I present and demonstrate a design framework for the iterative, data-informed design and evaluation of real-time teacher augmentation (*Chapter 6*), culminating in an in-vivo classroom experiment that evaluates *Lumilo*'s impacts on teacher and student behavior and students' learning (*Chapters 7* and *8*).

In **Part Four** I begin to explore how the concepts embodied by *Lumilo* might be prepared for wider use, through design studies (*Chapters 9* and *10*) and classroom piloting with teachers and students (*Chapter 10*).

In **Conclusions, Contributions, and Future Directions**, I present methodological reflections and recommendations based on my experiences designing real-time teacher augmentation with and for K-12 teachers (see *Conclusions*), followed by a summary of this dissertation's seven main contributions (see *Summary of Contributions*). Finally, I present a design space and discussion of future directions for educational systems that leverage human/AI complementarity (see *Future Directions*).

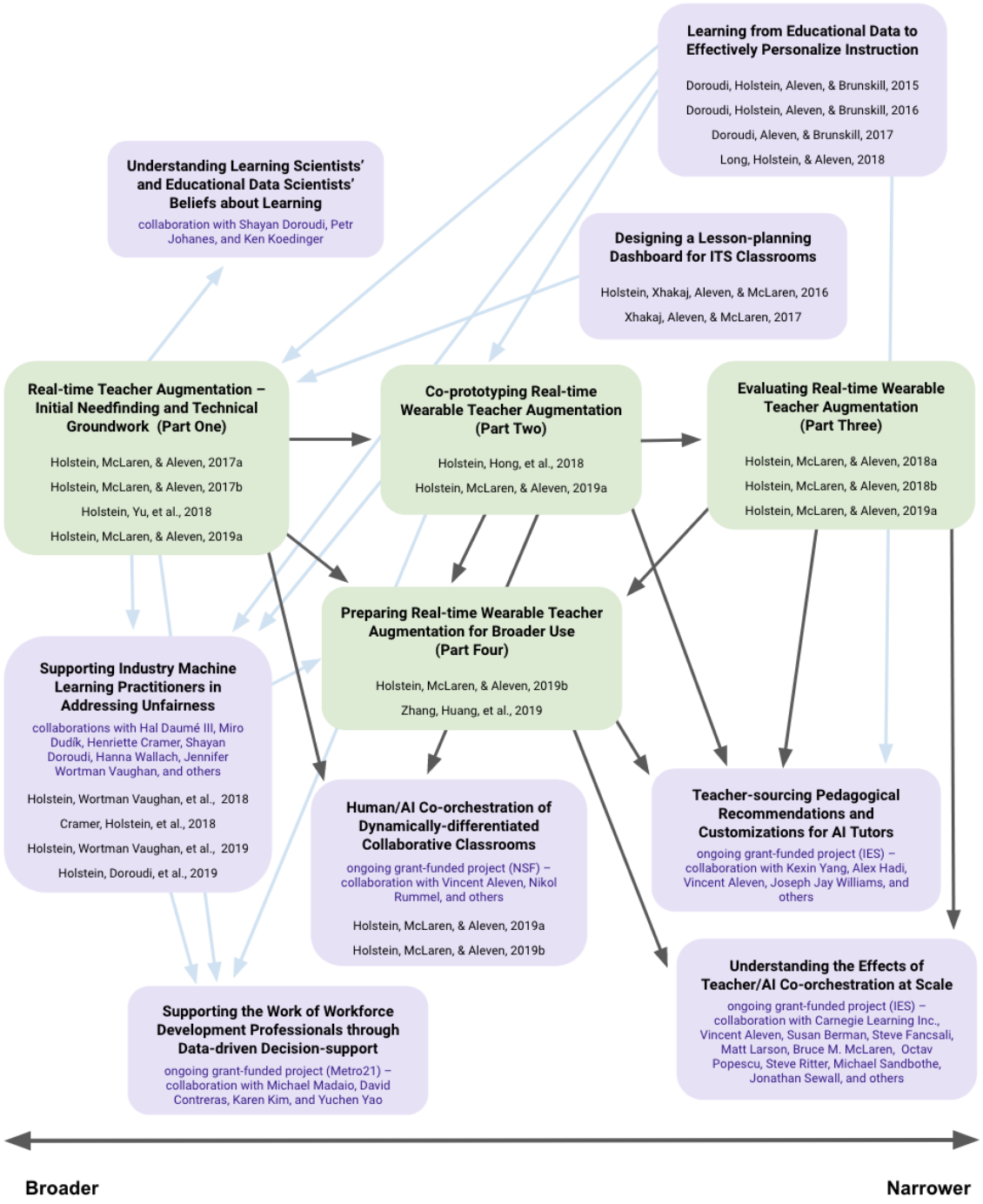

Figure 1. An overview of major research projects or subprojects I have worked on during my PhD, illustrating both breadth and depth. Nodes highlighted in green (the middle cluster) are the focus of this dissertation; others lie beyond its scope. Broader explorations (e.g., needfinding studies) are positioned towards the left side of this diagram, and narrower, more focused investigations (e.g., experimental evaluations) are positioned to the right. Dark arrows indicate strong dependencies (i.e., where one project directly builds upon another). Light arrows acknowledge indirect influences among projects (e.g., where observations and findings from one project help to inform the other).

Part One

Initial Needfinding Studies and Technical Groundwork

In *Part One* of this thesis, I present initial design and data mining explorations and technical groundwork that set a foundation for all of my subsequent research.

In *Chapter 1*, I present initial needfinding studies with K-12 teachers who use AI tutors in the classroom, aimed at understanding the challenges teachers face in orchestrating these personalized classrooms, as well as opportunities to better support teachers.

To complement the investigations presented in *Chapter 1*, and to further inform the design of real-time teacher support tools, I also wanted to better understand the nature of teacher–student interactions in AI-supported classrooms. In *Chapter 2*, I investigate potential relationships between teacher–student interactions and student behaviors and learning in these classrooms, using a new replay-based visualization method called Spatial Classroom Log Exploration (SPACLE).

Finally, in *Chapter 3*, I present an extended version of the *CTAT/TutorShop* architecture for ITS authoring and deployment, which facilitates the rapid development and prototyping of data-driven educational AI applications – including both student- and teacher-facing tools. In turn, this extended technical architecture supported most of the subsequent research presented in this dissertation.

Chapter 1

Exploring K-12 Teacher Needs and Desires for Real-time Analytics in AI-supported Personalized Classrooms

This chapter is based in part on the following publications:

- Holstein, K., McLaren, B. M., & Aleven, V. (2017b). Intelligent tutors as teachers' aides: Exploring teacher needs for real-time analytics in blended classrooms. In *Proceedings of the Seventh International Learning Analytics & Knowledge Conference (LAK 2017),* (pp. 257-266). ACM.
- Holstein, K., McLaren, B. M. & Aleven, V. (2019a). Co-designing a real-time classroom orchestration tool to support teacher–AI complementarity. *Journal of Learning Analytics (JLA)*.

1.1 Background and Motivation

In recent years, many real-time analytics tools have been designed and developed to aid teachers in orchestrating complex technology-enhanced learning scenarios (e.g., van Alphen & Bakker, 2016; Martinez-Maldonado, Clayphan, Yacef, & Kay, 2016; Matuk, Gerard, Lim-Breitbart, & Linn, 2016; Mavrikis, Gutierrez-Santos, & Poulovassilis, 2016; McLaren, Scheuer, & Mikšátko, 2010; Molenaar & Knoop-van Campen, 2017; Tissenbaum et al., 2016). However, design decisions about which analytics to present to teachers often appear to be driven more by the availability of data or pre-existing analytics measures than by an understanding of teachers' actual real-time information needs (Rodriguez-Triana et al., 2017). To the best of my knowledge, no prior work has conducted broad needfinding studies – untethered from specific, pre-existing prototypes – to understand teachers' needs and desires for real-time analytics. Furthermore, work on real-time analytics tools for personalized classrooms has tended to focus heavily on designing tools for use in university contexts, rather than for K-12 teachers (Rodriguez-Triana et al., 2017), and has rarely focused on supporting teachers in personalized, non-synchronous classroom contexts such as ITS classrooms (Holstein, Hong, et al., 2018; Olsen, 2017; but see van Alphen & Bakker, 2016).

In this chapter, I present the first broad investigation in the literature of teachers' challenges and needs for support in AI-supported personalized classrooms *(*see item 1 under *Summary of Contributions – "First broad design exploration of needs for real-time teacher analytics and orchestration support")*. In particular, I focus on classrooms that use intelligent tutoring systems (ITSs): a class of AI-based educational technologies that provide students with step-by-step

guidance during complex problem-solving practice and other learning activities. These systems continuously adapt instruction to students' current 'state' (a set of measured variables, which may include moment-by-moment estimates of student knowledge, metacognitive skills, affective states, and more) (Desmarais and Baker, 2012). Several meta-reviews have indicated that ITSs can enhance student learning, compared with other educational technologies or traditional classroom instruction (e.g., Kulik & Fletcher, 2016). However, design and ethnographic studies have revealed that, in K-12 classroom settings, teachers and students often use ITSs in ways not originally anticipated by system designers (e.g., Holstein et al., 2017a; 2017b; Ogan et al., 2012; 2015; Schofield, Eurich-Fulcer, & Britt, 1994). For example, Schofield et al. (1994) found that rather than replacing the teacher, a key benefit of using such AI tutors in the classroom may be that they free teachers to provide more individualized help while students work with the tutor. Although students in the Schofield et al. study tended to perceive that teachers provide better one-on-one help than an ITS, they also preferred ITS class sessions to more traditional class sessions – in part because of this increase in one-on-one teacher-student interactions.

Despite these benefits, modern ITSs have also been shown to have various limitations (e.g., Beck & Gong, 2013; Kai et al., 2018; Käser et al., 2016; Ogan et al., 2012; 2015). Rich strands of literature in human–computer interaction (HCI) human factors engineering (HF) have studied problems of "task/function allocation" between humans and machines in contexts where automation is helpful yet imperfect (e.g., Horvitz, 1999; Landén, Heintz, & Doherty, 2010; Sujan & Pasquini, 1998; Wickens, Gordon, Liu & Lee, 1998; Wright, Dearden, & Fields, 2000). Yet the question of how best to combine strengths of human and automated instruction has received relatively little attention within the HCI, Learning Analytics, and AI in Education literatures thus far. Over a decade ago, Yacef proposed a reframing of intelligent tutoring systems as "intelligent teaching assistants" (ITAs): systems designed with the joint objectives of helping human teachers teach and helping students learn, rather than only the latter of these objectives as is typical of ITSs (Yacef, 2002). In line with the literature on task/function allocation in human-machine systems, other researchers have since proposed optimizing student learning by leveraging complementary strengths of human and AI instruction (e.g., Baker, 2016; Ritter, Yudelson, Fancsali, & Berman, 2016). That is, ITSs might be more effective if they could adaptively enlist the help of human teachers (cf. Kamar, 2016), in situations teachers may be better suited to handle. While there has been some prior work on real-time teacher support tools for ITS classrooms since the vision of ITAs was introduced (e.g., Feng & Heffernan, 2007; Segedy, Sulcer, & Biswas, 2010), little work has explored teachers' actual needs and desires for such support, or how human instruction might most effectively be combined with AI instruction.

1.2 Overview of Methods

To better understand K-12 teachers' challenges and needs for support in AI-supported personalized classrooms, I conducted a series of formative design studies with 10 middle school

math teachers, across five schools and school districts in Pittsburgh and surrounding areas (see Table 1-1). All participating teachers had previously used some form of adaptive learning software in the classroom, and all but one had previously used an ITS as part of their classroom instruction. As detailed in the following sections, these studies included activities such as love/breakup letters (Hanington & Martin, 2012), directed storytelling (Evenson, 2006), generative card sorting exercises (Cairns & Cox, 2008), semi-structured interviews and field observations (Hanington & Martin, 2012), and speed dating (Davidoff et al., 2007; Zimmerman & Forlizzi, 2017). Choices of design research methods were made iteratively and adaptively, based on our research team's areas of greatest uncertainty at a given stage of the process.

Table 1-1. Demographic information for participating schools.

School	Region	Free/Reduced Price Lunch[1]	# of teachers	# of teachers with $\leq$ 2 years' experience
A	Suburban	18%	1	0
B	Urban	N/A	1	1
C	Suburban	23%	2	0
D	Suburban	29%	4	0
E	Rural	34%	2	1

1.3 Case Study: A Five Year Relationship between Teachers and AI Tutors

To first gain a better sense of key teacher needs that modern ITSs may meet or fail to meet, I conducted semi-structured interviews with two mathematics teachers from a middle school in the Pittsburgh area (school E in Table 1-1) who had previously used an ITS as part of their regular classroom instruction for a period of about five years. These interviews were conducted in the summer of 2016, and incorporated a version of the Love Letter and Breakup Letter design research method, which uses personification as a tool to probe participants' original reasons for adopting a technology (and continuing to use it for an extended period), as well as their reasons for eventually "breaking up" with that technology (Hanington & Martin, 2012). Findings from these interviews are briefly summarized below, charting these teachers' journey from adoption of the technology, to the eventual break-up, and then to the time of the interview.

[1] In the United States, the percentage of students eligible for free/reduced price lunch is often used as a proxy for the poverty rate within a school.

According to the interviewed teachers, teachers at their school originally pushed to adopt the ITS (and its associated curriculum) as part of a broader, teacher-led effort to move away from their existing mathematics curriculum. These teachers felt that the existing curriculum involved too much interleaving of topics, in which *"you never really get fully though a topic the first time, or the second time, or the third time…"*. Instead, they wanted to move to a curriculum that *"[teaches] a topic once, [making sure that] they master it", and then allows students and teachers to move on"*. They found this mastery learning approach to mathematics instruction particularly appealing because they felt it represented a kind of deeper, more focused learning that students would need to do in order to succeed in high school and beyond. As such, teachers at this school were motivated to adopt the ITS (which implemented a mastery learning approach to sequencing problems) in part because they felt it would support their students' transition from shallower and less independent learning (in elementary school) to deeper and more self-regulated learning (in high school).

The interviewed teachers noted that the first year of using the ITS in their classrooms was a challenging adjustment period. Despite the support materials that accompanied the ITS at that time—including a curriculum with associated materials such as textbooks, professional development, and a reporting system that allowed teachers to track their students' progress regularly—these teachers initially struggled to decide how best to monitor and help their students during class sessions in which students worked with the ITS.

In particular, teachers had trouble determining how to assess students fairly and accurately given the self-paced nature of adaptive learning technologies. A major constraint teachers face (in US public schools, at least) is that they need to provide students with letter grades and to communicate their reasons for assigning a particular grade to both students and their parents. During this first year, these teachers often found that it was difficult to justify their decisions to assign students grades based on their progress within the ITS—particularly when communicating with these students' parents. Teachers' grading decisions often involved a considerable amount of subjectivity, as it was often unclear how to balance between grading students based on the progress they've made in the software (i.e. how many units of the curriculum a student has covered), how well students have performed within those units (as shown in the software reports as probabilities that a student has mastered each of a number of fine-grained skills), and how much growth students have shown individually (i.e., change in students' per-skill probabilities of mastery over time).

After the first year using ITSs in their classrooms, teachers began to hold meetings to share reflections, insights, and strategies on how to use the system most effectively. Through these meetings, teachers at this school collectively developed common work practices and grading procedures to help mitigate some of the major challenges they had encountered over the previous year. For example, these teachers developed a uniform grading scheme by setting goals for where

students should be (in terms of the number of units covered) at regular checkpoints throughout each semester. The teachers decided upon these goals by pooling their recollections of the unit that most students had reached in the previous year by particular checkpoints (e.g., by the beginning of each month). If a student was one or more units behind the goal unit, at a certain checkpoint, the teacher would use a control panel in the software to manually push that student forward to the goal unit (and the student would receive no credit for any intervening units) (cf. Ritter, Yudelson, Fancsali, & Berman, 2016a). This was done to keep the whole class relatively synchronized over the course of the year, and to manage the complexity of assigning individual letter grades when different students have covered different amounts of material. Over time, teachers' grading schemes became more nuanced, as teachers would annotate printed versions of the ITS-generated reports with their own observations, collected each day while monitoring their classrooms. They would sometimes integrate these annotations with the ITS-generated metrics to assign grades – allowing for partial credit to be given based on their perception of the student's effort or students' growth over time, rather than just the speed at which they reached mastery.

According to the interviewed teachers, they (and other teachers at the school) ultimately agreed that continued use of the ITS was not worth the cost, for three primary reasons:

1. Challenges of curriculum alignment.

Late in the five-year use period, the school district began a shift to a new mathematics curriculum, and teachers needed to drop the curriculum that came with the ITS. During this time, teachers increasingly found that it was challenging to align the school's new mathematics curriculum with the content and instructional design of the ITS. Yet there was no convenient way for teachers to customize the ITS's content to meet their changing needs. One teacher suggested that the ability to make small customizations to the ITS's problem interfaces (e.g., editing the way math problems were represented, and altering the input format that the ITS would accept from students) would have helped, but only if such customizations could be made with very little investment of time from the teacher.

2. Semi-manual grading and monitoring systems were difficult to maintain.

Although the ITS generated detailed reports about students' progress and performance within the software (e.g. presenting probabilities that a student had mastered fine-grained skills in the curriculum, and reporting on the number of hints a given student had requested), teachers noted that these reports did not provide them with guidance about how to fairly and accurately assign students letter grades based on the data. As such, the teachers felt the need to develop their own grading system, which necessarily balanced efforts to be fair and accurate against teachers' time constraints. Another key limitation these teachers highlighted was that it was not always easy to identify students who were falling behind until it was already "too late" for the student to catch up with the rest of the class. That is, the most salient elements of the reports provided by the ITS tended to be information about the past (e.g. that a student had been overusing the ITS's hints, or that a student had not yet mastered finely-defined skills in the curriculum). But these reports

typically did not provide predictive analytics that could help teachers anticipate problems and proactively intervene. One teacher noted that they would have liked to be able to see the likelihood that a student who had fallen behind the class would actually be able to "catch up" with the other students, if given more time. Without this information, pushing a student forward almost always seemed like the most reasonable decision.

3. Perceived susceptibility of these systems to student misuse.

Some of the teachers in this school perceived that ITSs are particularly susceptible to "gaming" or "cheating" (e.g. abusing the hints that the ITS provides, or solving a math problem through systematic guessing). These teachers worried that, since they had often been unable to catch these behaviors in a timely manner, some of their students had likely wasted a large amount of learning time during ITS class sessions. Prior research supports these teachers' intuitions to a degree: gaming behaviors in ITSs have consistently been shown to have a negative impact on student learning, overall (although not all gaming behaviors are necessarily harmful) (Baker et al., 2013). These teachers were also skeptical that a fully automated mechanism could prevent students from gaming. One of the teachers I interviewed suggested that alerts about such misbehavior, which are easily hidden in large classrooms and computer labs, should be sent to the teacher right away.

Reasons 1 and 2 above align closely with two critical areas that Nye highlighted as under-considered in the literatures on ITSs and AI in education – namely the design of teacher-facing customization and monitoring capabilities (Nye, 2014). Each of these cases can be viewed as an instance of the teacher adapting to the technology, rather than the other way around (Dillenbourg & Jermann, 2010; Xhakaj, Aleven, & McLaren, 2016). The length and difficulty of teachers' adjustment to the use of ITSs in their instruction may also highlight a need for enhanced early support, in the form of improved teacher training tools and peer support systems that facilitate faster sharing of strategies and observations between teachers (as teachers eventually felt the need to band together, but did so only after significant struggle). Teachers' practice of "pushing students forward" when they do not achieve mastery within a pre-specified time interval represents an interesting case, as recent research suggests that such teacher "overrides" of ITSs' mastery learning algorithms may be harmful to student learning over the course of a school year (Ritter et al., 2016a). This simultaneously points to a need for caution in designing such customization and control options for teachers, and a need to better understand the constraints and beliefs that might lead teachers to make such decisions. Although teachers were aware that the practice of pushing students forward before they had mastered the skills in a given unit was counter to the idea behind mastery learning, they continued to do so in order to keep the class relatively synchronized and manage their own orchestration load.

The interviewed teachers also noted that, since discontinuing use of the ITS, they had not adopted any other learning technologies for regular use in their classrooms. They emphasized

that they had used the system for many years because they believed the personalized, detailed, and immediate feedback it provided to students was valuable for their learning. For this reason, they strongly preferred using ITSs to other educational technologies they had tried over the years. The primary obstacles to teachers' continued use of these systems did not lie in the perceived effectiveness of ITSs, but rather in the difficulties that their use in the classroom presented for teachers.

1.4 "Teacher Superpowers" as a Probe to Investigate Perceived Challenges

To encourage teachers to talk freely about challenges they face in AI-enhanced classrooms, without feeling constrained to those for which they believed a technical solution was currently possible, I initially avoided asking direct questions about "analytics" or orchestration tool functionality. Instead, I developed a new probe for teachers: in a series of one-on-one study sessions with five teachers (across schools C and D), I asked,

> *"If you could have any superpowers you wanted, to help you do your job, what would they be?"*

I first posed this question in a very broad sense, but then asked specifically about superpowers that teachers would find useful during classes in which their students work with an ITS or another adaptive learning technology.

In each study session, I asked teachers to immediately write down their "superpower" ideas on index cards the moment they thought of them—pausing ongoing conversation, if need be—to reduce the chance that they would lose track of an idea. In addition to identifying design opportunities within the cards teachers generated, I wanted to get a better sense of teachers' relative priorities among superpowers, and the underlying reasons behind these priorities (e.g., the relative severity of the daily challenges underlying these "superpower requests"). To this end, once a teacher finished generating superpower ideas, they were asked to sort them by subjective priority, while thinking aloud about the reasoning behind their sorting (cf. Cairns & Cox, 2008; Hanington & Martin, 2012).

Teachers were encouraged to generate new cards while sorting, in case the card sorting process inspired new ideas. After a teacher had finished sorting their cards, they were presented with cards generated by all teachers who had participated before them, and were given the option to include any of these cards in their own hierarchy. If a teacher found an idea generated by a previous teacher undesirable, they were instructed to omit that card from their hierarchy. If a teacher felt that a superpower idea generated by a previous teacher was synonymous or redundant with one of their own ideas, they were encouraged to align these cards horizontally, to indicate a "tie." For example, Figure 1-1 shows an excerpt from one hierarchy that emerged from

this iterative card generation and sorting process. One of this teacher's desired superpowers was *"Omniscience,"* which the teacher considered synonymous with *"Being able to see students' thought processes"* (a card that a previous teacher had generated).

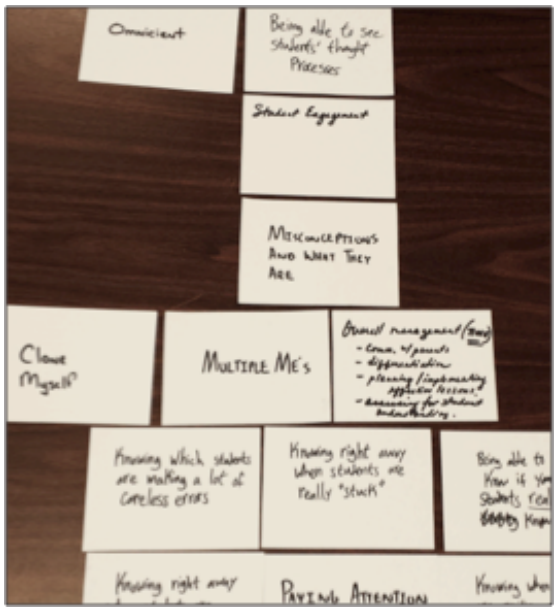

Figure 1-1. Excerpt of a hierarchy produced by one teacher's card sort. Superpower ideas the teacher considered more desirable are placed higher with the hierarchy (from Holstein et al., 2017b).

Figure 1-2 aggregates teachers' pairwise preferences between superpowers. Each row and column of this pairwise comparison matrix displays a superpower that appeared in at least two teachers' hierarchies. Cell shade indicates the number of teachers who ranked the row superpower higher than the column superpower, with darker shades indicating greater agreement (cells on the diagonal represent self-comparisons, and are thus blacked-out). The minimum observed agreement value was 0, and the maximum was 4 out of 5. "Be able to engage students" is highlighted in grey to indicate that this superpower was not present in all five teachers' card stacks. By the time a teacher first generated this card, no redundant cards were available among those generated by previous teacher participants.

Overall, teachers tended to prefer *"Seeing students thought processes"* over most other superpowers, including *"Seeing students' misconceptions."* Some teachers elaborated that if they could really see and understand students' step-by-step reasoning, this would likely reveal students' misconceptions and much more. It is also worth noting that, although estimates of student knowledge (e.g., in the form of probabilities that a student has mastered particular skills) are one of the most central analytics presented by common reporting systems for ITSs (e.g., Heffernan & Heffernan, 2014; Khachatryan et al., 2014; Ritter, Carlson, Sandbothe, & Fancsali,

2015), the superpower *"Knowing whether students really know something"* ranked relatively low compared with most of teachers' other common superpower ideas.

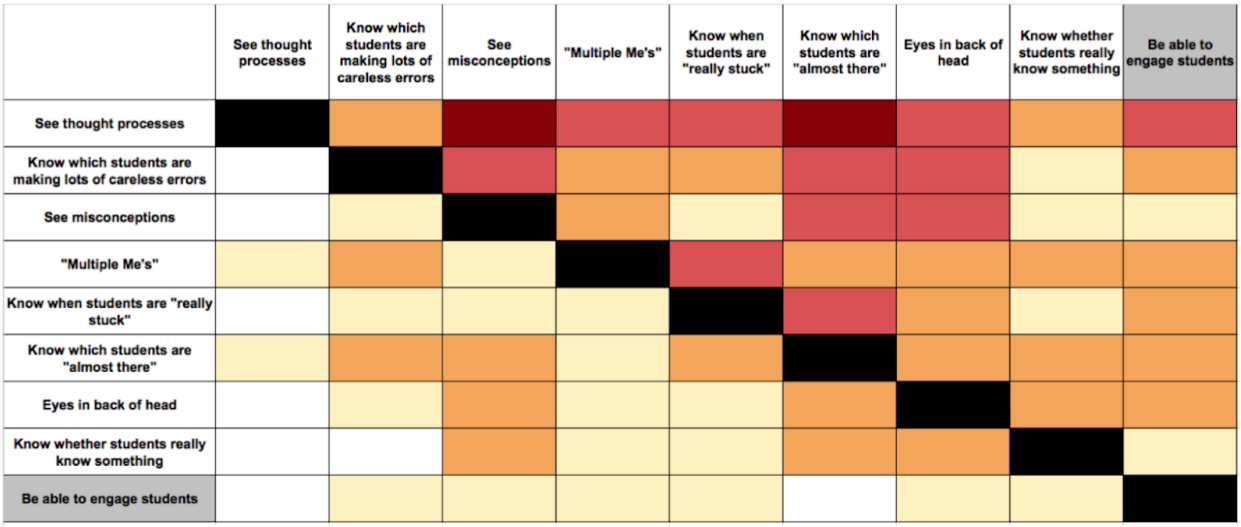

Figure 1-2. Teachers' relative preferences among "superpower" ideas they had generated (from Holstein et al., 2017b).

Across the card hierarchies teachers generated, some interesting regularities emerged. All five teachers wanted the abilities to:

See students' thought processes.
Teachers wanted to be able to see the chains of reasoning that led students from one mathematical expression to the next, without always having to ask students to "show their work," and without having to spend much time deciphering student work. Some teachers explicitly distinguished "seeing thought processes" from simply seeing percentage estimates of student's mastery over certain skills (which they were accustomed to seeing in reports from adaptive learning software they had used previously), noting that such skill mastery estimates were less actionable on their own. That is, if teachers could follow students' thought processes in real-time, this could provide opportunities for them to *"re-route"* students at the moment students *"take a wrong turn"* during a problem solving activity, rather than only providing delayed feedback once the student has moved past the relevant problem.

Know which students are *truly* stuck.
Teachers noted that students often raise their hands during lab sessions when they don't actually need help. At the same time, teachers believed that many students who actually need help the most were the least likely to raise their hands. Being able to see which students actually need the teacher's help, at any given moment, would enable the teacher to better prioritize help across students and *"fight the biggest fires first."*

Know which students are "almost there" and just need a nudge to reach mastery.
Teachers noted that one of the most fulfilling parts of their jobs is *"seeing students to the finish line"*: working with students who are currently on the verge of understanding a new concept, and helping them reach that understanding more quickly. One teacher was initially conflicted over whether to include this superpower in his hierarchy, noting that students in this situation would likely reach mastery even without their help. But this teacher ultimately decided to keep this superpower in the hierarchy, acknowledging that, while he generally tries to spend most of his time working with struggling students, he would find it demotivating to spend *all* of his time doing so.

In addition, four out of five interviewed teachers wanted the abilities to:

Temporarily clone myself (create "Multiple Me's").
Teachers wanted the ability to provide one-on-one support to multiple students simultaneously, rather than leaving real-time personalization entirely to the software. All of the teachers I interviewed reported that, while the level of personalization enabled by ITSs is one of its main attractions, such personalization also makes it more challenging for *teachers* to monitor their students' current activities and provide them with timely feedback.

Have "eyes in the back of my head."
Teachers noted that some students take advantage of the challenges such software poses for classroom monitoring. They shared stories of catching middle school students switching to non-academic websites when they thought the teacher was not watching, but then immediately switching back when they knew they were in visual range. Thus, much of these teachers' attention and energy during an class session is spent "patrolling" the classroom and trying to make sure that everyone is on task.

Detect students' misconceptions.
Similar to teachers' desire to see students' thought processes, their desire to see student *misconceptions* was rooted in the actionability of this information. While teachers viewed *"seeing students' thought processes"* as enabling real-time correction of particular student errors, to help shape students' knowledge of procedures, they viewed *"detecting students' misconceptions"* as enabling the correction of persistent false beliefs that might hinder students' future learning.

Know which students are making lots of careless errors.
Finally, teachers wanted to be able to more easily detect, in real-time, whether students are putting in the effort required to learn. Based on this information, they could decide on a case-by-case basis whether it would be most productive to spend their time providing additional *instruction*, or whether they should instead try to *motivate* the student to put in more effort.

1.5 Directed Storytelling

To more directly investigate teachers' needs for real-time support, I next conducted semi-structured interviews with 10 teachers across 5 schools. In these interviews, I asked teachers to walk me through specific, recent experiences using adaptive learning technologies in the classroom. When teachers brought up frustrations and challenges in the course of their storytelling, they were prompted to reflect on how they thought such systems might be better designed. Teachers were encouraged to imagine that there were no technical limitations, and in particular, no limits on what the system could measure about their students.

Two researchers then worked though transcriptions of approximately 5 hours of video and audio recorded interviews, to synthesize design findings using two standard techniques from Contextual Design: interpretation sessions and affinity diagramming (Beyer & Holtzblatt, 1997; Hanington & Martin, 2012). Interpretation sessions are aimed at helping design teams develop a shared understanding of collected interview and think-aloud data, by collaboratively extracting quotes representing key issues. Affinity diagramming is a widely used, bottom-up synthesis method, aimed at summarizing qualitative patterns across study participants' responses, by iteratively clustering participant quotes into successively higher-level themes (Beyer & Holtzblatt, 1997; Hanington & Martin, 2012). Following several interpretation sessions, the resulting 301 extracted quotes were iteratively synthesized into 40 level-1 themes, 10 level-2 themes, and 4 level-3 themes (see Figure 1-3).

Figure 1-3. A partial view of the affinity diagram, showing teacher quotes within level-1 categories.

The top-level (level-3) themes that emerged through Affinity Diagramming reflected strong desires to maintain control of the classroom, even when students are working with adaptive learning technologies, and to remain an effective force in the classroom, providing value over and above what these technologies can offer students. Quotes under these high-level themes were often accompanied by expressions of anxiety that educational technologists intend to *replace* their roles as teachers, instead of working to *support* these roles. In addition, the top-level themes reflected teachers' desires for analytics that could truly provide information they *did not already know* and teachers' concerns that real-time analytics in the classroom, if not designed carefully, could easily do more harm than good.

Within these top-level themes, teachers' design requirements and opportunities broke down into the following 10 mid-level themes:

Help me to intervene *where, when,* and *with what* I am most needed.
Teachers wanted support in deciding how best to prioritize their time across multiple students who may compete for their attention at once, when to help (or refrain from helping) a given student, and how best to help. Given teachers' limited time during lab sessions, recommendations about how best to help students might come in the form of in-the-moment, personalized advice about effective instructional strategies to use, to address students' specific areas of struggle.

Make sure the technology does not draw my attention away from my students!
Teachers worried that real-time analytics could easily draw their attention away from their students, thus defeating the purpose of using such technologies in the first place. Furthermore, teachers noted that some of the most useful real-time information comes from reading students' body language and other cues that likely would not be captured by a learning analytics dashboard alone. As such, they emphasized that an effective classroom analytics tool would need to be designed to keep teachers' eyes and ears on the classroom to the greatest extent possible.

How can I know whether what I'm doing is *actually working*?
Teachers noted that opportunities to receive immediate feedback on their own teaching are extremely rare. They often worry, especially after seeing students' test scores, that much of what they have taught students over several weeks or months may have had no impact. Observing that intelligent tutoring systems can already track aspects of *students' learning* in real-time, teachers wanted these systems to also provide *them* with timely feedback on the effectiveness of their own help-giving (e.g., one-on-one interactions with individual students or targeted mini-lectures provided to the whole class). Receiving such immediate feedback during a class session could allow them to adjust their instructional strategies on the fly.

Help me understand the "why," not just the "what".
Given how busy teachers are when working with students during ITS lab sessions, they wanted ITSs to provide them with summarized, directly actionable information whenever possible. A

real-time support tool would need to provide concise diagnoses of issues the teacher could act upon. For examples, rather than simply presenting teachers with the observation that a particular student is making frequent errors in the software, it would be valuable to also assist the teacher in determining whether this is due to carelessness or genuine difficulties with the material (and if the latter, to help the teacher diagnose specific areas of difficulty).

I'm just one person: help ease my load.
Teachers emphasized the usefulness of group work and peer tutoring activities in reducing their orchestration load in the classroom. Some teachers suggested that one way an ITS could help them during a lab session would be to recommend groups of students who are likely to be able to help one another (perhaps adaptively matched by the ITS based on its knowledge of their current mastery of specific skills). This would lift some of the responsibility of helping students from the teacher's shoulders, and also enable the teacher to work with a larger number of students who may be struggling with similar issues, by meeting with groups rather than individuals.

But how do I judge whether my students are *really* doing well?
Teachers wanted more support from the ITS in determining what constitutes "good" performance (e.g., is a 70% probability of mastery below or above "average" for a particular skill and amount of practice?).

Help me monitor and manage student motivation.
Teachers noted that it would be useful to have real-time analytics about their students' motivation and affective states in the classroom, not just analytics about student learning and performance. Receiving real-time notifications about student frustration, for example, could allow teachers to intervene before students became too demotivated.

What can you tell me about my students that I do not already know?
Teachers complained that reporting systems they had used in the past tended to provide them with a lot of unsurprising information about their students. Teachers wanted ITSs to take into account what they already knew about their students (e.g., *"[this student] is going to make slower progress, but that's only because she's so deliberate"*), and provide them with notifications only in cases that conflict with their expectations.

Allow me to customize the technology to meet my needs. Teachers emphasized that, in cases where an ITS's instructional design differs in some way from their own pedagogy (e.g., when the mathematical notation the teacher uses in their lectures differs from that the ITS will accept from students), teachers should be able to quickly and easy adapt the software to meet their needs.

Allow me to override the technology. In addition to customization, teachers also wanted the ability to take control of the ITS on-demand. For some teachers, this simply meant being able to "freeze" all of their students' screens while giving an impromptu lecture in the midst of a lab session, to ensure they had students' attention. For others, this meant being able to load a "quiz

problem" on all students' screens, to quickly assess the effects of a whole-class lecture on students' knowledge of particular skills.

1.6 Exploring Possible Classroom Futures Through Speed Dating

To further understand and validate needs teachers had revealed through the "superpowers" exercise and directed storytelling sessions I adopted a "speed dating" approach, presenting teachers with hypothetical classroom scenarios inspired by these needs. Speed dating is an HCI method for rapidly exploring a wide range of possible futures with users, intended to help researchers/designers elicit unmet needs and probe the boundaries of what particular user populations will find acceptable (which otherwise often remain undiscovered until after a technology prototype has been developed and deployed) (Davidoff et al., 2007; Zimmerman & Forlizzi, 2017). In speed dating sessions, participants are presented with a number of hypothetical scenarios in rapid succession (e.g., via storyboards) while researchers observe and aim to understand participants' immediate reactions. In addition to revealing ways technology concepts may cross boundaries of acceptability, this method can lead to the discovery of unexpected design opportunities when anticipated boundaries are found not to exist or when unanticipated needs are discovered. Importantly, speed dating can often reveal needs and opportunities that may not be observed through field observations or other design activities, such as those described in *Sections 1.3* through *1.5*.

I met with five teachers from my previous interviews, and presented them with futuristic classroom scenarios inspired by needs they had previously expressed. Teachers were presented with eleven storyboards. Each storyboard presented a scenario intended to probe the boundaries of acceptability, generated based on teachers' most commonly requested "superpowers," themes from the earlier directed storytelling interviews (*Section 1.5*), and notes from field observations in teachers' classrooms (Holstein et al., 2017a; 2017b). Key findings from these speed dating sessions are summarized below.

Despite teachers' expressed desire for real-time support in prioritizing their time across multiple students during a class session, teachers consistently rejected the idea of "time management" systems that explicitly nudge them to spend less of their time with certain students (e.g., those who seem to be doing well without the teacher's help) and more of their time with others (who may benefit from more assistance). For example, one teacher reacted strongly to this concept, stating,

> *"I don't need that… to remind me it's time to move on. I know that. As an educator, you know when you've got other kids to deal with."*

"If I know that I lectured and after I lectured, now my misconceptions decrease significantly. Then I know, okay we're on the right track."
- Teacher 3

"Yeah, all I care about... it's all about the kids. I don't need rewards and stickers and points for what I am doing. Like, that's my job."
- Teacher 8

"...Are you kidding me? This is his heart beating and sweating. Oh my god... [But] part of me thinks, 'It would be nice to know how nervous they are,' because I was a very nervous math student back in the day."
- Teacher 9

"...Could I, like use [the drone] to zap my students if they're misbehaving? Nothing that would actually hurt them... a light jolt"
- Teacher 6

"I love this because at a snapshot I see who's struggling. I don't need to go from student to student to student..."
- Teacher 3

"There are always students who are shy and just don't raise their hands... [Some] raise their hands when they really don't need help."
- Teacher 6

Figure 1-4. Examples of concepts explored with teachers and selected reactions. Left: examples of panels from speed dating storyboards; Right: selected excerpts from teacher reactions to the illustrated scenarios.

Although recent research suggests that teachers' intuitions about which students need the most help during personalized lab sessions can be limited (Holstein et al., 2017a; 2018a; 2018b), this teacher's comment reflects a core tension in the design of real-time, intelligent teacher supports. Teachers' comments in response to this storyboard suggested that such "time management" systems can be undesirable if they threaten teachers' autonomy in the classroom and come off as "judging" teachers based on their current activities. These systems can also be undesirable if they remove teachers' ability to choose, on a case-by-case basis, between two conflicting desires

during lab sessions (corresponding to two of the superpower ideas teachers generated): helping students who are "almost there" versus helping struggling students who may be most in need of help.

By contrast, teachers were highly receptive to technology designs that presented information to help them prioritize their time among students, without attempting to directly prescribe specific actions. For example, the bottom panel in Figure 1-4 comes from one of the most positively received storyboards. This panel shows a heads up display, through which an ITS can inform the teacher that a given student may need their assistance (even if the student is not necessarily aware that they need help), without explicitly suggested that the teacher should be helping a particular student at a particular moment. As shown to the right of this panel in Figure 1-4, some teachers noted that such a tool would be particularly helpful because students' hand raising behavior can be an unreliable indicator of their actual need for help.

A key reason teachers gravitated towards the concept of a heads up display was that, by displaying analytics directly overtop their view of the classroom, this technology would not draw their attention away from the classroom itself. Teachers' reactions to this and other storyboards, including a smartwatch-based classroom analytics tool, suggested that they might be quite open to and even *prefer* wearable interfaces for real-time use cases, as opposed to handheld displays such as tablets and mobile phones. In particular, teachers saw heads-up displays as an opportunity to have their own "private" smart classroom with analytics only they could see, preserving privacy between students.

1.7 Conclusions

In this chapter, I have presented the first broad investigation in the literature of teachers' challenges and needs for support in AI-supported personalized classrooms. To the best of my knowledge, no prior work has conducted broad needfinding studies – untethered from specific, pre-existing prototypes – to understand teachers' needs and desires for real-time analytics and orchestration support *(*see item 1 under *Summary of Contributions – "First broad design exploration of needs for real-time teacher analytics and orchestration support")*. Furthermore, work on real-time analytics tools has tended to focus heavily on designing tools for use in university contexts, rather than for K-12 teachers (Rodriguez-Triana et al., 2017), and has rarely focused on supporting teachers in personalized, non-synchronous classroom contexts such as ITS classrooms (Holstein, Hong, et al., 2018; Olsen, 2017; but see van Alphen & Bakker, 2016).

Working with 10 middle school math teachers, across five schools and school districts in Pittsburgh and surrounding areas, I have explored how intelligent tutoring systems (ITSs) might be better designed to work together with human teachers during a class session. This work has identified several design opportunities for real-time teacher support in AI-supported K-12 classrooms.

For example, through semi-structured interviews with middle school teachers who have worked with existing AI tutoring systems in their classrooms (see *Sections 1.3* and *1.5*), I have identified opportunities for these systems to better support them in fairly and accurately assessing their students' performance within the software. These interviews also suggested opportunities for predictive analytics to aid teachers in making challenging decisions, such as whether and when to override ITSs' built-in mastery learning algorithms in order to keep slower-moving (perhaps struggling) students relatively in pace with the rest of the class (a challenge revisited in *Chapter 9* of this dissertation).

Through card sorting and directed storytelling exercises (see *Sections 1.4* and *1.5*), I identified design features and requirements for real-time analytics tools that may help address some of the greatest challenges teachers face in such personalized classrooms. Importantly, these findings suggest that the analytics commonly generated by existing teacher dashboards and reporting systems for ITSs and other personalized learning technologies rarely align with those that teachers expect to be most useful and actionable *on-the-spot*, during an ongoing class session (Holstein, et al., 2017b; 2019a). For example, although estimates of student knowledge (e.g., in the form of probabilities that a student has mastered particular skills) are one of the most central analytics presented by common reporting systems for ITSs (e.g., Feng & Heffernan, 2007; Khachatryan et al., 2014; McGraw Hill, 2019; Ritter et al., 2007) this was not strongly favored by teachers relative to other real-time "superpower" ideas such as seeing student misconceptions or knowing when students are "really stuck" and may need human help (cf. Rosé, McLaughlin, Liu, & Koedinger, 2019).

By presenting teachers with a range of possible futures, through Speed Dating (see *Section 1.6*), I found that K-12 teachers were highly receptive to the concept of intelligent classroom tools that *support them in deciding* how best to allocate their time and attention across students during a lab session. At the same time, however, teachers recoiled at the idea of such a system providing *explicit, unsolicited recommendations* for how they could better allocate their time – especially when such recommendations were perceived as negatively "judging" teachers' current choices. I do not interpret these findings to mean that real-time teacher support tools should avoid making recommendations for action. Indeed, given previous findings that teachers sometimes make decisions that are suboptimal or even harmful to students' learning with ITSs (e.g., Ritter et al., 2016a), I suspect that such directness may sometimes be important in guiding teachers towards more effective interventions – perhaps especially in real-time usage scenarios, where teachers may have scarce time or motivation to pore over data visualizations and draw their own inferences. Rather, I believe that these findings highlight a delicate tension between learning technology designers' desire to "nudge" teachers towards instructionally effective patterns of behavior, on the one hand, and the need to privilege teachers' autonomy and rich prior knowledge, on the other – paralleling findings in other domains where intelligent systems are developed to support human experts' decision-making, such as healthcare (e.g., Yang,

Zimmerman, Steinfeld, Carey, & Antaki, 2016; Yang, Steinfeld, & Zimmerman, 2019). It may be, for example, that teachers would be more receptive to more explicit and direct action recommendations if these were presented only upon a teacher's request, rather than in the form of automated alerts (see *Chapters 9* and *10*). This question, and the broader question of how teacher support tools can achieve an effective balance between augmenting teachers' *awareness* in the classroom (Rodriguez Triana et al., 2017; Sherin, Jacobs, & Philipp, 2011) and *more directly supporting their decision-making* (Borko, Roberts, & Shavelson, 2008; Holstein, 2018; Schoenfeld 2008; 2010) remain interesting open questions for future design and experimental research (cf. An, Bakker, Ordanovski, Taconis, Paffen, & Eggen, 2019; Holstein, 2018; Holstein et al., 2018a; 2019a; 2019b; van Leeuwen et al., 2018; Ritter et al., 2016a).

Taken together, these findings provide novel insights into teachers' needs for real-time support in orchestrating AI-supported, personalized K-12 classrooms, which may inform the design of future classroom technologies. I expect that many of the findings presented in the chapter may generalize to other personalized classroom contexts, including contexts in which students do not interact with AI tutoring software, or even with "computers", per se (cf. Martinez-Maldonado, Echeverria, Santos, Santos, & Yacef, 2018).

Parts Two through *Four* of this thesis build upon findings from this initial needfinding phase, along with outcomes from the classroom data mining and technical groundwork presented in the remainder of *Part One*. The next chapter, *Chapter 2*, complements the investigations in the present chapter with exploratory data analyses – using classroom log data and field observations from live K-12 classrooms – of relationships between out-of-software, teacher–student interactions and students' learning and behavior within AI tutoring software.

Chapter 2

Investigating Relationships Between Teacher Attention, Student Behavior, and Learning in Personalized Classrooms

This chapter is based in part on the following publications:

- Holstein, K., McLaren, B. M., & Aleven, V. (2017a). SPACLE: Investigating learning across virtual and physical spaces using spatial replays. In *Proceedings of the Seventh International Learning Analytics & Knowledge Conference (LAK 2017)*, (pp. 358-367). ACM

2.1 Background and Motivation

In Chapter 1, I explored teachers' perceived challenges and needs for real-time support in AI-supported classrooms. To complement these investigations, and further inform the design of real-time teacher support tools, I also wanted to better understand the nature of teacher–student interactions within these AI-supported classrooms. In particular, I sought to explore how out-of-software, teacher–student interactions might relate to students' learning and behavior within AI tutoring software. Although student behavior in AI-supported classrooms has been extensively studied using log data from educational software, the impacts of students' out-of-software interactions (such as face-to-face teacher–student or peer tutoring interactions) are rarely studied. Yet such out-of-software interactions may play a critical role in mediating these technologies' effectiveness (Miller, et al., 2015).

In this chapter, I focus on classrooms using intelligent tutoring systems (ITSs). Although ITSs are often designed for use in K-12 classroom contexts, classroom studies that evaluate the effectiveness of these systems do not typically examine effects of out-of-software, human-to-human interactions (Liu, Davenport, & Stamper, 2016). For example, prior field studies suggest that a large proportion of K-12 students' help-seeking behavior in ITS classrooms may occur entirely outside of the software. Yet the existing ITS literature has focused on the effects of students' within-software help-related behaviors (e.g., hint requests) rather than out-of-software behaviors (e.g., asking a teacher for help) (Aleven, Roll, McLaren, & Koedinger, et al., 2016; Ogan et al., 2012; 2015). "In-vivo" classroom studies aim to study the effectiveness of ITSs in the presence of contextual variables that are likely to be present in real-world classroom contexts (e.g., help from a teacher or peer, external distractions affecting individuals

or groups of students, collaboration between students, etc.). Yet they do not typically measure the effects of the contextual variables themselves, instead treating these as background noise (e.g., Aleven, & Koedinger, 2002; Koedinger, Aleven, Roll, & Baker, 2009) (but see Liu, et al 2016; Baker, Corbett, Koedinger, & Wagner, 2004).

There is reason to expect, however, that some of these contextual variables may be important mediators of student learning (Miller, et al., 2015). In particular, gaining a better understanding of the effects of teacher–student student interactions in ITS classrooms may be critical to understanding these systems' effectiveness in real-world contexts. For example, a large-scale, two-year evaluation study of Carnegie Learning's Algebra I tutor suggested that variability in the out-of-software support teachers provided to students may have been at least partly responsible for inconsistent results across evaluation years (Pane, Griffin, McCaffrey, & Karam, 2013). Similarly, recent work has found that the extent to which teachers override ITSs' built-in, mastery learning based problem selection may negatively impact student learning (Ritter et al., 2016a).

This chapter introduces a new replay-based visualization method, Spatial Classroom Log Exploration (SPACLE), which aims to facilitate the discovery of relationships between out-of-software, human–human classroom interactions and student learning within educational software (see item 4 under *Summary of Contributions – "Novel design and prototyping methods"*). Through exploratory data analyses using these spatial replay visualizations, combined with causal data mining, I find suggestive evidence that students' mere awareness that they are being monitored by a teacher may contribute to greater student engagement and learning – perhaps in part by reducing "gaming the system" behaviors (an interpretation further supported by findings from an in-vivo classroom experiment, presented in *Chapter 7*).

While prior work (e.g., Stang & Roll, 2014) has investigated associations between teachers' monitoring behaviors and physical movement in co-located classrooms (e.g., looking over students' shoulders as they work) and students' behaviors and learning outcomes, using observational data, the work presented in this chapter is the first to investigate such associations in the context of AI-supported classrooms (see item 5 under *Summary of Contributions – "First investigations of relationships between teacher movement/monitoring and student behavior and learning in AI-supported classrooms"*).

2.2 Spatial Classroom Log Exploration (SPACLE)

Even when an "in-vivo" classroom study is primarily designed to test preconceived hypotheses (e.g. to test the effectiveness of a particular technology design), researchers sometimes collect qualitative classroom observations during the course of the study. These observations can allow researchers to gain a richer picture of what went on during a given class session, which may later help in interpreting and explaining study results. Such classroom observations can lead to

unexpected discoveries, which can later be investigated more thoroughly through targeted follow-up experiments or through offline data analyses (Liu, Davenport, & Stamper, 2016; Liu, Stamper, & Davenport, 2018). For example, classroom observations of the ways students misuse ITSs inspired a line of experimental and data mining work dedicated to uncovering the underlying causes behind these behaviors, as well as design work dedicated to intervening on these underlying causes (Baker, 2011). Similarly, my thesis direction was originally inspired by informal classroom observations of teachers' on-the-spot interactions with their students, in the context of in-vivo classroom experiments that were not intended to study (and did not explicitly consider the effects of) teacher–student interactions (Doroudi, Aleven, & Brunskill, 2017; Doroudi, Holstein, Aleven, & Brunskill, 2015; 2016; Long, 2015).

To extend this observation process, I developed a new replay visualization method, instantiated as a prototype tool[2], called Spatial Classroom Log Exploration (SPACLE). SPACLE visualizations replay moment-by-moment analytics about student and teacher behaviors in their original spatial context (e.g., overlaid upon a classroom seating chart, as in Figure 2-1). These visualizations enable researchers to interactively re-examine classroom ITS-use sessions within a virtual map of the classroom layout (cf. Vatrapu, Kocheria, & Pantazos, 2013), while visualizing moment-by-moment analytics about individual students. SPACLE replays are multimodal in the sense that they combine multiple data streams – visualizing both analytics about students' out-of-software interactions (e.g., whether or not a student is raising her/his hand, talking to a peer, or talking to the teacher), and analytics generated from students' interactions within the software, such as whether students are inactive, abusing the tutor's help functions (Aleven, Roll, McLaren, & Koedinger, 2016) making frequent careless errors (San Pedro, Baker, & Rodrigo, 2011), "stuck" on a current activity (Beck & Gong, 2013; Käser, Klinger, & Gross, 2016), confused, frustrated, or engaged in their current task (Baker et al., 2012; Liu, Pataranutaporn, Ocumpaugh, & Baker, 2013).

In each replay session, the SPACLE prototype allows researchers to specify the analytics – which can be implemented as custom plugin scripts – that they would like to examine about the teacher, the students, and/or any summary information they would like to display at the class level (e.g. the percentage of the class that is "stuck" on their current task at a given time). Then, given a map of the classroom layout where observations took place, as well as a mapping from student identifiers to their seating positions within the classroom (both of which may be obtained, in approximate form, by asking a teacher to provide a printed or hand-drawn copy of the seating chart), SPACLE can generate visual replays that preserve potentially important spatial information. Specifically, researchers can import a class roster and an image (e.g., a scanned drawing) of a classroom layout into SPACLE, and then construct a virtual map of the classroom within the interface, by dragging, rotating, and resizing graphical representations of students (which are automatically generated, and pre-labeled, based on the class roster) into place, using

[2] *https://github.com/d19fe8/SPACLE*

the image as a guide. Each student is represented as a small circle with a rectangle directly above it (representing the student's computer screen) and a name or other identifier directly below it.

Figure 2-1. A sequence of screenshots from a replay of an ITS class session generated using SPACLE (figure from Holstein et al., 2017a). In the displayed classroom, there is a long vertically-oriented row of desks in the center of the room, and several horizontal rows on either side of it. Students' idle time in the ITS – ranging from 30 seconds or less (black) to 90 seconds or more (bright green), is visualized on their "computer screens". The teacher's position in the classroom is indicated by a circle that takes on colors representing the teacher's current activities (orange: on-task conversation with a student, blue: inactive/distracted or off-task conversation). In the final panel, as the teacher spends time away from students, in the back of the classroom, several students in the class stop working in the software for extended periods of time.

Researchers can then choose to visualize moment-by-moment analytics about students by assigning certain analytics to appear either in students' circles (e.g., to visualize out-of-software behaviors such as hand raising), or on their "computer screens" (e.g., to visualize analytics about students' within-software interactions). In addition, if analytics about teacher behavior are present in a synchronized dataset, these can be visualized via a free-floating circle, which can change position on the map to represent the teacher's location in the classroom at a given time. Aside from teacher position, all other analytics in the SPACLE prototype are visualized through color. For continuous-valued or ordinal analytics, colors can be assigned to two arbitrary end

points within the range of values a given metric can assume, and these analytics will be visualized by interpolating between the two colors. For categorical analytics, colors can be assigned individually to different categories. Figure 2-1 shows a series of screenshots from a replay session (showing time slices several minutes apart). In this replay, the time elapsed since each student's last within-software interaction is displayed on their "computer screens", with end points of 30 seconds and 90 seconds. If a student has spent 30 or fewer seconds inactive, that student's screen will appear black, and if the student has spent 90 seconds or more inactive, the screen will appear bright green. In between 30 and 90 seconds of inactive time, a student's screen will appear to gradually transition from black to green. The teacher's position and current activities are also visualized in this replay, with "on-task conversation" indicated by an orange circle, and "inactive/distracted or off-task conversation" indicated by a blue circle. In this example, it is striking to see the amount of inactivity in the third frame, during a period when the teacher is inactive, and standing in the back of the classroom.

By examining a limited set of variables within a single replay session, SPACLE visualizations may support researchers in detecting qualitative patterns across multiple students more rapidly than would be possible by watching video recordings or conducting live classroom observations (Baker, Saxe, & Tenenbaum, 2005; Borko, Roberts, & Shavelson, 2008). In addition, by visualizing different sets of analytics across multiple replay sessions, researchers can iteratively explore questions about potential mediators of student learning and behavior within the software (cf. Harpstead, 2017). After formulating hypotheses based on replay analyses of a small number of classrooms, researchers can investigate further through quantitative modeling on larger samples.

The SPACLE prototype is currently designed to work with ITS log data from DataShop, a widely-used educational data repository (Koedinger et al., 2010). Prior to generating replays, this prototype first synchronizes records of out-of-software events in the classroom (e.g. student and teacher behaviors, or class-level disruptions) with log data that is automatically generated from students' interactions within the software. The records of out-of-software behaviors may be generated by hand (i.e., field observations conducted by human observers), or, in the future, via automated means using classroom sensors (e.g. An, Bakker, Ordanovski, Taconis, & Eggen, 2018; Martinez-Maldonado, Echeverria, Santos, Santos, & Yacef, 2018; Prieto, Sharma, Dillenbourg, & Rodriguez-Triana, 2016; Raca & Dillenbourg, 2013) or machine-learned detectors that attempt to infer out-of-software behaviors from ITS log data (e.g. Miller et al., 2015). Indeed, in *Chapter 4* of this dissertation, I describe a "mixed reality sensor" approach used in my subsequent work to collect such synchronized, out-of-software interaction data automatically (Holstein, Hong et al., 2018; Holstein et al., 2019a). The primary requirements the SPACLE prototype imposes on these out-of-software logs are that they either include continuous measurements (e.g. moment-by-moment recordings of a teacher's location and movements in the

classroom) or discrete observations marked with approximate start and end times for a given behavior.

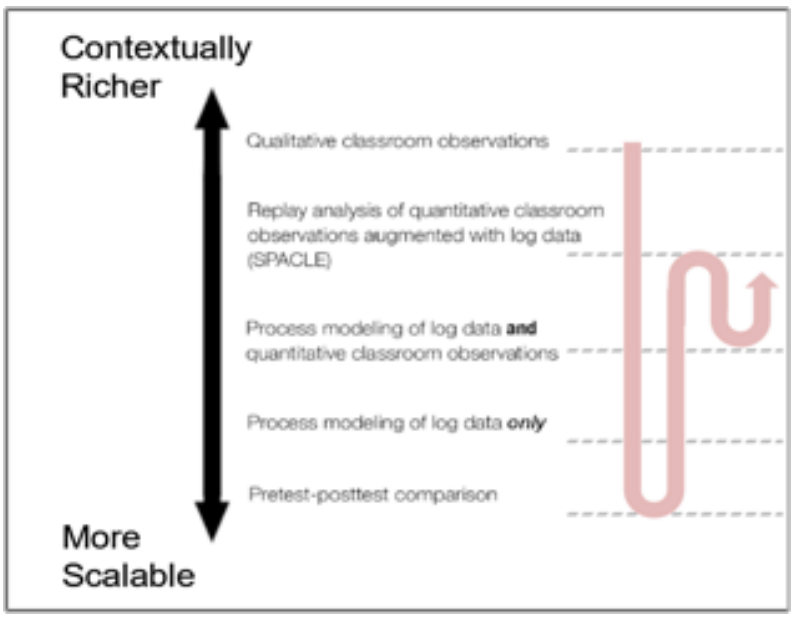

Figure 2-2. A visual representation of an exploratory data analysis path, using a combination of spatial replay visualizations and other data analysis methods.

In my work thus far, I have primarily used these spatial replay visualizations to better understand and interpret the effects of out-of-software classroom dynamics on student learning with ITSs. Although not the focus of this chapter, I have also begun to explore broader uses of SPACLE replays as a design probe for teachers (see *Chapter 5* for related explorations). Figure 2-2 illustrates an example of an exploratory data analysis path using SPACLE. In this example, researchers first run an in-vivo classroom study to evaluate the effectiveness of an educational technology. While running this study, researchers may collect qualitative classroom observations that can help guide and constrain future exploratory analyses. After the study, researchers statistically compare students' pre- and posttest scores, finding an overall effect of the technology on students' learning outcomes. To explore whether and how out-of-software interactions may have *mediated* student learning within the software, researchers conduct in-depth exploratory analyses with spatial replays, using data from a small sample of classrooms. Researchers then use observations from replay analyses to guide the development of (or search for) quantitative models that relate within- and out-of-software classroom phenomena (cf. Kery & Myers, 2017; Tukey, 1977). The fit of the resulting models is evaluated on a larger, held-out dataset. If multiple models fit the data, researchers may return to richer qualitative analyses with spatial replays, to support evaluation among competing hypotheses.

In the following sections, I illustrate how I have used SPACLE visualizations as a bridge between qualitative analysis of classroom observation data and larger-scale data mining, in my own early investigations into potential effects of teacher behavior in ITS classrooms.

2.3 Case Study: Data Collection

My collaborators and I collected the data reported in this case study during a classroom experiment aimed at evaluating how analytics generated from students' interactions with an ITS, presented on a prototype teacher dashboard after class, could help teachers plan more effective lectures for subsequent class sessions (Xhakaj, Aleven, & McLaren, 2017). However, the data analyzed in this paper are from a class period during which students worked with ITSs and teachers *did not* yet have access to a dashboard. Thus, these teachers often relied on direct observations of their students' computer screens, while walking around the classroom, in order to monitor their students' progress. This is a typical situation when teachers use ITSs in their classes (d'Anjou, Bakker, An, & Bekker, 2019; Holstein et al., 2017b; 2019a; Schofield et al., 1994).

In this study, 299 middle school students used *Lynnette*, an ITS for algebraic equation solving (Long & Aleven, 2013; Long & Aleven, 2016; Long, Holstein, & Aleven, 2018), for 60 minutes, spread across up to two class sessions. Students' performance in equation solving was measured before and after using *Lynnette* via computer-based pre- and post-tests, which were focused on measuring procedural skills. We used two test forms, which differed only by the particular numbers used in equations. Test forms were presented in counterbalanced order across pre- and post-test. Test items were graded automatically, based on the correctness of students' final responses (i.e. without providing partial credit for intermediate steps in equation solving).

I collected live classroom observations from a sample of 9 out of 17 classrooms, taught by 4 teachers with a total of 151 students. Students who were absent during any of the pretest, ITS-use sessions, or posttest were excluded from subsequent analyses, leaving 135 students in total. In the remainder of this paper, only data from these 135 students are considered. Due to privacy concerns, it was not possible to collect audio or video data during class sessions. Instead, during each class session, a member of our research team sat in the back of the classroom (in order to minimize any disturbance caused by their presence) and recorded coarse-grained field observations of teacher and student behavior. I recorded observations using LookWhosTalking[3], a tool for coding live classroom observations, which I customized with a coding scheme developed to facilitate both coding and eventual analyses. This coding scheme was adapted from the Baker-Rodrigo observation method protocol (BROMP) (Ocumpaugh, 2015), and the TA observation protocol developed by Stang & Roll (2014). This coding scheme extends the TA

[3] *https://bitbucket.org/dadamson/lookwhostalking*

observation protocol by distinguishing between different types of teacher interactions with students – namely, distinguishing whether a teacher is monitoring/observing a student or holding a conversation with that student, and further distinguishing between on-task and off-task teacher-student conversations (cf. Baker et al., 2004; Ocumpaugh, 2015).

Following BROMP, I asked teachers to provide up-to-date seating charts prior to each class session, both to enable coding of student–teacher interactions during class, and for use as classroom maps during replay analysis (Ocumpaugh, 2015). Field observers recorded instances in which students raised or lowered their hands, and coded teacher behavior with reference to 6 broad categories:

1. On-task conversation: The teacher is engaged in a discussion with a student about the activity they are currently working on

2. Off-task conversation: The teacher is engaged in an unrelated discussion with the student.

3. Talking to class: The teacher is addressing the entire class (e.g., giving a "mini-lecture" based on observations made during a lab session)

4. Monitoring: The teacher is watching the class from a fixed location (e.g., the teacher's desk), or standing behind a student and scanning that student's computer screen over their shoulder (disambiguated by the teacher's current location, as described below)

5. Outside the room: the teacher is not in the classroom

6. Inactive: the teacher is in the classroom, but engaged in an activity other than one of the above (e.g., grading papers or checking email)

Within each of the broad behavior categories above, the position of the teacher in the classroom was recorded if the behavior persisted for at least two seconds. The teacher's position was coded either as the name of a student the teacher was standing behind (e.g., if the teacher was monitoring or conversing with that student), or a description of another location in the classroom, such as the teacher's desk. These field observations were then synchronized offline, using the SPACLE prototype, with the DataShop log data generated from students' interactions with *Lynnette*.

2.4 Case Study: Analyses and Results

Pre-post analysis

A student's prior knowledge of equation solving (as measured by the pretest) was a strong predictor of their posttest score ($r = 0.79$, $p < .001$). Students went from an average of 43% on the pretest to 52% on the posttest – a significant improvement ($F(1, 133) = 17.66$, $p < .001$).

Replay Analysis

On average, teachers spent roughly 47% of their time either inactive or outside of the room. The proportions of time teachers were observed engaging in each of the other coded activities, within the remainder of the time, are reported in Table 2-1.

In examining replays of a small number of class sessions, I observed a number of unexpected patterns – often re-running the replay with different combinations of analytics in order to explore particular questions more deeply. Almost immediately, I noticed that teachers tended to actively monitor their students in concentrated bursts, interleaved with (often lengthy) idle periods in which the teacher might either monitor the whole class from a fixed position in the room, or attend to an unrelated activity. During periods in which teachers were walking around the classroom, they occasionally provided students with apparently unsolicited feedback (i.e. feedback that was not preceded by the student raising their hand) based on their observations while watching a student's computer monitor over their shoulder.

Table 2-1. Frequency of coded teacher and student behaviors during teachers' active time. Top row: average percentage of teachers' active time that was spent engaged in each of the coded behavior categories. Bottom row: average percentage of students for which a category was observed at least once.

	Teacher–student: On-task conversation	**Teacher–student: Off-task conversation**	**Teacher: Talking to class**	**Teacher: Monitoring**	**Student: Hand-raising**
Teacher time	33%	19%	4%	44%	N/A
% of students	28%	7%	N/A	34%	26%

In these replays, teachers appeared to selectively monitor certain students while consistently passing others by. In interviews with some of these teachers, they noted that they monitor their students strategically during computer lab sessions, relying on prior knowledge about their students' abilities and behavioral tendencies. In particular, two of the teachers I interviewed emphasized that they tend to focus on monitoring students who they expect are more likely to be off-task (e.g. browsing external websites instead of working with the software). However, replays displaying the amount of time each student spent inactive in the software suggested that teachers tended to neglect certain regions of the classroom, overlooking students who may truly tend towards greater time off-task (Holstein 2017a). This may be viewed as early, suggestive evidence that teachers' intuitions can be limited when it comes to judging which students are

more likely to engage in off-task behavior. However, it is also possible that students sitting in regions of the room where a teacher is more active are *more likely* to remain on-task. Indeed, replay analyses lent some support to this interpretation, as students frequently appeared to go off-task when the teacher moved to another region of the classroom, but then resumed working with the software once the teacher started moving in their general direction. Similarly, many students appeared to go off-task during periods in which the teacher was either inactive or outside of the room (see Figure 2-1).

A major takeaway from these replay analyses was that prior work may have underestimated the importance of spatial factors in the classroom when analyzing ITS log data. Although my original goal in collecting classroom observation data was to investigate the impacts and predictors of teachers' helping behaviors in the classroom, replay analyses revealed that teachers' proximity seemed to have much more salient effects on student learning and behavior. A teacher's location in the classroom appeared to be related to whether or not particular students were on-task at any given moment, and the activity of students sitting next to one another often appeared to be temporally synchronized (similar to the "distraction ripples" observed by Raca & Dillenbourg (2013)). Furthermore, when the teacher was either distracted or outside of the classroom, many students appeared to stop working with the software entirely, and students' willingness to raise their hands (as well as their likelihood of receiving help from the teacher as a result) appeared to increase during time intervals in which the teacher was nearby.

Relationships between student-teacher interactions and student learning outcomes

After using replay analysis to gain a richer qualitative picture of classroom dynamics in a small set of class sessions, I conducted quantitative analyses on the synchronized logs generated by the SPACLE prototype in order to investigate the robustness of some of the patterns I had observed. Since I am ultimately interested in students' learning outcomes, I began by examining relationships between the frequencies of various student–teacher interactions (evaluated per-student) and students' pre-post learning gains.

As shown in Table 2-2, neither a student's frequency of on-task conversations with the teacher nor their frequency of requesting help (via hand-raising) were significantly correlated with their performance on the posttest, even when controlling for the student's pretest score. Interestingly, the frequency with which a teacher directly monitored a student was the only measured aspect of students' and teachers' interactions in the classroom that correlated significantly with posttest. The relationship with direct monitoring remains significant even when controlling for pretest.

Table 2-2. Zero-order and partial correlations (controlling for pretest) between student–teacher interactions and posttest scores. $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

	On-task conversation	**Off-task conversation**	**Direct monitoring**	**Hand raising**
Zero-order correlation	0.00	0.13	0.39***	-0.02
Partial correlation	-0.08	-0.14	0.20*	-0.08

In order to better understand the mechanisms by which this apparent link might arise, I adopted a causal model search approach, using directed acyclic graphs (DAGs) to represent the qualitative causal structure among measured variables. I used the PC algorithm in the Tetrad V program[4] to search for an equivalence class of graphs that are consistent with a set of conditional independence constraints (Spirtes, Glymour, & Scheines, 2000) I included background knowledge about the experimental design as a search constraint: namely, that the pretest precedes all process variables, which in turn are all prior to the posttest. The PC algorithm is asymptotically reliable, and its primary limitations lie in its assumptions that the underlying causal dependencies between variables can be modeled with linear functions, and that there are no unmeasured common causes among variables. To relax the second of these assumptions, I also used the FCI algorithm to learn an equivalence class of graphs, represented by partial ancestral graphs (PAGs). PAGs are representationally richer than DAGs, and may contain edges representing uncertainty over the nature of pairwise relationships between variables (Spirtes et al., 2000):

- **X → Y:** X causes Y in every member of the equivalence class represented by this PAG.
- **X ↔ Y:** X and Y share a latent common cause in every member of the equivalence class represented by this PAG.
- **X o→ Y:** Either X causes Y, X and Y share a common cause, or both.
- **X o—o Y:** X is a cause of Y or Y is a cause of X. Alternatively, X and Y may share a latent common cause (either in the absence of a direct causal link between the two variables, or in addition to one).

[4] Available at: *http://www.phil.cmu.edu/projects/tetrad/*

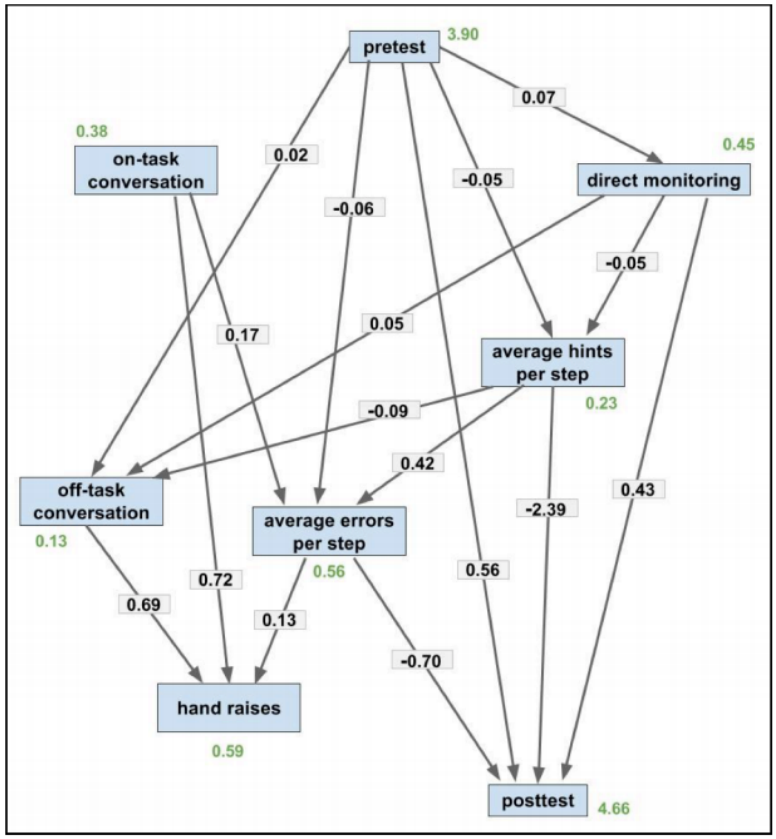

Figure 2-3. The model found by PC, with parameter estimates included. This model fits the data well: $\chi^2 = 11.31$, df = 12, p = .50.

Figure 2-3 shows the model found by PC, with path coefficient estimates included. The model fits the data well ($\chi^2 = 6.03$, df = 10, p = .81)[5], and contains a number of properties that are consistent with findings in prior literature on the effects of student help-seeking behaviors on learning gains with ITSs. For example, under this model, increased use of the ITS's hint functionality appears to inhibit learning, *in general* (Aleven et al., 2016). Also, compatible with previous findings that on-task conversations with peers and teachers during ITS use may be negatively related to student learning *in general*, the model found by PC suggests that on-task conversations with teachers may increase students' within-software error rates (cf. Baker et al., 2004). However, I did not replicate Baker et al.'s finding of a negative relationship between on-task conversations and student *learning gains*, instead finding no significant relationship (perhaps owing, in part, to differences in the quality and effects of *peer help* versus *teacher*

[5] Note that in path analysis, the null hypothesis is that the estimated model is the true model, and the p-value represents the probability that a difference between the estimated and the observed covariance matrices at least as large as the realized difference would have been observed under the null hypothesis. As such, a p-value above a specified threshold (conventionally alpha = .05) implies that the model cannot be rejected.

help). Note that the observation of a negative relationship between on-task conversations and student error rates, and the absence of an observed relationship with learning gains may be, at least in part, due to a selection effect. Students who have more on-task conversations with their teacher may be those who are having more difficulties in the software (for reasons that may not be captured by their performance on the pretest alone), and who are in turn likely to learn less (Aleven et al., 2016; Baker et al., 2004). In addition, it is possible that a finer-grained coding of the nature or content of these on-task conversations would have revealed particular circumstances under which such conversations produce a measurable increase or decrease in student learning, as measured by the posttest.

The observed positive relationship between the frequency of direct monitoring by the teacher and student posttest scores appears to have been mediated, in part, by students' hint-use behavior. One possibility this suggests – made more plausible by my observations during spatial replay analyses – is that students who are more aware that the teacher is monitoring them are less likely to engage in maladaptive learning behaviors such as abusing software-provided hints, and are therefore more likely to learn the material. It is also possible, however, that the apparent link between teachers' direct monitoring behaviors and student learning gains reflects a selection effect. For example, teachers may tend to more frequently monitor students who show signs of making progress in the software (or who the teacher believes are more likely to make progress). Interestingly, this model suggests that students with higher prior domain knowledge (as measured by pretest scores) may have been somewhat *more* likely to receive additional monitoring from the teacher. In a follow-up interview, one of the teachers who participated in this study claimed to have intentionally placed a group of students in a relatively isolated and inaccessible area of the classroom, as these students were *"a pain to deal with"* – hinting at mechanisms by which this apparent bias could have arisen.

Prior work from Stang and Roll (2014) found similar results at the university level. In their study of interactions between teaching assistants (TAs) and students in 'hands-on' laboratory sections of large introductory physics courses, Stang and Roll found that the frequency of TA–student interactions was a strong and positive predictor of student engagement (defined as on-task behavior), which was in turn an even stronger predictor of students' learning gains than their pretest scores. Compatible with my findings, the authors found that this relationship held for interactions that were initiated by TAs, but not for those initiated by students. In addition, very brief visits by the TA appeared to be just as effective as lengthy interactions. The authors posited that this might be due either to a "policing" effect (i.e., frequent interactions motivate students to not stray off-task), or a "ventilation" effect (i.e., TA-initiated visits open the door for productive conversations with students).

To gain a sense of the relative plausibility of these two explanations in my own dataset, I ran follow-up replay analyses with SPACLE, across two teachers and class sessions – visualizing the rate of student hint requests on each student's "computer screen" by displaying a pulse of color

each time a student asked for a hint. These replays suggested that students might have been less likely to request hints when the teacher was nearby. In addition, students who were frequently observed asking for multiple hints in rapid succession appeared to pause this behavior when the teacher was nearby or directly monitoring them – lending some support to Stang and Roll's "policing" hypothesis, but while also remaining compatible with their "ventilation" hypothesis.

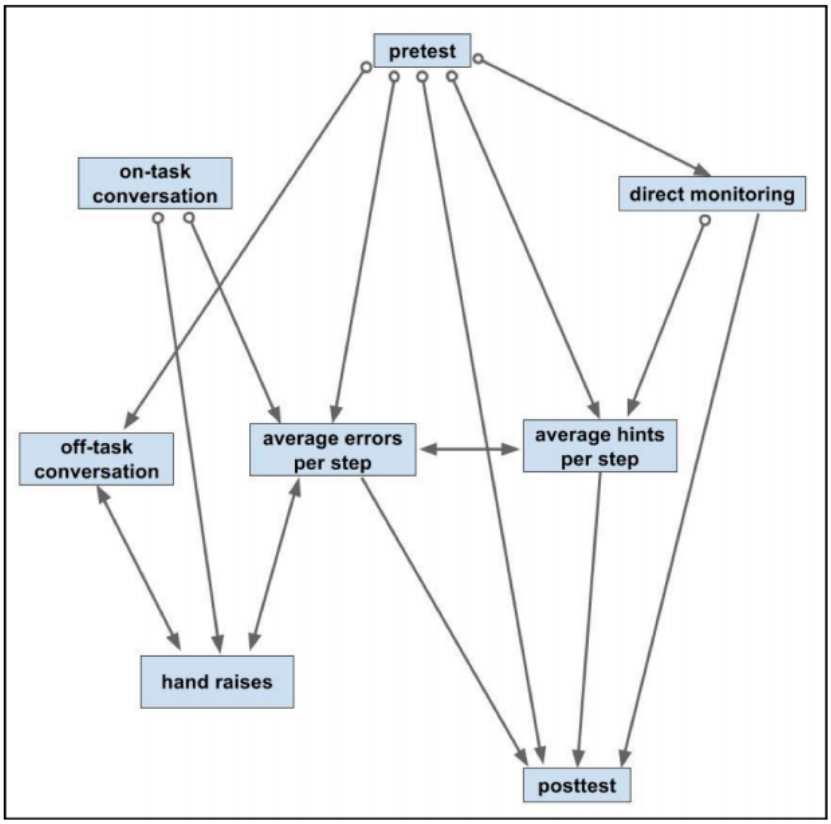

Figure 2-4. The PAG equivalence class found by FCI, which encodes the possibility of unmeasured common causes.

Given the potential for confounding factors, I used the FCI algorithm to learn a PAG causal model, relaxing the assumption of no unmeasured common causes (see Figure 2-4). The learned structure is largely the same, except that this model encodes the possibility that students' pretest scores may be related to direct monitoring, off-task conversation, hint use, and/or error rate by a *common unmeasured cause*, and that the same may be true for the relationships between direct monitoring and hint use, as well as on-task conversation and its children: hand raises and error rate. In addition, the learned structure suggests that individual students' frequency of hand-raising shares common unmeasured causes with their frequency of off-task conversation and their within-software error rates (which in turn may share a common cause with students' rate of hint-use) – perhaps indicating that these behaviors are symptoms of unmeasured cognitive, motivational, and affective states such as confusion and frustration (Baker, 2011).

However, the positive link between direct monitoring and student learning gains remains in every member of the equivalence class found by FCI.

2.5 Conclusions

Classroom studies that evaluate the effectiveness of educational technologies do not typically examine the effects of interactions occurring outside of the software, such as face-to-face teacher-student or peer interactions. Yet these out-of-software interactions may mediate these technologies' effectiveness (Miller et al., 2015). To facilitate the discovery of relationships between out-of-software interactions and student learning within educational software in personalized classrooms, I have introduced a replay visualization method, Spatial Classroom Log Exploration (SPACLE), which replays moment-by-moment analytics about student and teacher behaviors in their original spatial context (e.g., overlaid upon a classroom seating chart).

Through exploratory data analyses using these spatial replay visualizations, combined with causal data mining, I found suggestive evidence that students' mere awareness that they are being monitored by a teacher may contribute to greater student engagement and learning – perhaps in part by reducing "gaming the system" behaviors (Holstein et al., 2017a). Specifically, the observational findings presented in this chapter suggest that students who receive more frequent monitoring from teachers in ITS classrooms may learn more, and that this effect may be partially mediated by students' hint-use behavior within the software. This hints at the usefulness of a broader notion of "gaming the system" than has been used previously – taking into account student behaviors that extend outside of the software (Holstein et al., 2017a; 2018b). The use of SPACLE replays on a small subset of our data throughout the analysis process enabled us to evaluate the relative plausibility of various hypotheses that were compatible with these causal models.

These findings partially replicate Stang & Roll (2014) – lending support to the authors' prediction that their observed relationship between teacher visits and student engagement would generalize beyond their study's specific context (inquiry-based laboratory sessions in an introductory, university-level physics course). In addition, these findings may help interpret Stang & Roll's observation that a teachers' frequency of interaction with a student predicts student engagement, independent of the length of these interactions. Our findings suggest that teachers' interactions may not need to have a verbal component in order to be effective – that is, K-12 students' mere awareness of being monitored may have a positive impact on their learning in personalized, self-paced classroom settings – at least over short timescales. Results from a subsequent in-vivo classroom experiment presented in *Chapter 7* appear to lend some additional support to this observation, although without providing conclusive evidence.

In addition, although teachers self-reported (in post-interviews following ITS class sessions) that they tried to help students who they believed were struggling the most with the material, in

practice these teachers appeared to exhibit biases in the opposite direction – for example, by spending more of their time with students who had higher prior domain knowledge, as measured by a pretest. Preliminary findings from subsequent classroom data mining investigations (see *Chapter 6*) suggest one possible mechanism by which such biases may have arisen. In classrooms not using a real-time teacher analytics tool, students who exhibit patterns of "help avoidance" (Aleven, Roll, McLaren, & Koedinger, 2016) within educational software may tend to receive less teacher attention (Holstein et al., 2018a). The teachers we worked with anticipated this pattern, often expressing a belief that the students who need help the most tend to be among the least likely to request it (e.g., by raising their hands) (see *Chapters 1*, *4*, *5*, and *9*).

Teachers who were shown spatial replays of their own class sessions (within a week following those sessions) often expressed surprise at how much was happening in the classroom outside of their awareness (e.g., students who were idle for extended periods of time, making many errors in the software, or gaming-the-system), or reflected that they were not distributing their time across students in the way they *aspired to do so* (Holstein et al., 2017a; 2017b). These early findings suggested that, in addition to being useful for exploratory data analyses, spatial classroom replays might provide an effective way to promote teacher reflection on their own behavior, following a class session (cf. Gerritsen, Zimmerman, & Ogan, 2018; Martinez-Maldonado, 2019; Prieto, Magnuson, Dillenbourg, & Saar, 2017), and relatedly, a means of quickly prototyping and co-designing new forms of orchestration support for AI-supported classrooms (Holstein et al., 2017a; 2019a).

Before concluding, several limitations of these early investigations should be mentioned. The causal models presented in this chapter should not be viewed as the "true" models. First, although the data under consideration come from an experimental study, the subset of the data analyzed in this chapter are from a portion of the study in which researchers did not intervene on any of the measured variables between the pre- and post-test (except insofar as running an in-vivo classroom study can be considered an intervention in itself). As such, the data presented in this chapter should be considered observational. Future experimental investigation is required to evaluate the causal nature of each link identified in these causal models (see *Chapter 7*). Second, the causal search algorithms used in the present work assume that the underlying relationships between the modeled variables are truly linear, which may not hold in practice. Nonetheless, this model assumption is not unreasonable, as the relationships within this dataset do appear approximately linear. Third, although the sample of students used to construct these causal models is relatively large compared to many ITS studies, the reliability of model search would be improved with access to larger samples, and it is generally impossible to compute confidence bounds when dealing with finite samples (Robins, Scheines, Spirtes, & Wasserman, 2003; Spirtes, Glymour, & Scheines, 2000).

In addition, it is worth noting that the present analyses examined student–teacher interactions over a relatively brief period (60 minutes). A promising direction for future research is to

observe ITS classrooms over longer time periods, in order to study how teacher practices (and their effects on student learning) may evolve over time (see *Chapter 10* and *Conclusions, Contributions, and Future Directions* for a discussion). Finally, in the current study, a single human observer manually collected classroom observations of teacher-student interactions. This necessitated the use of a very coarse-grained coding scheme, and also limited the number of classrooms our research team could feasibly observe. An important direction for future work is to automate more components of the classroom data collection process (cf. Prieto, Sharma, Dillenbourg, & Jesús, 2016). A semi-automated approach, using a combination of low-cost sensors and manual observations, may enable more detailed coding schemes by freeing human observers to focus on recording higher-level features (e.g. semantic features of on-task conversations in the classroom). *Chapters 4* through *7* present some progress in this direction: an automated, "inside out" approach to tracking teachers' physical movement in the classroom that does not require instrumentation of the classroom space (see item 5 under *Summary of Contributions – "First investigations of relationships between teacher movement/monitoring and student behavior and learning in AI-supported classrooms"*).

Taken together, the investigations presented in this chapter provided preliminary evidence that – in addition to serving teacher needs and desires – there may be opportunities for real-time teacher analytics to benefit student learning in ITS classrooms, by redirecting teachers' attention towards students who may need their help the most, and providing additional signals of help-need beyond students' own help-seeking behavior (Holstein et al., 2017a; 2018a; Holstein, Hong, et al., 2018). Through both replay analyses and causal modeling, I observed rich relationships between students' out-of-software interactions in and their within-software learning and behavior. Some of the most salient observed effects appeared not to necessarily involve verbal interactions between students and their teachers, but rather appeared to be due to spatial factors such as the teacher's position in the room, relative to a student. I view these observations as suggestive that the influence of such out-of-software factors on student learning with ITSs and similar educational technologies has perhaps been under studied previously. The observed relationship between teachers' monitoring behaviors and students' learning gains suggest that one potential mechanism by which real-time teacher analytics tools might be effective in promoting student learning is by simply making students aware that they are being monitored. These hypothesized causal paths from teacher behavior to students' learning outcomes are further investigated in *Chapters 6* and *7*, through iterative classroom piloting and in-vivo classroom experimentation.

Although student behavior in AI-supported, personalized classrooms has been extensively studied using log data from educational software, the impacts of students' out-of-software interactions are rarely studied (Miller, et al., 2015). While prior work (e.g., Stang & Roll, 2014) has investigated associations between teachers' physical movement and monitoring behaviors in co-located classrooms (e.g., looking over students' shoulders as they work) and students'

behaviors and learning outcomes, using observational data, the present work is the first to investigate such associations in the context of classrooms using ITSs (see item 5 under *Summary of Contributions – "First investigations of relationships between teacher movement/monitoring and student behavior and learning in AI-supported classrooms"*).

Chapter 3

Opening up an Intelligent Tutoring System Development Environment for Extensible Student Modeling and Learning Analytics

This chapter is based in part on the following publications:

- Holstein, K., Yu, Z. Popescu, O., Sewall, J., McLaren, B. M., & Aleven, V. (2018). Opening up an intelligent tutoring system development environment for extensible student modeling. In *Proceedings of the 19th International Conference on Artificial Intelligence in Education (AIED 2018).*

3.1 Background and Motivation

To enable the next phase of my thesis work – the development and rapid prototyping of actual real-time teacher analytics tools for use with ITSs (discussed in *Part Two* of this thesis) – I needed authoring tool functionality that did not yet exist.

Taking this as an opportunity to support the broader research community, my collaborators and I substantially extended *CTAT/TutorShop* (Aleven et al., 2016), a widely-used technical architecture for ITS authoring and deployment. The extended architecture, *CTAT/TutorShop Analytics (CT+A)*[6] (Holstein, Yu, et al., 2018), is designed to support *extensible student modeling*: the authoring, sharing, and re-use of a broad and open range of student modeling techniques, for use in running ITSs (i.e., to drive adaptive tutoring behavior) and/or external learning analytics tools (see Holstein, Hong, et al., 2018; Holstein, Yu, et al., 2018; Paquette, Baker, & Moskal, 2018).

Over the last few decades, authoring tools have made the development of intelligent tutoring systems (ITSs) substantially more cost effective (Aleven et al., 2016; Blessing, Aleven, Gilbert, Heffernan, Matsuda, & Mitrovic, 2015; MacLellan, Koedinger, & Matsuda, 2014; Razzaq et al., 2009). Yet these tools are not always geared towards easily and flexibly accommodating advances in student modeling, which may limit the degree to which they drive innovation in ITS research and the degree to which advances in student modeling spread across ITSs. Student models have long been (and remain) a key element of ITSs. They track many

[6] CTAT/TutorShop Analytics is available in the following open repository:
https://github.com/d19fe8/CTAT-detector-plugins/wiki

pedagogically-relevant features of student learning and behavior, including the moment-by-moment development of student knowledge (e.g., Corbett & Anderson, 1995; Desmarais & Baker, 2012; Khajah, Lindsey, & Mozer, 2015), metacognitive skills (e.g., Aleven, Roll, McLaren, and Koedinger, 2016), affect (e.g., D'Mello, Lehman, & Graesser, 2011; Fancsali, 2014; Liu et al., 2013), and motivation (e.g., Baker et al., 2006). They are a foundation for adaptive tutoring behaviors in ITSs (Desmarais & Baker, 2012), which in turn can lead to more effective instruction (Aleven, Roll, et al., 2016; Baker et al., 2006; D'Mello et al., 2011; Holstein et al., 2018b; Long & Aleven, 2013). Student models, and learning analytics more broadly, are also increasingly being used in tools such as learning analytics dashboards, open learner models, and classroom orchestration tools, where they can augment the perceptions of both teachers (e.g., Bull & Kay, 2016; Feng & Heffernan, 2007; Holstein et al., 2016; 2017b; 2018b; Xhakaj et al., 2017; Yacef, 2002) and learners (e.g., Bull & Kay, 2016; Clow, 2012; Long & Aleven, 2013; 2017; Ritter et al., 2007).

However, various factors work against novel student modeling methods spreading widely in ITSs. These methods (e.g., Desmarais & Baker, 2012; Khajah et al., 2015; Paquette, Baker, & Moskal, 2018; Yudelson, Koedinger, & Gordon, 2013) are often developed and tested on historical log data from educational software (i.e., "offline"). They are not commonly implemented or evaluated in real-world educational technologies; see, for example, AFM (Cen, Koedinger, & Junker, 2006), PFA (Pavlik, Cen, & Koedinger, 2009), and various innovations on Bayesian Knowledge Tracing (BKT) such as Khajah et al. (2015) and Yudelson et al. (2013); but see Corbett & Anderson (1995). Even when an advance in student modeling has been demonstrated in a live tutoring system, it often stays confined to that system, without being taken up in other systems (e.g., Aleven, Roll, et al., 2016; Baker et al., 2006; D'Mello et al., 2011; Grawemeyer, Holmes, Gutiérrez-Santos, Hansen, Loibl, & Mavrikis, 2015). ITS authoring tools, and the ITS architectures with which they are integrated, could help address these challenges if they provided support for easy integration of a wide and open range of student modeling methods and analytics. Given that for many ITS authoring tools, many classroom-proven tutors exist, such authoring tool functionality could facilitate testing the generality of new student modeling methods across a range of tutors. Further, easy integration could facilitate further experimentation with new student modeling methods, beyond the initial offline testing, regarding how best to use these methods to enhance an ITS's functionality (e.g., with new adaptive tutoring behaviors or external learning analytics tools). Eventually, researchers may conduct more close-the-loop studies, in which the effects of new student modeling methods and analytics are rigorously tested in "live" tutoring systems (e.g., Aleven, Roll, et al., 2016; Baker et al., 2006; Clow, 2012).

Results from such studies could accelerate a cumulative science of student modeling, as well as extend student modeling advances into working ITSs and educational practice. However, authoring tools for ITSs rarely support extensible student modeling. For example, prior to the

work reported in this chapter, CTAT/Tutorshop, an authoring environment for cognitive tutors and example-tracing tutors that has been used to build many dozens of ITSs (Aleven, McLaren, et al., 2016), supported only student models comprising a set of BKT mastery probabilities for knowledge components (KCs) within the authored tutors. An author could not add other types of variables to the student model (e.g., to track the student's affective or motivational state, or metacognition) or easily experiment with different methods for updating or using the student model. Similarly, ASSISTments Builder (Razzaq et al., 2009) and ASPIRE (Mitrovic, 2009), other major ITS authoring tools, do not support easy extension of their student models with new types of variables. By contrast, GIFT (Sottilare, Baker, Graesser, & Lester, 2017) does support an extensible student model based on multiple data sources (e.g., sensor data) with different time scales and granularity. Yet GIFT has been designed with a different focus than CTAT, and thus has other limitations (Fancsali, Ritter, Stamper, & Nixon, 2013). For example, unlike CTAT, GIFT does not support non-programmer authoring of tutors with their own tutor interface and an extended step loop. We see these related, somewhat divergent, efforts as synergistic and a useful point of reference.

To address this challenge, we have extended CTAT/Tutorshop so that authors can easily plug in an open-ended range of student modeling techniques. The extensions also support the authoring of an open-ended range of adaptive tutoring behaviors and facilitate the development of an open-ended range of student-facing and teacher-facing support tools, including real-time tools for classroom awareness and orchestration (Clow, 2012; Holstein, Hong, et al., 2018; Rodriguez Triana et al., 2017). We refer to the new architecture as CTAT/TutorShop Analytics (CT+A). In creating this extended architecture, we aim to lower the barriers to the sharing, re-use, and re-mixing of advanced student modeling methods across researchers and research groups, with the goal of accelerating progress within a cumulative science of student modeling (cf. Desmarais & Baker, 2012; Paquette et al., 2018; Sottilare et al., 2017) (see item 7 under *Summary of Contributions – "CTAT/TutorShop Analytics, an extended architecture for ITS development that supports 'extensible student modeling' "*).

3.2 The CTAT/Tutorshop Analytics (CT+A) Architecture

Overview

CTAT is a widely used ITS authoring tool that supports both a non-programmer approach (example-tracing tutors) and an AI-programming approach (Cognitive Tutors, a form of model-tracing tutors) to tutor authoring. TutorShop is a learning management system (LMS) built for classroom use of CTAT tutors. Use of CTAT has been estimated to make ITS development 4-8 times as cost effective, compared to historic estimates of development time (Aleven, McLaren, et al., 2016). As evidence that CTAT and Tutorshop are robust and mature, CTAT has been used by more than 750 authors. Dozens of tutors built with CTAT have been used

in real educational settings (Aleven, McLaren, et al., 2016). As of 2015, CTAT-built tutors had been used by 44,000 students, with roughly 48,000,000 student/tutor transactions, for a total of 62,000 hours of student work. Since then, there has been substantial additional use.

I first describe key elements of the CT+A architecture that existed prior to adding new support for extensible student modeling (shown in Figure 3-1, top). At a functional level, each tutor created in this architecture comprises a "step loop" nested within a "task loop" (VanLehn, 2006; 2016). The step loop supports within-problem tutoring, and the task loop supports problem selection. The step loop has two key components, namely, a tutor interface and a tutor engine, both running on the client (i.e., the student's machine). The interface is where student–tutor interactions occur; it is custom-designed for each problem type. The tutor engine interprets student actions and decides what feedback or hints to give, employing either the model-tracing or example-tracing algorithm, depending on the tutor type. The tutor's task loop is implemented in TutorShop and runs on the server. CT+A offers various problem selection algorithms that can be used within a tutor, including individualized mastery learning (Corbett & Anderson, 1995). This method relies on a student model that, as mentioned, contains estimates of the probability that the student has mastered each of a set of KCs targeted in the current tutor unit, computed (by the tutor engine, as part of the step loop) according to a standard BKT model (Corbett & Anderson, 1995). TutorShop takes care of permanent storage of the student model. It also provides learning management functionality for teachers (e.g., managing student accounts and assignments), as well as content management (e.g., it stores tutor curriculum content).

This architecture has been used to build many tutors, but it cannot easily accommodate new student modeling methods. To address this limitation, my collaborators and I added the following key extensions (shown in Figure 3-1, bottom):

1. An extensible student model. An author can now add new variables to the student model.
2. An API and template for automated plug-in detectors for any new student modeling variables (i.e., computational processes – oftentimes machine-learned – that track psychological and behavioral states of learners based on the transaction stream with the ITS). For the time being, the architecture focuses on sensor-free detection of student modeling variables. We have started to create a library of compatible detectors, so as to facilitate sharing, re-use, and remixing of plug-in detectors among authors;
3. Multiple mechanisms by which authors can craft tutor behavior that adapts to student model extensions, in the tutor's step loop and task loop;
4. A forwarding mechanism within TutorShop that allows authors to pass student models to web-connected learning analytics displays on a broad range of platforms (from browser-based dashboards to wearable devices);
5. The beginnings of a library of "dashlets," to facilitate building learning analytics tools. Dashlets are re-usable interface components that can be associated with sets of analytics and configured to visualize these analytics.

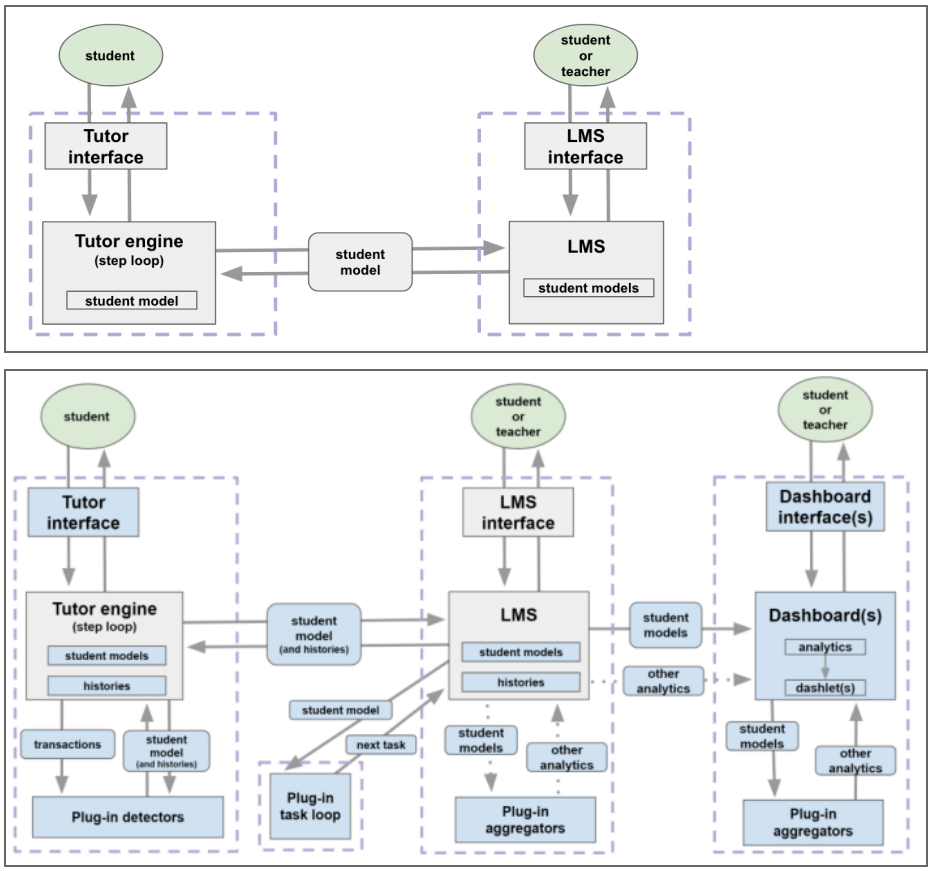

Figure 3-1. Comparison before and after architectural extensions. Top: Overview of the *CTAT/TutorShop* architecture prior to extensions. Bottom: Overview of the modular *CT+A* architecture, illustrating the flow of information between architectural components, with the top level (ovals) representing users. Items in blue represent configurable components. Rounded boxes indicate information being passed between architectural components. Dotted arrows represent pathways that are not presently implemented.

The Extensible Student Model

Whereas previously the student model of a tutor built in CTAT/Tutorshop comprised only a set of KC probabilities, the student model is now extensible, with authors having full control over the set of variables it contains. An author can add any number of variables to the student model that capture student behaviors and inferred psychological states (e.g., knowledge, metacognitive, affective, or motivational states; see Desmarais & Baker (2012) for a review). The KC probabilities remain as variables in the student model if the author so wishes. With the exception of these KC probabilities, TutorShop is oblivious to the semantics of the analytics in the student model (i.e., it does not have any built-in functionality that responds to the student model analytics; all such functionality must be provided by the author). A key advantage of this "semantic ignorance" is flexibility and control on the part of authors defining and using these

analytics. Transparent to the author, the CT+A architecture maintains, in real time, two up-to-the-second copies of the student model, one within the tutor engine, one within TutorShop. Within the tutor engine, the student model can support adaptive tutoring behaviors. Within TutorShop, it can support external learning analytics tools an author may wish to create or hook in (e.g., a real-time dashboard or orchestration tool). The copy of the student model stored by TutorShop is kept in between problems and student sessions and is sent back to the tutor engine at the beginning of each problem/session, again transparent to the author.

Plug-in Detectors

To extend the student model, an author needs to provide automated detectors for all new student model variables, that is, code that computes these variables. Tutor authors can write plug-in detectors in Javascript, working from either previously-created detectors or from a generic template, available in a central, open source code repository. The template defines a small number of code modules that each detector should have, namely, student model variable computations, internal feature computations, and trigger conditions for each.

To support a "remix" approach to student modeling, I have started a library of detectors that conform to this template. The library is freely available[7], and it is my hope it will continue to grow through community authoring. Many of the detectors currently available have been used in running ITSs and dashboards, including: multiple variants of the Help Model (Aleven, Roll, et al., 2016), BKT (Corbett & Anderson, 1995), various moving average detectors of student knowledge growth (Pelánek & Řihák, 2017), and detectors of unproductive persistence (Beck & Gong, 2013; Kai et al., 2018). Paquette et al. have also recently developed and shared a detector of "gaming the system" behavior in ITSs (Baker et al., 2006) that generalizes well across a diverse range of systems (Paquette et al., 2018).

Detectors in CT+A are plug-in agents that rely on three sources of input. First, they listen to the transaction stream coming from the tutor engine; each transaction describes a student action, such as an attempt at solving a step or a hint request, as well as the tutor's response, such as whether the student action was correct and what KCs were involved. Each detector also listens for updates to the extensible student model (i.e., updates made by other detectors), and has access to all student model variables, in addition to any intermediate variables that the detector itself maintains (see below). Based on these inputs, each detector responds with newly computed values for its targeted student model variables. As a result, both copies of the student model (the one within the tutor engine and the one within TutorShop) are updated, transparent to the author. Student model updates are sent to TutorShop in a fine-grained, transaction-based message format we have adopted, a subset of LearnSphere's Tutor Message format (Fancsali et al., 2013; Ritter & Koedinger, 1995).

[7] *https://github.com/d19fe8/CTAT-detector-plugins/wiki/*

Each detector can maintain an internal state in the form of a set of intermediate variables specified to conform to the detector template. Intermediate variables are not considered to be part of the student model and are therefore not accessible to other architectural components such as other detectors or aggregators. They are, however, sent to TutorShop, so that TutorShop can save a (compact) "history" for each detector. These detector histories are sent back to the tutor engine at the beginning of each problem, so that the previous state of each associated detector can be restored.

Although CTAT detectors typically run in live tutoring systems, they can also be used, without modifications, for offline data analyses (e.g., Holstein et al., 2018a; 2018b; Paquette et al., 2018). LearnSphere (Stamper et al., 2016), a large online data repository with many analysis tools, provides a workflow component for CTAT detectors, in the Tigris visual workflow tool, that enables running detectors against historical log data (from the same or other CTAT tutors).

Extended Support for Authoring Adaptive Tutor Behaviors

To enable the authoring of a wide and open range of adaptive tutor behaviors, my collaborators and I added two mechanisms to CTAT by which an author can make a tutor's behavior in the step loop (i.e., the within-problem tutoring support it offers (Aleven, Mclaughlin, Glenn, & Koedinger, 2017; VanLehn, 2016) contingent on the extensible student model. We also made provisions for plugging in new task selection algorithms in the tutor's task loop.

As a first mechanism for creating step-loop tutor behaviors that are responsive to the extensible student model, authors of example-tracing tutors can use Excel-like formulas that reference student model variables. The use of formulas, attached to the tutor's behavior graph, has long been part of CTAT (Aleven, McLaren, et al., 2016); what's new is that formulas can reference variables in the extensible student model (see Figure 3-2 below). Formulas can affect many aspects of tutor behavior, including how the tutor interprets a student's problem-solving behavior against a behavior graph, the content of feedback and hints, and tutor-performed actions. Using these building blocks, an author can craft a wide and open-ended range of adaptive tutor behaviors, for example, presenting abstract hints to advanced students, presenting empathic hints to frustrated students, presenting unmastered steps as worked-out steps to be explained by the student, and having the tutor perform highly mastered steps for the student to reduce "busy work." Authors of rule-based tutors can also craft rules that support adaptive behaviors, taking of advantage of the extensible student model's availability in working memory.

A second mechanism addresses a limitation of the first, namely, that it cannot be used to craft adaptive tutor behaviors that respond to the very last (i.e., the most recent) student action – it lags by one student action. Sometimes, tutor behaviors are needed that are contingent upon updates of the extensible student model triggered by the very last student action. Our second mechanism lets author craft such tutor behavior, although to do so, the author must write Javascript code.

Specifically, all tutors have a dedicated plug-in agent called the "Tutor's Ear", that continuously listens for updates to the student model. The Tutor's Ear has unique access to the tutor engine, meaning that it can directly trigger tutor responses. Authors can customize this detector by specifying (in Javascript code) conditions involving one or more student model variables under which a particular tutor response should be triggered. Authors can then specify desired response actions (e.g., "ShowMessage ('Try explaining to yourself what needs to be done on this step')"), via a simple API. Ideally, CTAT would have a single mechanism for step-loop adaptivity based on student model variables, but a substantial re-architecting would be necessary to merge the two mechanisms.

In addition to supporting the authoring of adaptive behaviors in the tutor's step loop, we support the plugging-in of adaptive task selection methods (i.e., plug-in task loops), by making the student model available to external task selection processes.

Support for Using Learning Analytics in External Support Tools

Finally, authors may use Tutorshop to forward student models to web-connected learning analytics displays on a range of platforms, from browser-based dashboards to wearable devices (Holstein, Hong et al., 2018). While detectors in CT+A operate client-side, within individual students' tutors, and thus can only compute analytics for individual students, it is often useful for learning analytics applications (e.g., teacher dashboards and classroom orchestration tools) to compute analytics at higher units of analysis, such as groups of students or whole classes. For example, in classrooms in which students work with CTAT tutors collaboratively (cf. Olsen et al., 2014), information about the relative performance and contributions of the students in a group might be useful to display to teachers. To address this need, the extended architecture provides an "aggregator" API to enable authors to compute custom group- or class-level analytics from student model variables across multiple students.

Authors of learning analytics tools can write custom "aggregators" in JavaScript to calculate new values from detector analytics across specified sets of students. Aggregator calculations can be triggered by incoming student model updates. We created the Aggregator House (AggHouse), a JavaScript/Node.js[8] library that can invoke aggregators either on the Tutorshop server, or directly on a dashboard client. Results from aggregators can, in turn, be used to update real-time dashboard displays.

To facilitate building analytics tools that can be used in conjunction with CTAT-built tutors (e.g., dashboards and orchestration tools), we developed an API called the CT+A Live Dashboard, which includes the beginnings of a library of "dashlets," interface components for analytics tools. Authors may use the built-in dashlet components or create new dashlets (using Javascript). In addition to supporting the building and deployment of web browser-based dashboards,

[8] *https://nodejs.org/en/*

Tutorshop can also forward analytics to tools running on external hosts, via a real-time event stream in JSON format, to support analytics tools across a range of hardware interfaces.

Lessons Learned: Guiding Principles for Extensible Student Modeling

In designing CT+A so it can support an open range of student modeling applications, with provisions for real-time support tools, a set of guiding architectural principles has emerged. These principles capture the key architectural elements added to CT+A.

Maximize tutor-side computations.
We have structured detectors to promote incremental (e.g., per transaction) computation of analytics. This supports offloading of student model computations to the tutor clients, rather than the LMS, since incremental computations spread processing load over time.

Keep data streams "lean".
In designing key data streams (i.e., the transaction stream into the detectors, and the student model update stream from tutor to LMS), we settled on a small subset of the information CTAT tutors currently send to LearnSphere (Stamper et al., 2016). We originally attempted to anticipate many possible author needs and build these into the transaction messages (Ritter & Koedinger, 1995) that serve as primitive inputs to plug-in detectors, but decided against this approach. Keeping this set small can reduce unnecessary message traffic and redundancy by acknowledging the wide range of analytics authors may wish to compute and enabling them to compute only those needed.

Maintain the student model both locally and centrally.
Prior to these architectural extensions, an up-to-the-second copy of the student model was maintained on the tutor side, but the LMS-side copy was updated only as needed to preserve the student model in-between problems. We have found it valuable to instead maintain both a local (tutor-side) and central (LMS-side) up-to-the-second copy of the student model, with each copy supporting different use cases, namely, tutor adaptivity versus analytics tools; the latter typically require both class-level analytics and real-time updating, which is why central copies of the student models are useful.

Support easy re-mixing of existing components.
In addition to supporting plug-and-play of architectural components, we have found it valuable to make individual components easily-customizable. For example, each detector contains a module that exposes configurable parameters. This feature is intended to facilitate the creation of variants of student modeling techniques, including those created and shared by others, to support authors not only in comparing against each other's' models, but also in building upon and contributing to each other's modeling work (cf. Kery & Myers, 2018; Sottilare et al., 2017; Stamper et al., 2016).

3.3 Case Studies

In this section, we present case studies of prototype systems that use the CT+A architecture to enhance tutoring systems' adaptive capabilities and/or to support teachers.

A Prototype Tutor that Provides Metacognitive Scaffolding

The experience of some of the participants during our institution's yearly LearnLab summer school[9] illustrates how a detector library can be helpful in quickly prototyping adaptive tutor behaviors. During this summer school, designers, teachers, and researchers build their own systems using CT+A. In the summer of 2017, participants were able to author detector-enhanced ITSs, by re-using pre-existing detectors available in the detector library. They embedded pre-existing detectors into their tutors and authored custom adaptive tutor behavior based on detectors' outputs.

A team of two students, Dennis Bouvier and Ray Martinez, used the CT+A architecture to implement an ITS prototype that provided metacognitive feedback in addition to feedback at the domain level, which is standard in CTAT tutors. This tutor, the Binary Search Tutor, was intended to help undergraduate Computer Science students learn binary search algorithms. It allows students to practice applying a binary search algorithm to an array of numbers. The Binary Search Tutor uses a plug-in implementation of the Help Model, which can identify patterns in student-tutor interactions that indicate abuse (e.g., rapidly clicking through hints without reading) or avoidance (e.g., not using hints in situations where they are likely to be needed) of the tutoring software's built-in hints (Aleven, Roll, et al., 2016). Using custom response actions authored in the Tutor's Ear, the Binary Search Tutor responds to both types of student behavior. In the case of hint avoidance, the tutor prompts the student to ask for a hint. In the case of hint abuse, the tutor encourages the student to try attempting more steps without hints.

A Prototype Fraction Addition Tutor with Hybrid Adaptivity

Using CT+A, we have also created a tutor prototype that implements a form of "hybrid adaptivity" (Aleven et al., 2017), meaning that it adapts to combinations of student states. This tutor, an example-tracing tutor for 4th and 5th grade fraction addition problems, adjusts the level of scaffolding provided based jointly on the values of cognitive variables (skill mastery) and metacognitive variables (hint use, unproductive persistence).

[9] *https://learnlab.org/index.php/simon-initiative-summer-school/*

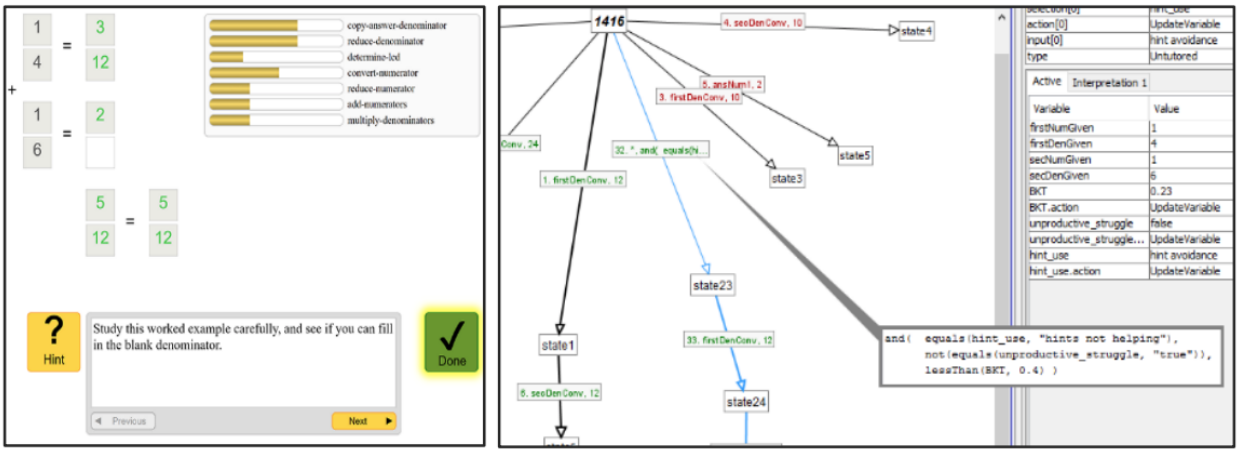

Figure 3-2. Left: The Fraction Addition Tutor uses multiple plug-in detectors to decide whether to provide more scaffolding. Right: Authoring the Fraction Addition Tutor in CTAT.

For example: if a student is detected as having low knowledge on KCs involved in the current step (by a plug-in of BKT (Corbett & Anderson, 1995)) and the student is detected as "using all available hints yet remaining stuck" (by the Help Model; Aleven, Roll, et al., 2016) but the student is not currently detected as necessarily "unproductively persisting" (by a detector of wheel-spinning; Beck & Gong, 2013; Kai et al., 2018), then the Fraction Addition Tutor will dynamically convert the student's current problem into a completion problem, by filling out all steps except one, and prompt the student to study the worked-out steps and fill in the remaining step (Figure 3-2, left). This capability was authored using a formula (expressed in CTAT's formula language) that references student model variables (i.e., the first of the two mechanisms described above for authoring adaptive tutor behavior). This formula was attached to a new path in the behavior graph (the main representation of domain knowledge in an exampletracing tutor), added by the author (Figure 3-2, right). The path specified the tutor-performed actions needed to fill in the worked-out steps.

Teacher Smart Glasses that Support Orchestration of Personalized Classrooms

The CT+A architecture has been used to develop *Lumilo*, a mixed reality smart glasses application, co-designed with K-12 math teachers, and developed for the Microsoft HoloLens 1 (*Lumilo* is presented in *Part Two* of this dissertation)[10]. *Lumilo* is designed to aid teachers in orchestrating personalized class sessions, in which students work with ITSs at their own pace. When a teacher puts these glasses on, they can see visual indicators floating directly over students' heads, based on changes in a student's extensible student model. The teacher can also view more detailed student-level analytics, as well as class-level summaries (see *Chapter 4*;

[10] *https://www.microsoft.com/en-us/hololens*

Holstein, Hong, et al., 2018; Holstein et al., 2018a; 2018b; 2019a). All student model updates are computed within students' tutor clients (using several plug-in detectors) and forwarded to TutorShop, which forwards them to *Lumilo*. Although *Lumilo* is not browser-based (and was thus authored outside of Live Dashboard, described above), TutorShop provides hooks for *Lumilo* to connect to each classrooms' analytics streams. *Lumilo*'s dashlets are then updated by aggregators on the *Lumilo* client.

A Prototype Dashboard that Supports Data-informed Lesson Planning

The CT+A architecture was used to develop *Luna,* a front-end prototype of a browser-based dashboard for K-12 teachers. Unlike *Lumilo*, which is designed to support real-time monitoring, *Luna* is intended to support teachers in lesson planning, using analytics generated by *Lynnette,* an ITS for equation solving (Holstein et al., 2016; Xhakaj et al., 2017).

Figure 3-3. An early prototype of the *Luna* dashboard for lesson-planning (developed using Tableau[11]) showing the class-level view (Holstein et al., 2016; Xhakaj et al., 2017).

Luna allows teachers to review students' knowledge and amount of practice on each of a number of fine-grained skills and error categories, either at the level of a class summary, or at the individual student level. In addition, teachers can use *Luna* to review individual students'

[11] *https://www.tableau.com/*

progress through the software, relative to the time they have spent working (Figure 3-3). As with *Lumilo*, the primitive level of data upon which *Luna* relies are student model updates, computed by plug-in detectors which are distributed across students' client machines.

A Fractions Tutor with a Custom Adaptive Task Selection Policy

Finally, the CT+A architecture was used to develop an adaptive fractions tutor (Doroudi, Aleven, & Brunskill; 2017; Doroudi, Holstein, Aleven, & Brunskill, 2015; 2016) which can use a variety of custom instructional policies to drive adaptive task selection (e.g., adaptive policies learned via reinforcement learning). The Fractions Adaptive tutor makes its student model available to external, custom task selection processes (Python web applications) via the TutorShop LMS. TutorShop, in turn, selects a next task for each student based on the output of this plug-in task loop.

3.4 Conclusions

If advances in student modeling made by the AI in Education (AIED), Educational Data Mining (EDM), Learning Analytics (LAK), and User Modeling (UM) communities are to have a measurable impact on the design and effectiveness of real-world systems, and contribute to a cumulative science of student modeling, it is critical to develop authoring tools that can support these goals. Toward this end, this chapter has introduced CT+A, an open architecture to support extensible student modeling (see item 7 under *Summary of Contributions – "CTAT/TutorShop Analytics, an extended architecture for ITS development that supports 'extensible student modeling' "*). This architecture supports the plugging in, sharing, re-mixing, and use of advanced student modeling techniques in ITSs and associated teacher- and student-facing analytics tools. The work is unique in that it supports extensible student models in the context of non-programmer ITS authoring tools that support building tutors with a dedicated problem-solving interface and elaborate step loop. In addition to the architecture itself, this chapter presents a small set of "lessons learned," in the form of principles summarizing the main architectural elements, which may inform future projects focused on extensible student modeling.

The case studies presented in this chapter illustrate some of the range and flexibility of CT+A and demonstrate progress towards four key goals for an analytics-integrated ITS architecture:

- Authors can add new variables to the student model by embedding detectors in running tutoring systems. This chapter has presented an API and template for creating these plug-in detectors, requiring only that authors are familiar with basic JavaScript.
- Existing detectors can be re-used and/or re-mixed.
- Authoring new adaptive tutoring behavior is feasible without programming.

- The CT+A architecture can support the development of a variety of external analytics tools, including both real-time and lesson-planning dashboards, and both web-based and wearable tools.

Limitations of the work are, at least for the time being, that CT+A focuses on transaction-based (in other words, sensor-free) student modeling (Desmarais & Baker, 2012). Although transaction-based student modeling can be a practical and widely useful approach (e.g., Baker et al., 2006; Desmarais & Baker, 2012; Fancsali et al., 2013; Stamper et al., 2016), questions surrounding how a student model can be updated with multiple data streams of different granularity (transactions and sensor output) are left for future work. As mentioned, such issues are being explored in the GIFT architecture (Sottilare et al., 2017). An additional limitation of the current architecture is that, in authoring tutoring behaviors responsive to the extensible student model, immediate tutor responses involve a different mechanism than tutor responses in subsequent tutor cycles. A more flexible and general solution might be give detectors and the tutor engine equal status, with a coordinating agent that has the final word regarding the tutor response (Ritter & Koedinger, 1995). Finally, adding *student model extensions* requires some programming (namely, to create detectors in Javascript) and thus falls outside CTAT's non-programmer paradigm. The amount of programming required can be greatly reduced, however, by re-using existing detectors, shared among authors in the CT+A detector library. In the future, new practices developed and tested within architecture might inform the design of extensions that can support their use without programming.

Beyond this dissertation, it is our hope that CT+A will help to lower the barriers to sharing advanced student modeling methods between researchers, which in turn may accelerate progress within a cumulative science of student modeling (cf. Desmarais & Baker, 2012; Paquette et al., 2018; Sottilare et al., 2017). Support for plugging in and sharing student modeling methods can help tutor authors and researchers not only in comparing against each other's' models (e.g., by evaluating systems that use these models in classroom experiments), but even in *building upon* and *contributing to* others' student modeling work (cf. Kery & Myers, 2018; Sottilare et al., 2017; Stamper et al., 2016). Such support might even help increase the number of close-the-loop studies that researchers undertake. We also hope that architectures like CT+A will result in broader representation of advanced student modeling methods in both research systems and in real-world educational software.

In *Parts Two* and *Three* of this dissertation, I use the newly-developed CT+A architecture to enable and support the next steps of my research: the iterative co-prototyping of real-time teacher support tools (see *Part Two*), and the classroom evaluation of a specific prototype (*Lumilo*) that emerged from this iterative prototyping process (see *Part Three*).

Part Two

Co-prototyping Real-time, Wearable Teacher Augmentation

Part Two of this dissertation focuses on the iterative prototyping of a new form of real-time teacher augmentation, building upon findings from my prior design research.

Findings from my initial design research (*Chapter 1*) laid the foundation for a broad research program around real-time teacher augmentation. These design studies revealed strong needs for classroom AI systems that can effectively support teachers in addition to their students, enabling human teachers to remain in control of their classrooms, while freeing them up to do what they are uniquely good at (see *Conclusions, Contributions,* and *Future Directions* for further discussion).

Building on these findings, in *Part Two* I decided to narrow my scope, at least for an initial prototype, to the design of tools that specifically support real-time teacher *awareness and decision-making* in AI-enhanced classrooms, as opposed to system *customization and control*.

I next wanted to gain a more concrete sense of which real-time analytics would be most helpful to K-12 teachers during ITS class sessions, and how teachers would envision actually using such analytics during a class session, to inform their in-the-moment decision-making.

In addition, I decided to further explore the idea of using heads up displays such as smart glasses, given teachers' desire to keep their heads up and their attention focused on the classroom (see *Section 1.5*), and given the enthusiasm around this concept that I had observed during speed dating (see *Section 1.6*). The decision to consider such interfaces at all in *Part Two* – rather than restricting these narrower explorations to hardware interfaces that were already lower cost, more widely adopted, and more familiar in K-12 classroom settings – represented a conscious choice to innovate on a longer timescale (see Harrison, 2018). This issue discussed further in *Part Four*, which begins to explore how such technologies might be prepared for wider use, beyond a research context.

To these ends, I conducted a series of iterative, participatory design studies with a total of 16 middle school math teachers across nine schools and six school districts who currently use adaptive learning technologies in their classrooms (see *Section 4.2*).

Chapter 4

Lumilo: Real-time, Wearable Cognitive Augmentation that Facilitates Teacher–AI Co-orchestration of Personalized Classrooms

This chapter is based in part on the following publications:

- Holstein, K., Hong, G., Tegene, M., McLaren, B. M., & Aleven, V. (2018). The classroom as a dashboard: Co-designing wearable cognitive augmentation for K-12 teachers. In *Proceedings of the Eighth International Learning Analytics & Knowledge Conference (LAK 2018), (pp. 79-88)*. ACM.
- Holstein, K., McLaren, B. M. & Aleven, V. (2019). Co-designing a real-time classroom orchestration tool to support teacher–AI complementarity. *Journal of Learning Analytics (JLA)*.

4.1 Background and Motivation

Building on findings from my initial design research with K-12 teachers (*Chapter 1*), I next conducted a series of iterative, participatory design studies with middle school math teachers who currently use adaptive learning technologies in their classrooms. I began with lo-fi experience prototyping and participatory comicboarding sessions (Hiniker, Sobel, & Lee, 2017; Moraveji, Liu, Ding, O'Kelley, & Woolf, 2007), to validate teachers' desires for real-time analytics, further probe underlying needs, and explore how teachers envisioned *actually using* this information during a class session. I also further explored the idea of "teacher smart glasses", to understand their unique affordances for orchestrating personalized class sessions. Then, following a mid-fidelity experience prototyping phase using the Microsoft HoloLens 1, I developed a fully-functional prototype of a mixed reality smart glasses based real-time analytics tool called *Lumilo*.

Many existing real-time orchestration tools have been designed with the assumption that a class of students progresses through instructional activities in a relatively synchronized manner (cf. van Leeuwen, 2015; but see Olsen, 2017). Understanding how best to support teachers in orchestrating highly-differentiated, non-synchronous classrooms, such as those using AI tutoring systems, remains an important and challenging research problem. Orchestration support for such classrooms must alleviate the implementation challenges that these classrooms raise for the

teacher (e.g., see *Chapter 1*, Alphen & Bakker, 2016; Bingham, Pane, Steiner, & Hamilton, 2018; Holstein et al., 2017b; Holstein, Hong, et al., 2018).

Prior work has begun to investigate the potential of emerging wearable technologies for real-time teacher support (e.g. Quintana, Quintana, Madeira, & Slotta, 2016; Zarraonandia, Aedo, Díaz, & Montero, 2013). Such technologies hold great promise to enhance teacher awareness, while allowing teachers to keep their heads up and eyes focused on their classroom – acknowledging the highly active role teachers play in personalized classrooms (Holstein et al., 2017a; 2017b; Quintana, et al., 2016, Schofield et al., 1994). While prior research suggests that teachers may prefer wearables over handheld devices for use in personalized classrooms (e.g., Quintana, et al., 2016), this work has not involved the human-centered design and evaluation of an actual wearable orchestration tool. Furthermore, while prior work has tested the use of smart glasses to help students provide live feedback to their instructors in university lecture contexts (Zarraonandia, et al., 2013), the present work represents the first exploration in the literature of the affordances of smart glasses to support teachers in orchestrating personalized classroom sessions.

Together with the design explorations presented in *Chapter 1* and *9,* and the field studies presented in *Chapters 6* through *8* the present work represents the first broad exploration in the literature of teachers' needs for real-time analytics and orchestration support in personalized classrooms, as well as the first exploration in the literature of the use of wearable, heads-up displays to support teachers in orchestrating personalized classrooms (see items 1 and 2 under *Summary of Contributions – "First design exploration of needs for real-time teacher analytics and orchestration support"* and *"First design exploration and prototypes of wearable, heads-up displays to support orchestration of personalized classrooms"*).

4.2 Overview of Methods

I conducted a series of iterative, participatory design studies with a total of 16 middle school math teachers across nine schools and six school districts. All participating teachers had previously used adaptive learning technologies in their classrooms, and 12 out of 16 had used an ITS as a regular component of their teaching (see Table 4-1). As detailed in the following sections, these studies included activities such as experience prototyping (Buchenau & Suri, 2000), role-playing and bodystorming exercises (Oulasvirta, Kurvinen, & Kankainen, 2003), participatory sketching and comicboarding (Kusunoki, Sarcevic, Zhang, & Yala, 2015; Moraveji, Li, Ding, O'Kelley, & Woolf, 2007; Tohidi, Buxton, Baecker, & Sellen, 2006), and behavioral mapping (Hanington & Martin, 2012; Veitch, Salmon, & Ball, 2007). In Chapter 5, I present Replay Enactments, a novel prototyping method for dynamic, data-driven algorithmic experiences, which I developed to conduct higher-fidelity prototyping sessions with teachers, prior to piloting in actual classroom settings (Holstein et al., 2019a; Holstein, Hong, et al., 2018).

As discussed in the following sections, choices of design research methods were made iteratively and adaptively, based on our team's areas of greatest uncertainty at a given stage of the process.

Table 4-1. Demographic information for schools participating in prototyping studies

School	Region	Free/Reduced Price Lunch	# of teachers	# of teachers with ≤ 2 years' experience
C	Suburban	23%	1	0
E	Rural	34%	4	0
F	Suburban	78%	1	1
G	Urban	36%	4	1
H	Urban	67%	1	0
I	Urban	63%	2	0
J	Suburban	99%	1	0
K	Suburban	71%	1	1
L	Urban	87%	1	1

4.3 Iterative Low-Fidelity Prototyping

To further understand teachers' needs and desires for real-time awareness support, before developing specific prototypes, I conducted a sequence of three lo-fi experience prototyping (Buchenau and Suri, 2000) and participatory comicboarding (Moraveji et al., 2007) sessions with middle school math teachers. For all studies, researchers traveled to schools to work with teachers in their own classrooms.

In each study session, a teacher viewed a computer screen showing a full-screen image of a classroom full of students working with adaptive learning software. A researcher asked the teacher to put on a pair of plastic eyeglass frames, which the teacher was asked to pretend were "smart glasses." As soon as the teacher put on these glasses, a researcher pressed a button on the computer, triggering additional layers of information to appear in front of the image (simulating the experience of using actual smart glasses). Floating text labels appeared over individual students' heads, alerting teachers to students' current detected knowledge or behavioral states, in accordance with common teacher "superpower" ideas from my earlier design studies (discussed

in *Chapter 1*). For example, by looking around the classroom, teachers could instantly see that certain students were currently struggling in the software, might be off-task, or were frequently making careless errors. In addition, two class-level dashboards appeared against the front wall of the classroom, visible only through the "smart glasses," based on teachers' expressed desires for real-time information at the class-level. One of these dashboards showed a list of skills that had been practiced by multiple students in the class, but mastered by very few students, and the other dashboard showed a sorted list of common errors that multiple students in the class had recently exhibited.

The image showed a single instant during a class session, frozen in time, and the teacher was asked to think aloud while imagining how they might, or might not, act on the information they were seeing through their glasses if this were an actual class session. Teachers were encouraged to remark on any information that was displayed to them, but which they did not find useful, as well as information that was not visible but which might inform their decision-making. For example, although one of teachers' "superpower" ideas was to be able to see when students are frequently making "careless errors," all teachers participating in this prototyping study expressed discomfort with the idea of a computer making judgments about students' motivation (e.g., "carelessness"), viewing this as a judgement that a human teacher may much be better equipped to perform than a computer system.

Figure 4-1. Working with a K-12 teacher to generate concepts and potential use scenarios during a low-fidelity prototyping session.

To facilitate brainstorming, teachers were also provided with a large, printed copy of the same classroom image shown on-screen, but with blank rectangles in place of the individual student labels and classroom analytics displays (see Figure 4-1). Throughout each session, teachers could

use these blank spaces to sketch out new ideas for real-time information that might be displayed through the glasses. Each time a teacher generated an idea for new information, a researcher would press the teacher to describe how they envisioned using that information during a real class session. I found that the process of generating hypothetical use cases for particular analytics often led teachers to refine their ideas, as they realized that more, or different kinds of information might be needed to support particular decisions. As in the "superpowers" study, the ideas that a teacher generated during one study were ultimately incorporated into the version of the prototype (i.e., the image and overlaid analytics) shown to the next teacher.

At opportune moments throughout each study, researchers also probed teacher reactions to specific classroom scenarios involving the use of smart glasses, using storyboards that were prepared before the study. I took a participatory comicboarding approach (Moraveji et al., 2007), typically leaving the final panel or two of a comicboard blank. This allowed teacher to fill in the details of how *they* would imagine a classroom scenario progressing, or what decisions and actions they might take in that scenario, rather than relying entirely on a researcher-generated sequence of events.

During the first lo-fi prototyping session, I found that it was challenging for the teacher to imagine the actual experience of using mixed-reality smart glasses in the classroom. So, for the second and third sessions, I added an experience prototyping phase at the beginning of the study, using actual mixed-reality smart glasses (although with Wizard of Oz'd analytics, presented at a single instant in time). I used the Microsoft HoloLens 1[12], which made it possible to place readily available, default HoloLens assets at fixed spatial positions throughout a teacher's classroom. Although the form factor of the HoloLens 1 was not ideal for regular use in classrooms, I used this device for prototyping purposes given that it was the lightest-weight option available at the time with sufficient spatial mapping capabilities to reliably embed mixed reality displays throughout a classroom space (Holstein, Hong, et al., 2018). When teachers began the lo-fi prototyping and sketching exercises, they were then able to refer back to the experience of using this device.

4.4 Iterative Mid-Fidelity Prototyping

I next moved on to higher-fidelity prototyping sessions, given that I had observed strongly positive reactions to the concept of teacher smart glasses in early prototyping sessions and had also begun to gain a more detailed understanding of teachers' real-time information needs. I conducted an iterative sequence of prototyping sessions with 5 middle school math teachers. As in earlier prototyping studies, researchers traveled to middle school sites and worked with teachers in their own classrooms.

[12] *https://www.microsoft.com/en-us/hololens/hardware*

Each study session lasted for 90 minutes. The teacher wore the HoloLens for the first hour and participated in experience prototyping activities (Buchenau and Suri, 2000), experimenting with different combinations and spatial configurations of analytics displays while generating ideas for potential use cases. Following this experience prototyping phase, teachers participated in a 30-minute semi-structured post-interview in which they had the opportunity to reflect on their experiences and provide more detailed design feedback. For these and subsequent prototyping studies, I narrowed my focus specifically to the context of middle school math classrooms using ITSs for equation solving.

In order to present teachers with a range of design alternatives, I used a modified version of HoloSketch[13], a HoloLens application for rapid prototyping of mixed-reality experiences. Using HoloSketch, I was able to position two-dimensional assets, including mock-ups of student- and class-level analytics displays created in Photoshop, throughout a teacher's physical classroom space. For example, when a teacher put the HoloLens on, they could see indicator symbols floating over empty student seats, and class-level analytics displays appearing as "wall decorations" that the teacher was able to reposition as they saw fit (see Figure 4-2, left). Throughout each prototyping session, the teacher had the opportunity to move about their classroom. Teachers were asked to think-aloud during these sessions, imagining what actions they might take in response to the displayed analytics if this were a real class session, and what other information might support them in making such decisions.

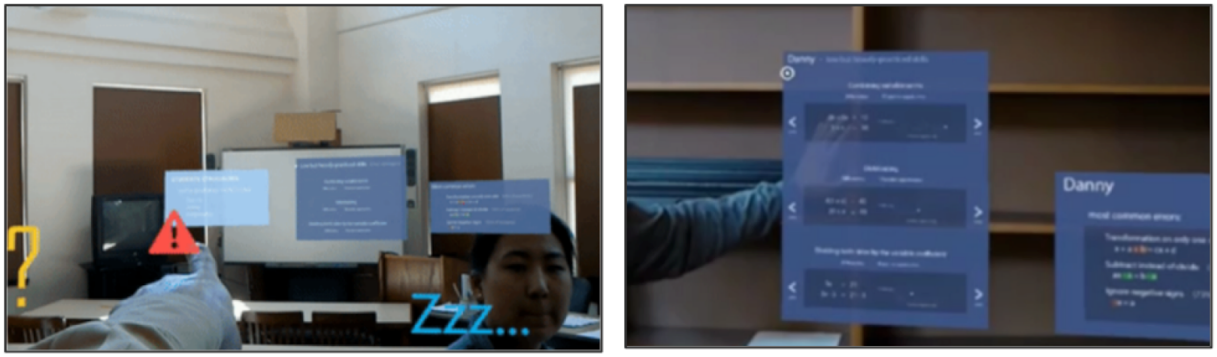

Figure 4-2. Screenshots from the teacher's point-of-view during a mid-fidelity prototyping session. Left: the teacher thinks-aloud while positioning combinations of analytics displays throughout the classroom. Right: the teacher discusses design alternatives in a "gallery" at the back of the classroom (from Holstein et al., 2019a). Note: student names shown in this figure are fabricated.

In the first mid-fi experience prototyping session, I included all of the indicator symbols and analytics displays that teachers had consistently requested up until this point (in lo-fi prototyping studies). Then, in-between sessions, my collaborators and I rapidly iterated on the design of

[13] *https://github.com/Microsoft/MRDesignLabs/tree/master/ReleasedApps/HoloSketch*

individual student- and class-level displays, incorporating new ideas that teachers had generated during the previous session. Since the design mock-ups were synchronized with the HoloLens app as 2D assets, we were also able to make modifications during a session based on teachers' live design feedback, by editing these assets on a laptop as a teacher viewed them (in an appropriate spatial context) through the HoloLens.

Between prototyping sessions, we also reflected on our areas of greatest uncertainty. For each open question, we mocked up several design alternatives. Towards the end of each session, I brought the teacher to the back of their classroom, where (in mixed-reality) we had arranged an immersive "gallery" of these new design alternatives (see Figure 4-2, right). Teachers could reposition these information displays and experiment by decorating their classrooms with different combinations and arrangements of displays, all while thinking aloud and providing design feedback. Based on this feedback, we iterated on these designs prior to the next prototyping session, providing opportunities to validate previous teachers designs (and the needs underlying these designs) with new teachers in subsequent sessions.

4.5 Design Findings from Low- to Mid-Fidelity Prototyping

Our research team worked through transcriptions of approximately 12 hours of audio/video-recorded prototyping sessions, across 8 teachers, to synthesize design findings through Interpretation Sessions and Affinity Diagramming (Beyer & Holtzblatt, 1997; Hanington & Martin, 2012). Following a series of Interpretation Sessions, the resulting 655 quotes were iteratively synthesized into 77 level-1 themes, 23 level-2 themes, 10 level-3 themes, and 7 level-4 themes. Key high-level findings (level-4 themes) are summarized below:

Student-level indicators. Five major categories of student learning and behavioral states emerged from these co-design sessions, shown in Figure 4-3. Teachers strongly preferred to keep these indicators visually simple, displaying a single graphical symbol above each student's head (as in Figure 4-4) to avoid information overload during a class session. However, it was also important to teachers that they could access brief elaborations on-demand (e.g., by having such elaborations appear when looking directly at an indicator, as shown in Figure 4-4), which could help teachers better understand why an indicator was appearing for a student at a particular time. In line with my prior findings, all teachers expressed a desire to see positive information about individual students, not just negative information. In particular, teachers wanted to be able to see when students have been performing particularly well recently. Teachers found this valuable for several reasons, including: motivating themselves (since seeing only negative alerts might be discouraging), motivating students (by identifying and praising students who have been doing well lately), and identifying students who may be under-challenged by the software.

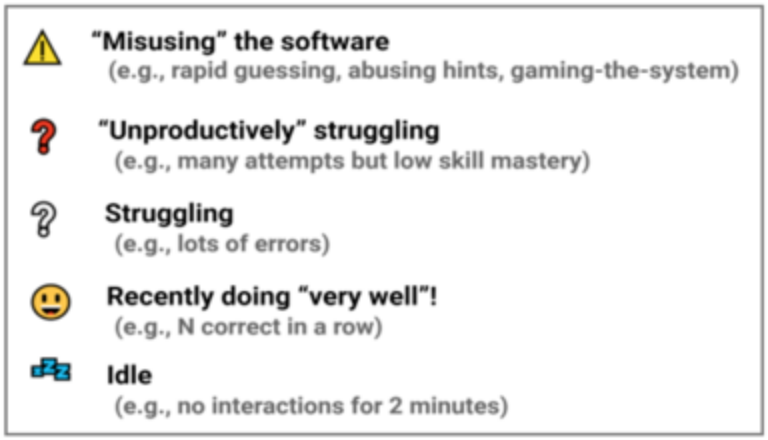

Figure 4-3. Consistently requested categories of real-time indicators in low- to mid-fidelity prototyping sessions (from Holstein, Hong, et al., 2018).

Sequences of student states can be information-rich. In addition to seeing indicators that reflect students' current "states," teachers noted that it would be useful to see sequences of detected states that preceded a student's current state. For instance, if a student is currently "idle" or "misusing the software" in some way, it would be useful to know whether that student was *also* recently struggling. Teachers would then interpret the prior struggle as a possible cause of the student's current behavior, and respond accordingly.

The classroom as a dashboard. During experience prototyping sessions, teachers remarked that it felt natural to reference information displays that were distributed throughout their physical classroom spaces. In the absence of a real-time awareness tool, teacher were used to monitoring their students by scanning the physical classroom (e.g., reading students' faces and body language), and "patrolling" rows of student seats to catch quick glances of individual students' screens. One teacher remarked,

> *"I would also use their body language to judge the situation, but the initial [alert] would help, so I know to go over there."*

Teachers also revealed that they *already* used their classrooms as distributed information displays. For example, during a typical class session, teachers would often leave notes and images for themselves on boards or projected displays, to reference throughout the session.

Figure 4-4. Design mock-ups based on findings from low- to mid-fidelity prototyping sessions (from Holstein, Hong, et al., 2018). Top: Teacher's default view of the class. Each student has an indicator display floating above their head, and class-level analytics displays are positioned at the front of the class. Bottom: "Deep-dive" screens shown if a teacher 'clicks' on an indicator. Note: student names shown in this figure are fabricated.

Needs for selective shared awareness. All participating teachers noted that the analytics they found most useful in informing their real-time decision-making tended to be ones they would not be comfortable sharing with students. Teachers expected that these analytics could do more harm than good, by promoting unproductive social comparison and competition among their students (cf. Aguilar, 2018). As one teacher put it,

> *"In middle school, kids don't know what they don't know, [but] kids care so much about how they're seen by others [… they] don't want to look stupid or feel stupid."*

However, teachers also noted that they would want a mechanism to selectively share particular analytics during the course of a class session. Five out of eight teachers suggested it would be useful to customize the shared visibility of particular analytics on a class-by-class basis. All of these teachers predicted an interaction effect in which real-time analytics might *motivate* higher-achieving classes by promoting healthy competition among students, while *demotivating* lower-achieving classes.

Support anonymous teacher-student communication ("Invisible hand raises"). Although most of teachers' design feedback focused on ways real-time analytics could help *them* regulate students' learning, some teachers emphasized the importance of also creating opportunities to develop students' help-seeking skills (Aleven, Roll, McLaren, & Koedinger, 2016). Several teachers proposed the idea of giving students an "Ask the teacher" button within the tutoring software, which would trigger a "raised hand" symbol within the glasses. Teachers expected that, by providing students with a mechanism to request help that was not easily visible to other students, more students would feel comfortable requesting help (cf. Schofield et al., 1994). Otherwise, as one teacher put it, *"for a number of students in my class, unless I [walk over], they are never going to say anything"* (cf. *Chapter 2* and Holstein et al., 2017a)

4.6 Development of a Higher-fidelity Prototype: *Lumilo*

Up until this point, all prototyping sessions had relied upon Wizard of Oz'ing analytics, presented "frozen in time" at a single instant of a class session. However, the behavior of a real-time analytics tool can be heavily dependent on the dynamics of specific data-generating contexts in combination with specific analytic methods/algorithms. I next wanted to begin prototyping the experience of using smart glasses to monitor a class session unfolding over time, using real student data and analytics.

Based on findings from low- to mid-fidelity prototyping, I developed a fully-functional prototype of a mixed reality smart glasses based real-time analytics tool called *Lumilo* (see Figure 4-5), using the Microsoft HoloLens 1, Unity3D[14], the HoloToolkit[15], and the extended CT+A architecture for ITS authoring and deployment (see *Chapter 3* and Holstein, Yu, et al., 2018).

Lumilo tunes teachers in to the rich analytics that ITSs generate: It presents real-time indicators of students' current learning, metacognitive, and behavioral "states", projected in the teacher's view of the classroom. The specific indicators displayed by *Lumilo* (see Figure 4-3) are ideas generated and refined by teachers throughout the design and prototyping process (as described in *Sections 4.3* through *4.5,* and in *Chapter 5*) and implemented using established student modeling methods (Desmarais & Baker, 2012), using the *CT+A* architecture (introduced in *Chapter 3*). By

[14] *https://unity3d.com*
[15] *https://github.com/Microsoft/HoloToolkit-Unity*

directing teachers' attention, in real-time, to situations the ITS may be ill-suited to handle – and by aiding teachers in determining how to address these situations – *Lumilo* is designed to facilitate productive mutual support or *co-orchestration* (Prieto, 2012) between the teacher and the ITS, by leveraging complementary strengths of each (cf. Alkhatib & Bernstein, 2019; Davidoff et al., 2007; Holstein et al., 2014; 2017b; 2019a; 2019b; Kamar, 2016; Lake et al., 2017; Lubars & Tan, 2019).

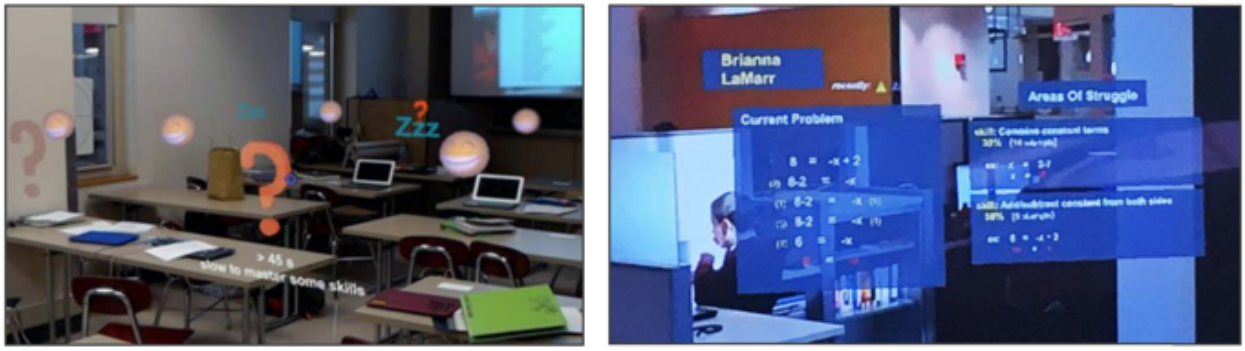

Figure 4-5. Point-of-view screenshots from teachers using *Lumilo* (from Holstein, Hong, et al., 2018). Left: A teacher's view of student indicators, immediately following a pilot study in a live classroom. Right: A teacher's view of a classroom while wearing *Lumilo* (photo taken with no students present in the room, to preserve student privacy). Note: student names shown in this figure are fabricated.

The use of transparent smart glasses allows teachers to keep their heads up and focused on the classroom, enabling them to continue monitoring important signals that may not be captured by the tool alone (e.g., student body language and looks of frustration (Holstein, Hong, et al., 2018; Holstein et al., 2017b)). The smart glasses provide teachers with a private view of actionable, real-time information about their students, embedded throughout the classroom environment, thus providing many of the advantages of ambient and distributed classroom awareness tools (e.g., Alavi & Dillenbourg, 2012; van Alphen & Bakker, 2016), without revealing sensitive student data to the entire class (van Alphen & Bakker, 2016; Holstein, Hong, et al., 2018).

Over the course of the design and prototyping process, *Lumilo*'s information displays shifted towards strongly minimalistic designs (with progressive disclosure of additional analytics only upon a teacher's request), in accordance with the level of information teachers desired and could handle in fast-paced classroom environments. *Lumilo* presents mixed-reality displays of three main types, visible through the teacher's glasses: student-level indicators, student-level "deep-dive" screens, and class-level summaries (as shown in Figure 4-4). Student-level indicators and class-level summaries are always visible to the teacher by default, at a glance.

Student-level indicators display above corresponding students' heads (based on teacher-configurable seating charts), and class-level summaries can display at teacher-configurable locations throughout the classroom (Holstein, Hong, et al., 2018). As shown

in Figure 4-5 (left), if a teacher glances at a student's indicator, *Lumilo* automatically displays a brief elaboration about the currently displayed indicator symbol (i.e., how long the alert has been active and/or a brief explanation of why the alert is showing). If no indicators are currently active for a student, *Lumilo* displays a faint circular outline above that student (see Figure 4-4). In addition, to give teachers "eyes in the back of their heads" – a commonly requested ability in the early "teacher superpowers" exercise) (see *Chapter 1*) – *Lumilo* provides ambient, spatial sound notifications to enhance teachers' awareness of events lying outside of their field of vision. For example, while a teacher is working one-on-one with a student at one end of the room, the teacher may hear a gentle ping that seems (from the teacher's perspective) to be emanating from the location of a given student behind the teacher – indicating that the far-away student has consistently, repeatedly been gaming-the-system or abusing hints in the software (Holstein, Hong, et al., 2018; Holstein et al., 2019a). This is one of several features of *Lumilo* that emerged by iteratively prototyping *Lumilo* using Replay Enactments (a new prototyping method described in *Chapter 5*.

If a teacher clicks on a student's indicator (using either a handheld clicker or by making a 'tap' gesture in mid-air), *Lumilo* displays "deep-dive" screens for that student. As shown in Figure 4-4 (top) and Figure 4-5 (right), these screens include a "Current Problem" display, which supports remote monitoring, showing a live feed of a student's work on their current problem. Each problem step in this feed is annotated with the number of hint requests and errors the student has made on that step. In classroom observations (see *Part Three*), I have found that because *Lumilo* enables monitoring of student activities from a distance (i.e., across the room), teachers using *Lumilo* often interleave help across students: During a pause while helping one student at that student's seat, the teacher might quickly peek at another struggling student's recent activities from across the room, and then call over to that student to provide quick guidance and try to get the student "unstuck" (Holstein et al., 2018b; 2019a).

The deep-dive screens also include an "Areas of Struggle" screen, which displays the three skills for which a student has the lowest probability of mastery. For each skill shown in "Areas of Struggle", the student's estimated probability of mastery is displayed, together with a concrete example of an error the student has made on a recent practice opportunity for the skill. Think-alouds during Replay Enactments sessions (see *Chapter 5*) suggested that providing these brief, concrete examples of student errors (cf. Bull and Kay, 2016; Kay, 2000) was useful not only for aiding teachers in interpret where exactly students were struggling, but also in enabling teachers to perform further diagnosis of the nature of student difficulties and/or second guess the ITS's skill labeling. In classroom observations, I have found that teachers often focus their conversations with individual students around a concrete example of an error the student has recently made – for example, initiating a conversation with a student by asking them to explain how they would solve a similar problem to one on which they had erred (Holstein et al., 2019a).

In addition, in the current version of *Lumilo*, a class-level summary display is available to the teacher: the "Low Mastery, High Practice" display (illustrated in Figure 4-4, left). This display shows the three skills that the fewest students in the class have mastered (according to Bayesian Knowledge Tracing (Corbett and Anderson, 1995)), at a given point in the class session, out of those skills that many students in the class have already had opportunities to practice within the software (Holstein, Hong, et al., 2018). In classroom observations, I have found that this display is somewhat rarely used by teachers to inform instructional interventions, given that different students tend to be working on different material in the software at any one moment. However, teachers occasionally use this feature to pause the class and provide a brief mini-lecture on a topic with which many students appear to be experiencing difficulties. Based on teacher feedback from prototyping sessions and classroom studies, a future version of *Lumilo* may provide teachers with recommendations of *small groups* of students who are struggling with similar material at a given point during a class session, in addition to providing class- and individual-level analytics (see *Chapters 8* and *10*).

To support subsequent design explorations using the *Lumilo* prototype (discussed in *Chapters 5 and 6*), my collaborators and I architected the initial prototype of Lumilo in a highly modular fashion, to enable rapid design iteration in-between future prototyping sessions, and even to make small adjustments within a single prototyping session, based on live teacher feedback. For example, alternative student modeling (detector) algorithms intended to measure the same teacher-identified construct (such as "unproductive persistence") could be interchanged for comparison during a prototyping session. All detectors included in the next round of prototyping sessions were drawn from the Learning Analytics, Educational Data Mining, AI in Education, and User Modeling literatures—where many automated detectors of student learning and behavior have been introduced and validated, based upon students' interactions within the software (e.g., Aleven et al., 2016; Beck & Gong, 2013; Desmarais & Baker, 2012; Käser et al., 2016). For example, in order to drive a real-time indicator of "unproductive persistence" (here defined as a phenomenon in which an AI tutor is *failing to help the student learn*, on one or more skills; see Holstein, 2018) I explored the use of simpler methods such as Beck and Gong's detector of "wheel-spinning" (Beck & Gong, 2013), in addition to more sophisticated methods such as Käser et al's "predictive stability" (Käser et al., 2016). Each detector was implemented in a parameterized fashion, so that aspects of a detector's behavior (e.g., tunable alert thresholds) could be adjusted during and in-between prototyping sessions, based on teachers' feedback.

I also developed a new logging library for *Lumilo*, which appropriates the HoloLens 1's spatial mapping capabilities as a means of automatically logging teachers' actions in a physical classroom space over the course of a class session (i.e., to automate much of the manual coding process described in *Chapter 2*). For example, using these "mixed reality sensors," *Lumilo* can record time-stamped logs of a teacher's physical proximity to a given student in the class moment-by-moment, as well as the teacher's absolute location in the classroom, their proximity

to pre-specified landmarks (such as the teacher's desk or whiteboard), the target of a teacher's gaze, and all teacher interactions within the tool interface. These logs are recorded to DataShop, a major educational data repository (Koedinger et al., 2010). Unlike most prior work on physical teaching analytics (e.g., An et al., 2019; Echeverria et al., 2018; Martinez-Maldonado, 2019; Martinez-Maldonado et al., 2018; but see Prieto, Dillenbourg, Sharma, & Jesús, 2016), this mixed reality sensor approach uses an "inside out" approach to teacher tracking, and thus does not require instrumenting the classroom space with external sensors or "beacons". Rather, this approach relies entirely on the HoloLens 1's built in sensors and spatial mapping algorithms for tracking of teachers' behavior (see item 5 under *Summary of Contributions – "First investigations of relationships between teacher movement/monitoring and student behavior and learning in AI-supported classrooms"*).

4.7 Conclusions

Together with the design explorations presented in *Chapter 1* and *9,* and the field studies presented in *Chapters 6* through *8* the present work represents the first broad exploration in the literature of teachers' needs for real-time analytics and orchestration support in personalized classrooms (see item 1 under *Summary of Contributions – "First design exploration of needs for real-time teacher analytics and orchestration support"*), as well as the first exploration in the literature of the use of wearable, heads-up displays to support teachers in orchestrating personalized classrooms (see item 2 under *Summary of Contributions – "First design exploration and prototypes of wearable, heads-up displays to support orchestration of personalized classrooms"*).

Many existing real-time orchestration tools have been designed with the assumption that a class of students progresses through instructional activities in a relatively synchronized manner (cf. van Leeuwen, 2015; but see Olsen, 2017). Understanding how best to support teachers in orchestrating highly-differentiated, non-synchronous classrooms, such as those using AI tutoring systems, remains an important and challenging research problem (e.g., see *Chapter 1*, Alphen & Bakker, 2016; Bingham, Pane, Steiner, & Hamilton, 2018; Holstein et al., 2017b; 2019a; 2019b; Holstein, Hong, et al., 2018).

Emerging wearable technologies technologies hold great promise for real-time teacher support in personalized classrooms – acknowledging the highly active role teachers play in such classrooms (Holstein et al., 2017a; 2017b; Schofield et al., 1994) by allowing them to keep their heads up and their eyes focused on their students rather than a screen (Holstein et al., 2017b; Quintana, et al., 2016). While prior research suggests that teachers may prefer wearables over handheld devices for use in personalized classrooms (e.g., Quintana, et al., 2016), this work has not involved the human-centered design and evaluation of an actual wearable orchestration tool. Furthermore, while prior work has tested the use of smart glasses to help students provide live

feedback to their instructors in university lecture contexts (Zarraonandia, et al., 2013), the present work represents the first exploration in the literature of the affordances of smart glasses to support teachers in orchestrating personalized classroom sessions.

In the next chapter, *Chapter 5*, I introduce a novel prototyping method, Replay Enactments, to involve teachers in prototyping and iteratively shaping the experience of using *Lumilo* in a classroom – including *algorithmic elements* of *Lumilo*'s design, which could not be easily prototyped using the lower-fidelity methods presented in this chapter.

Chapter 5

Replay Enactments: A Prototyping Method for Data-driven Algorithmic Experiences

This chapter is based in part on the following publications:

- Holstein, K., Hong, G., Tegene, M., McLaren, B. M., & Aleven, V. (2018). The classroom as a dashboard: Co-designing wearable cognitive augmentation for K-12 teachers. In *Proceedings of the Eighth International Learning Analytics & Knowledge Conference (LAK 2018), (pp. 79-88)*. ACM.
- Holstein, K., McLaren, B. M., & Aleven, V. (2018a). Informing the design of teacher awareness tools through Causal Alignment Analysis. In *Proceedings of the 13th International Conference of the Learning Sciences (ICLS 2018)* (pp. 104-111).
- Holstein, K., McLaren, B. M. & Aleven, V. (2019a). Co-designing a real-time classroom orchestration tool to support teacher–AI complementarity. *Journal of Learning Analytics (JLA)*.

5.1 Background and Motivation

To rapidly prototype the experience of using *Lumilo* in a classroom, *prior* to running studies with the system in live classrooms with actual students (which can be costly in K-12 settings, and may even be harmful to students if the prototype's effects are poorly understood), I created a new prototyping method for dynamic, data-driven algorithmic systems: Replay Enactments (Holstein, Hong, et al., 2018; Holstein, et al., 2018a; 2019a).

Replay Enactments uses authentic data and (imperfect) algorithms to reveal important nuances that other methods – such as Wizard of Oz studies (Lovejoy, 2018; Odom, Zimmerman, Forlizzi, Dey, & Lee, 2012) – may be ill-suited to surface (e.g., UX impacts of a prototype's false positives and negatives or issues that arise only in particular data-generating contexts; see Dove et al., 2017; Holstein, Wortman Vaughan, et al., 2019; and a brief discussion of "Global Design" challenges in Zimmerman & Forlizzi, 2019). As such, Replay Enactments represents one response to recent calls within the HCI, HAI, and LA communities (e.g., Dennerlein et al., 2018; Doshi-Velez & Kim, 2017; Dove et al., 2017) for "new kinds" of prototyping methods that can address challenges of prototyping data-driven algorithmic systems (see item 4 under *Summary of Expected Contributions – Novel design and prototyping methods)*. The behavior of such systems can be highly dependent on interactions between particular data-generating contexts (e.g., specific socio-cultural and classroom contexts from which educational data was collected) and

particular algorithms (e.g., specific machine learning models trained on specific datasets with specific biases), which cannot easily be imagined ahead of time by system designers (Holstein, Wortman Vaughan, Daumé III, Dudík, & Wallach, 2019; Ocumpaugh, Baker, Gowda, Heffernan, & Heffernan, 2014; Ogan et al., 2012; Yang, Sciuto, Zimmerman, Forlizzi, & Steinfeld, 2018). Developing methods to engage non-technical stakeholders in shaping *algorithmic elements* of complex, data-driven AI systems remains a central open challenge for the UX design of data-driven AI systems (e.g., Baumer, 2017; Chen & Zhu, 2019; Dennerlein et al., 2018; Kyung Lee et al., 2018; Prieto-Alvarez, et al., 2018; Zhu & Terveen, 2018).

5.2 Replay Enactments

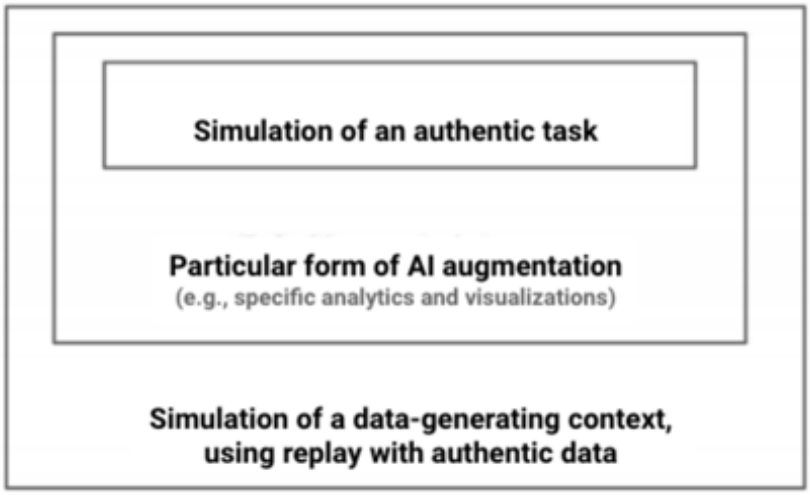

Figure 5-1. A high-level diagram illustrating modular, nested components of a RE prototyping study. Components at each level can be swapped to compare across multiple options, while components at other levels are held constant (figure from Holstein et al., 2019a).

Figure 5-1 shows a general, high-level description of a Replay Enactments (RE) prototyping study. REs involve the simulation of a relevant, dynamic data-generating context (such as a classroom of students working with adaptive learning technologies). To generate this simulated context, authentic (rather than Wizard-of-Oz'd or otherwise fabricated) data streams are replayed at the same speed at which the data were originally generated. Within this simulated context, the user(s) participating in an RE study are equipped to receive two key streams of sensory input: first, a simulated approximation of what the user would typically experience in the target environment (e.g., in an actual classroom); and second, a particular form of cognitive augmentation (i.e., a specific tool design, including particular choices of analytics and visualizations). Given a simulated context and a particular from of cognitive augmentation, the user is asked to complete (an approximation of) an authentic, complex task. Figure 5-1 shows the components of an RE session as nested boxes, to indicate that each can be swapped out for

comparison purposes. For example, the same classroom analytics system design might be tested across replays of datasets generated from multiple, diverse classroom contexts (see Figure 5-2).

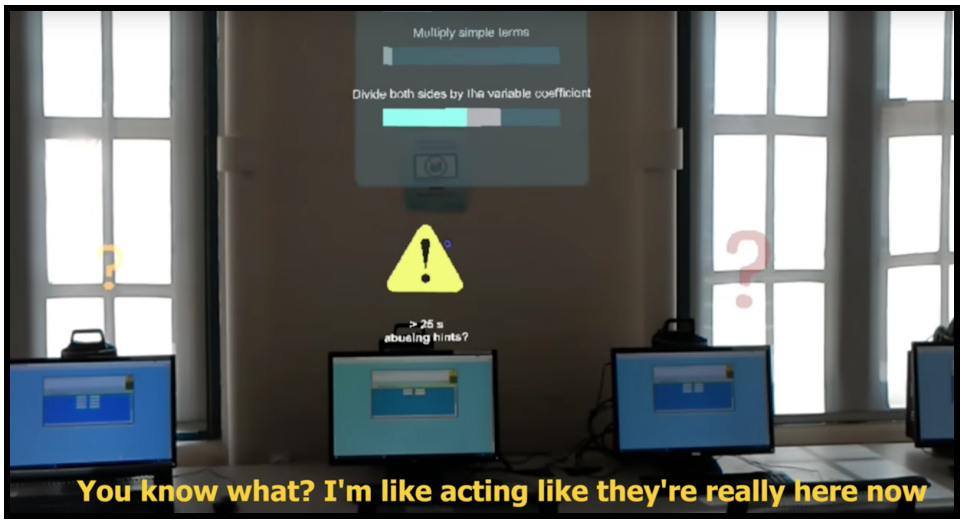

Figure 5-2. Still image from a video[16] showing a teacher's point of view during an RE session, while iteratively prototyping *Lumilo*. The teacher is in a computer lab on our university's campus, with no students present. An ITS interface is displayed on each computer screen, and previously collected student interaction data from a full class of students is replayed, at original speed, through these interfaces (i.e., with different replayed students assigned to different computers). The teacher wears a particular form of augmentation (a particular version of *Lumilo*, with particular choices of algorithms/analytics), and is asked to think aloud while walking throughout the classroom and helping "students" based on the analytics they see. Teacher dialogue is displayed at the bottom of the frame; this teacher notices he has begun talking to students as if they were actually present.

In addition to generating qualitative insights (e.g., through user think-alouds during experience prototyping), REs can be used to provide early insight into the impacts different tool designs (e.g., particular choices of algorithms/analytics) might have on user decision-making and behavior. Since the use of data replays removes the possibility that user behavior will influence the data streams being replayed (and thus removes the possibility of feedback loops), REs can also support early quantitative evaluations of how effectively a predictive analytics system (e.g., an early warning system) might steer user's attention (see Holstein, Hong, et al., 2018; Holstein et al., 2018a).

Much like other recently proposed prototyping methods in the learning analytics (LA) literature, such as the simulation methods presented in Martinez-Maldonado et al. (2012) and Mavrikis et al. (2016), Replay Enactments involve replaying log data from students' interactions with educational technologies in order to prototype real-time analytics and visualizations with end-users (such as teachers or students). However, in the spirit of recent HCI methods for

[16] Video clips are available at: *https://youtu.be/ELY9p_HijEw*

prototyping radically new experiences, such as User Enactments (Odom, Zimmerman, Davidoff, Forlizzi, Dey, & Lee, 2012), Replay Enactments builds on prior LA approaches by emphasizing embodied role-playing in physical classroom environments (Holstein, Hong, et al., 2018; Holstein et al., 2019a). In my initial work piloting this prototyping method with teachers, I found that pushing teachers to role-play while actually navigating throughout a physical classroom space seemed to contribute to an illusion of "actually being there" (see Figure 5-2). In addition, having the teacher move throughout the classroom provided early insight into potential effects of a classroom's layout and students' seating arrangement relative to this layout (cf. *Chapter 2* and Holstein, et al., 2017a).

Whereas methods like User Enactments typically involve Wizard-of-Oz'd scenarios, Replay Enactments prototype an experience using authentic data and algorithms, evolving over time. Although this requires earlier investment in technical development, doing so can enable earlier, detailed observations of the interplay between human and machine judgments, and the ways in which a system's false positives and false negatives may impact the experience of using a data-driven algorithmic system (cf. Dove, Halskov, Forlizzi, & Zimmerman, 2017; Holstein et al., 2019a).

As such, Replay Enactments represents one response to recent calls within the HCI, HAI, and LA communities (e.g., Dennerlein et al., 2018; Doshi-Velez & Kim, 2017; Dove et al., 2017) for "new kinds" of prototyping methods that can address challenges of prototyping data-driven algorithmic systems (see item 4 under *Summary of Expected Contributions – "Novel design and prototyping methods"*). The behavior of such systems can be highly dependent on interactions between particular data-generating contexts (e.g., specific socio-cultural and classroom contexts from which educational data was collected) and particular algorithms (e.g., specific machine learning models trained on specific datasets with specific biases), which cannot easily be imagined ahead of time by system designers (Holstein, Wortman Vaughan, Daumé III, Dudík, & Wallach, 2019; Ocumpaugh, Baker, Gowda, Heffernan, & Heffernan, 2014; Ogan et al., 2012; Yang, Sciuto, Zimmerman, Forlizzi, & Steinfeld, 2018). For example, a recent study of industry product teams' challenges around algorithmic bias at major companies, including several companies working on learning analytics applications, found that a lack of suitable pre-deployment prototyping methods was a central pain point (Holstein, Wortman Vaughan, et al., 2019).

The following subsections present a concrete case study of the use of Replay Enactments to iteratively prototype *Lumilo*.

5.3 Iteratively Experience Prototyping *Lumilo* through Replay Enactments

Using the initial functional prototype of *Lumilo*, I next conducted an iterative sequence of higher-fidelity experience prototyping sessions, with a total of 10 math teachers across 5 schools. All participating teachers had previously used an adaptive learning technology in their classrooms, and seven out of 10 teachers had used an ITS as a regular component of their classroom instruction.

In each of an initial round of five Replay Enactments study sessions, each held with a single teacher at a time, I brought teachers into a computer lab on our university's campus. At each empty seat in the lab, I had placed a nametag with a fabricated student name before the study session began. On the corresponding computer screen, I had logged into the tutoring software, under that student's name. In addition, using *Lumilo*, I had positioned mixed-reality holograms throughout the computer lab so that indicators, associated with corresponding student accounts in the software, would appear over "student" heads. Class-level analytics displays were also positioned along the walls of the computer lab.

Using a newly developed log replay system, I was able to replay log data from an entire class of students, using datasets previously collected from a multi-classroom study in which middle school students used *Lynnette*, an ITS for linear equation solving (Long & Aleven, 2013; Long, Holstein, & Aleven, 2018; Waalkens, Aleven, & Taatgen, 2013). When a researcher pressed a button in a web-based "controller" interface, the entire class sprung to life, replaying a 40-minute class session from beginning to end, at actual speed. The teacher wore *Lumilo* during this simulation phase and was asked to pretend that this was an actual class session, thinking aloud as they moved throughout the classroom space. If the teacher thought they might focus attention on a particular student at a particular time, based on the information they were seeing, they were asked to verbalize what they might say to that student in-the-moment if the student were actually there. Teachers often became quite immersed in this task. For example, one teacher remarked, about halfway through the a REs session,

> *"You know what? I'm acting like [the students are] really here now [...] I'm thinking that I'm gonna tell them something and [the indicator] is gonna change."*

I ran separate Replay Enactments sessions with a total of five teachers. Each of these sessions began with a 35-minute training and familiarization phase, during which the teacher could acclimate to using the system. This was then followed by a 40-minute simulation phase, during which the teacher was asked to think-aloud, and a 15-minute post-interview to elicit additional design feedback. To prototype the experience of using *Lumilo* under a diverse range of classroom dynamics, I selected one dataset from a "remedial" middle school math class, one dataset from an "advanced" class, and one dataset from an "average" class, where class tiers were based on those assigned by the schools from which these datasets were drawn. I then randomly assigned

datasets to Replay Enactments study sessions, so that the remedial and average classes were replayed for two teachers each, and the advanced class was replayed for the remaining teacher. To account for potential influences of a classroom's spatial layout, I used different computer labs, with a range of spatial layouts, across study sessions.

During Replay Enactments, my goal was to elicit teacher feedback not only on *Lumilo*'s interface design and the visual presentation of analytics, but also on the specific choices of learning analytics that were used to drive *Lumilo*'s real-time indicators and class-level dashboards. During each session's training and familiarization phase, teachers were provided with definitions for each indicator symbol. These included brief summaries of a detector's structure, the main features it relies upon, and the default settings of any free parameters (e.g., alert thresholds) used by an indicator or its corresponding detector.

Within the simulation phase of each session, teachers frequently monitored students' "raw" activity within the software (either by approaching a student's computer terminal and observing their screen, or by opening the student's deep-dive window through the glasses interface). In doing so, they often observed ways in which particular detectors might have been over- or under-sensitive, or may have been overlooking key features of student thinking and behavior entirely.

Such feedback provided opportunities to iterate on the selection and design of detectors and alert policies that drove *Lumilo*'s real-time indicators in-between REs sessions (and sometimes even within a single REs session). For example, over several iterations, the definition of the "struggling" indicator evolved to not only indicate when a student had surpassed a certain recent error rate, but also to provide: (1) a visual indication of whether a student has been avoiding using the software's built-in help functions (i.e., hints) (Aleven et al., 2016); (2) a visual indication of whether a student has remained stuck *despite* having made good use of the software's hints; and (3) a visual indication of *how long* a student has been struggling (with the corresponding "question mark" symbol glowing gradually brighter, the longer the student remained stuck). By the final two REs sessions, teacher observations of under- or over-sensitivity, or mismatches between the analytics and a teacher's own judgments of a student's knowledge or behavior, had become relatively rare.

Examples of other design features that entered the prototype during this iterative refinement process included the ability to set visual "reminders" on an individual student by clicking-and-holding on the student's indicator. Teachers found this useful as a reminder to check back with a student, for example if that student appears to be struggling currently, but it is unclear to the teacher whether the student might overcome this struggle on her/his own within the next few minutes. As one teacher put it,

> *"You want to stay with a kid until they have it mastered but... there's that advantage to saying 'Okay, try a few of these, I'll come back to you.' I've never found a good answer to that one."*

In addition, I found that teachers saw great value in the ability to monitor individual students' activities from a distance, while walking around the classroom or while working face-to-face with a student seated across the room. As such, I enhanced *Lumilo* so that a teacher could have the deep-dive screen "tag along" with them as they walked (as opposed to hanging in space near the corresponding student, visible only when the teacher was looking in that student's direction). Finally, to give teachers "eyes in the back of their heads," a common need revealed by the early "superpowers" design study (*Chapter 1*), I added ambient, spatial sound notifications. For example, if a student was misusing the software, a teacher could privately perceive a soft auditory notification, as if it were emanating from that student's location in the classroom.

5.4 Design Findings from Replay Enactments with *Lumilo*

As before, our research team conducted Interpretation Sessions and Affinity Diagramming to synthesize design findings from transcriptions of approximately 18.5 hours of audio/video recorded think-aloud data and design feedback. The resulting 486 quotes were iteratively synthesized into 43 level-1 themes, 26 level-2 themes, 13 level-3 themes, and 5 level-4 themes. Key high-level findings from this synthesis (level-4 categories) are highlighted below.

I see these design findings as fruitful directions for future work (indeed, several of these findings have subsequently re-emerged in in-vivo classroom studies; see *Chapters 8* and *10*).

Value of continuous, real-time feedback on instruction.
Although *Lumilo* did not provide direct feedback to teachers about their own instruction, teachers frequently inferred potential effects of their instructional interventions (e.g., helping an individual student or providing a brief whole-class lecture) by monitoring changes in student and class state following an intervention. In fact, teachers were often tempted to infer causality even during Replay Enactments sessions, in which no students were actually present. In line with findings from my earlier directed storytelling and speed dating studies (see *Chapter 1*), teachers emphasized that receiving more direct, in-the-moment feedback about the effects of their own teaching on students' learning could help them adjust their instruction on-the-spot, and perhaps even improve their teaching over time (especially if this in-the-moment feedback were constructive).

When many students need help on different topics at the same time, choice can be anxiety-inducing.
During Replay Enactments, teachers realized that when they were made more aware of student struggle during a class session, they also became more aware of their limited ability to actually

help all of their students. The main way teachers proposed addressing this was through dynamically adjustable alert thresholds, which could help them better focus their attention during times when they would otherwise be overloaded (e.g., when many students need their help simultaneously, or in more chaotic classes that require teachers to devote more attention to basic classroom management). As one teacher put it,

> *"I'm going to be able to handle different [numbers of alerts] in different classes […] I'd want to be able to control that."*

Action recommendations *in addition to* awareness support.

As I progressed to higher-fidelity prototyping, teachers consistently noted that it would be helpful to have more explicit action recommendations from the system, to help them prioritize their attention across students and/or to decide how best to help particular students. For example, one teacher suggested that it would sometimes be helpful to receive recommendation for "conversation starters," such as self-explanation prompts they could give a student, targeted to that student's current areas of struggle, to avoid providing "too much" scaffolding. At the same time, it is clear from my early design explorations (see *Chapter 1*) that such as system would need to be designed with great care, to respect teachers' autonomy, and ensure that system recommendations are aligned with teachers' goals and not perceived as passing inappropriate judgment.

Automated support for dynamic, adaptive peer matching.

In line with findings from my early directed storytelling sessions (*Chapter 1*), teachers noted it would be useful to receive support in adaptively and dynamically assigning students to serve as peer tutors throughout a class session (cf. Diana et al., 2017; Olsen, 2017).

Trade-offs between accuracy and interpretability.

Although teachers had expressed a preference for simpler, more interpretable analytics in lower-fidelity prototyping sessions, it became apparent during higher-fidelity prototyping sessions that the strength of this preference may depend heavily on: (1) the underlying construct that a real-time indicator was purporting to measure; and (2) the kinds of teacher actions that this information might inform (cf. Lipton, 2016). For example, when it came to detection of "system misuse," it was important to teachers that they could easily understand (and thus justify to students) precisely the patterns of student actions that had led to this classification. By contrast, teachers appeared to be more open to the use of "black box" algorithms for detecting "unproductive persistence" if this meant alerting them to these students earlier (given that after this initial alert, teachers could apply their own discretion, using other information available to them).

5.5 In-lab evaluation of *Lumilo*'s Impacts on Teacher Behavior using Replay Enactments

Prior to piloting *Lumilo* in live K-12 classrooms, I wanted to better understand its effects on teachers' behavior. I ran an in-lab evaluation study, consisting of an additional six Replay Enactments sessions. Across these six sessions, *Lumilo*'s design was held constant to support investigation of whether and how the then-current version of *Lumilo* might influence teachers' time allocation (cf. Martinez-Maldonado, Clayphan, Yacef, & Kay, 2015) across students of varying prior domain knowledge and learning rates compared with business-as-usual (i.e., without an orchestration tool).

Each session replayed data from a 40-minute class session, randomly selected from a pool of five "average" and "remedial" classes. An "average" class was replayed in four Replay Enactments sessions, and a "remedial" class was replayed in the remaining two sessions. Advanced classes were omitted from the selection pool for this study, given that there was relatively little between-student variation in test scores in these classes. To minimize potential effects of student names or seating positions on teachers' behavior, replayed students were randomly assigned to names and seats in each session.

As discussed in *Chapter 4* (*Section 4.6*), in order to track how teachers allocated their time across students, I architected *Lumilo* so that the indicators positioned above students' heads doubled as mixed reality proximity sensors within a physical classroom space. Each teacher's allocation of time to a given student was measured as the cumulative time (in seconds) that they spent within a 4-foot radius of that student's indicator. If a teacher was within range of multiple students, time was accumulated only for the nearest of these students. I used hierarchical linear modeling (HLM) to predict teachers' time allocation across replayed "students" as a function of either students' prior domain knowledge (measured by a pretest in the original class session that was being replayed) or students' learning (measured by a posttest, controlling for the student's pretest score). As is the case in a typical classroom study, teachers participating in these sessions did not have access to pre- or post-test data (e.g., since a pre-test may not necessarily be administered when a tutoring system is used outside of a research study, and since in this context, post-test data comes from the "future"). Accordingly, *Lumilo* did not use any pre- or post-test data to generate the real-time analytics it presented to teachers. Using 2-level models with students nested within classrooms provided a better fit than 1-level or more complex models.

Standardized coefficients for student-level variables are shown in row 2 of Table 5-1. As shown, teachers using *Lumilo* in Replay Enactments sessions spent significantly more of their time attending to replay "students" who had relatively lower pretest scores, or lower posttest scores (controlling for pretest). By contrast, row 1 of Table 5-1 shows results from a prior in-vivo classroom study with four teachers (across seven live middle school classrooms), in which students worked with *Lynnette* while their teacher monitored and helped students (without access

to an orchestration tool). Performing the same analysis as above, this time with data from the classroom study (with time allocation recorded via manual classroom coding), I again found that 2-level models provided the best fit. Coefficients for these models are provided in Table 5-1 (row 1). Although all participating teachers reported attempting to devote most of their time to students whom they expected would struggle with the material, I found no significant relationships between students' pre- or post-test scores and teachers' time allocation across students.

Table 5-1. Relationships between teachers time allocation across replayed students (in seconds) and students' prior knowledge (pretest score) and learning (posttest score controlling for pretest).
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Class Type	Number of Teachers	Number of Classrooms	Average class size	Using *Lumilo*?	Pretest	Posttest | Pretest
Live	4	7	16	No (business-as-usual)	6.29	-5.49
Replay Enact-ment	6	3	15	Yes	-4.66*	-21.19**

I took this contrast as preliminary evidence that *Lumilo* may aid teachers in focusing on and helping those students with lower prior knowledge. More importantly, I interpreted these results as suggestive that *Lumilo* may successfully aid teachers in identifying students who would have gone on to exhibit the lowest *learning* in an actual classroom session, at least without the teacher's help.

Since the use of replay removes the possibility of a causal arrow from teacher behavior to students' learning within the software, Replay Enactments allow investigation into *counterfactuals* such as the above, for different forms of teacher augmentation (e.g., different algorithms and visualizations). On the other hand, classroom studies – although much costlier to run – enable investigation into the effects of a tool in the *social context* where it is ultimately intended to be used, in the presence of many competing influences on a teacher's attention and judgment.

5.6 Conclusions

Developing methods to engage non-technical stakeholders in shaping *algorithmic elements* of complex, data-driven AI systems remains a central open challenge for the UX design of data-driven AI systems (e.g., Baumer, 2017; Chen & Zhu, 2019; Dennerlein et al., 2018; Holstein et al., 2019a; Kyung Lee et al., 2018; Prieto-Alvarez, et al., 2018; Zhu & Terveen, 2018). This chapter has introduced Replay Enactments, a replay-based prototyping method that uses authentic data and (imperfect) algorithms to reveal important nuances that other methods – such as Wizard of Oz studies (Lovejoy, 2018; Odom, Zimmerman, Forlizzi, Dey, & Lee, 2012) – may be ill-suited to surface (see item 4 under *Summary of Expected Contributions – Novel design and prototyping methods)*.

Much like other recently proposed prototyping methods in the learning analytics (LA) literature, such as the simulation methods presented in Martinez-Maldonado et al. (2012) and Mavrikis et al. (2016), Replay Enactments involve replaying log data from students' interactions with educational technologies in order to prototype real-time analytics and visualizations with end-users (such as teachers or students). However, in the spirit of recent HCI methods for prototyping radically new experiences, such as User Enactments (Odom, Zimmerman, Davidoff, Forlizzi, Dey, & Lee, 2012), Replay Enactments builds on prior LA approaches by emphasizing embodied role-playing and task performance (Holstein, Hong, et al., 2018; Holstein et al., 2019a). Whereas methods like User Enactments typically involve Wizard-of-Oz'd scenarios, Replay Enactments prototype an experience using authentic data and algorithms, evolving over time. Although this requires earlier investment in technical development, doing so can enable earlier, detailed observations of the interplay between human and machine judgments, and the ways in which a system's false positives and false negatives may impact the experience of using a data-driven algorithmic system (cf. Dove, Halskov, Forlizzi, & Zimmerman, 2017; Holstein et al., 2019a).

I view Replay Enactments as one step towards developing and formalizing a broader class of prototyping methods that can address the unique challenges of designing and prototyping data-driven algorithmic systems. Moving forward, I expect that methods for prototyping with authentic (imperfect) algorithms and data collected from diverse data-generating contexts, will be invaluable in designing AI systems that are usable, useful, fair, and trustworthy. A promising direction for future work is to explore how replay-based prototyping methods like Replay Enactments might be refined to further structure participants' feedback. While the current version of Replay Enactments has participants engage in a relatively unstructured think-aloud while performing a task, future refinements might focus users' attention on specific aspects of a system's design—for example, to test the usefulness of particular forms of AI "explanations" in the context of specific user tasks and data-generating contexts (cf. Doshi-Velez & Kim, 2017; Poursabzi-Sangdeh, Goldstein, Hofman, Vaughan, & Wallach, 2018) or to support the discovery

of spurious and undesirable biases in a systems' behavior (cf. Holstein, Wortman Vaughan, et al., 2019). Similarly, instead of replaying a full class session, future work on replay-based prototyping methods like REs might explore methods to curate *specific scenarios* (i.e., data clips) that have properties desirable for answering particular kinds of questions a research/design team may have. At the same time, a potential tradeoff in developing more structured methods such as these is that this upfront structuring and curation may reduce opportunities for unexpected design findings to emerge (Odom et al., 2012). As such, conducting combinations of less- and more-heavily structured studies may be desirable.

In order to test the interacting dynamics of specific data-generating contexts, algorithms, visualizations, and human judgments and decisions in a simulated task context, Replay Enactments differs from related prototyping methods like experience prototyping and User Enactments by requiring fairly heavy upfront investment in technical development. To a certain extent, a greater degree of upfront technical investment may be unavoidable when prototyping complex, data-driven algorithmic systems (Dove et al., 2017). However, a promising direction for future work may be to explore the design of *lighter-weight prototyping methods* that can reap some of the relative benefits of Replay Enactments earlier on in the design process.

Similarly, a fruitful direction for future work may be to explore new methods that can provide earlier insight into *social nuances* before deploying a system in the real world—an aspect that was not deeply explored in the Replay Enactments studies presented in this chapter—while still keeping development and recruitment costs low. This may involve, for example, including multiple human participants in role-playing exercises, a mix of multiple live and replayed participants, or a mix of multiple live and simulated, interactive participants (cf. Harpstead, 2017; Maclellan, Harpstead, Patel, & Koedinger, 2016).

In the next chapter of this dissertation, *Chapter 6*, I build off of the analysis approach presented in *Section 5.5* to explore how Replay Enactments can support the iterative alignment of a real-time analytics tools with *educational goals*.

Part Three

Evaluating Real-time, Wearable Teacher Augmentation

Part Three of this dissertation focuses on the evaluation of real-time teacher augmentation in live classroom settings. In *Part Three*, I run iterative classroom pilots and an in-vivo classroom experiment with *Lumilo* with a total of 14 teachers, across 30 middle school math classrooms (see Table 3 for an overview).

In *Chapter 6*, I present Causal Alignment Analysis (CAA): a design framework for the iterative, data-informed design and evaluation of real-time teacher augmentation. I demonstrate CAA through a case study describing the iterative piloting and design refinement of *Lumilo* (introduced in *Chapter 4*). Whereas *Chapter 5* focused primarily on iterative experience prototyping, through Replay Enactments, to better understand teachers' *perceived* needs, *Chapter 6* focuses on further shaping *Lumilo*'s design in ways that are likely to benefit students' learning (building upon the evaluation approach presented in *Section 5.5*).

In *Chapter 7*, I then conduct an in-vivo classroom experiment with the resulting version of *Lumilo*, to evaluate the impacts of real-time teacher analytics and teacher–AI co-orchestration on classroom dynamics and student learning.

Finally, whereas *Chapters 6* and *7* focus primarily on quantitative research findings from classroom pilots and in-vivo classroom experiments, in *Chapter 8*, I share observations from piloting *Lumilo* "in the wild," with a focus on needs and nuances that were not captured in my earlier design research with teachers (*Parts One* and *Two*).

Table 3. Demographics for schools participating in live classroom pilots and in-vivo experiments.

School	Region	Free/Reduced Price Lunch	# of teachers	# of teachers with ≤ 2 years' experience	# of classrooms
C	Suburban	23%	4	1	8
E	Rural	34%	3	0	7
I	Urban	63%	2	0	5
M	Urban	60%	4	0	8
N	Suburban	11%	1	0	2

Chapter 6
Causal Alignment Analysis: A Framework for the Outcome-driven, Data-informed Design of Teacher Analytics Tools

This chapter is based in part on the following publications:

- Holstein, K., McLaren, B. M., & Aleven, V. (2018a). Informing the design of teacher awareness tools through Causal Alignment Analysis. In *Proceedings of the 13th International Conference of the Learning Sciences (ICLS 2018)* (pp. 104-111).

6.1 Background and Motivation

The design and development of real-time teacher analytics tools is often motivated by the assumption that enhanced teacher awareness will lead to improved teaching, and ultimately, to improved student outcomes. Yet there is a paucity of evidence to support these claims, and scientific knowledge about the effects that such tools may have on teaching and learning in real educational settings is scarce (Molenaar & Knoop-van Campen, 2017; Rodríguez-Triana et al., 2017). As such, it is a challenging problem to design effective teacher analytics tools. Designers must not only anticipate the effects analytics may have on teacher awareness, but also how this enhanced awareness might affect teacher decision-making and behavior, and how these changes in behavior will ultimately influence student learning. Compounding these challenges, while existing design workflows such as LATUX (Martinez-Maldonado et al., 2016) support the user-centered design of real-time analytics tools based on teacher feedback, there is a lack of standard methodology for the outcome-driven improvement of such tools, to achieve targeted educational goals. Furthermore, justifications for design decisions (e.g., what information to present in a dashboard) are rarely reported in the literature (Rodríguez-Triana et al., 2017).

Researchers in other areas of educational technology research have adopted data-informed, outcome-driven approaches to iteratively guide the design of technologies towards educational goals (e.g., Koedinger, Stamper, McLaughlin, & Nixon, 2013). For example, the design of ITSs often includes an iterative refinement process, in which historical student data is leveraged to increase alignment between the software's instructional design and the way students actually learn the material, as inferred from data (Aleven, McLaughlin, Glenn, & Koedinger, 2016; Liu & Koedinger, 2017). By contrast, teacher analytics tools are not typically optimized to guide teacher behavior in ways that are productive for learning. Given the complexity of designing teacher analytics tools, and the substantial causal distance between enhancing teacher awareness

and enhancing student learning (Xhakaj, et al., 2017), bringing such outcome-driven approaches to the design of teacher analytics tools may be key to ensuring their effectiveness.

In this chapter, I introduce Causal Alignment Analysis (CAA), a framework for the data-informed, outcome-driven design of teacher analytics tools which links the design of such technologies to educational goals (see item 6 under *Summary of Contributions – "Causal Alignment Analysis, a framework for the data-informed, iterative design of teacher augmentation"*). I illustrate CAA via a case study, demonstrating the iterative design refinement of *Lumilo* over a sequence of pilot studies. Finally, I discuss conclusions and highlight directions for future work.

6.2 Causal Alignment Analysis

Beginning from a specification of educational goals (e.g., improving student learning and/or engagement), CAA involves gradually aligning the design of a teacher analytics tool with these goals, by repeatedly evaluating the tool's effects along hypothesized causal paths from teacher tool use to targeted student-level outcomes. Specifically, CAA begins by generating answers to the questions below, which may represent open hypotheses where theory is absent or underspecified:

Q1. What student outcomes do we wish the tool to support?
Q2. What student-level processes promote or hinder progress toward the goals specified in an answer to question **(Q1)**?
Q3. What teacher-level processes promote or hinder the student-level processes identified in an answer to question **(Q2)**?
Q4. How can the tool be better designed with respect to the processes identified in answers to questions **(Q2)** and **(Q3)**?

Taken together, answers to these questions specify hypothesized causal paths from a teacher's use of a particular tool to enhanced student outcomes (as illustrated in Figure 6-1). Making the goals and hypothesized mechanisms of action of an analytics tool explicit early on may usefully constrain the design of an initial prototype. Once an initial prototype has been developed, CAA then involves prototyping the tool with teachers and students. Using data from these prototyping sessions, designers evaluate the alignment (or lack thereof) between the prototype's observed effects on teacher behavior, and one or more hypothesized causal paths to improved student outcomes. Based on this analysis, designers can refine the prototype with the goal of increasing alignment, increasing the chances that the tool will have a positive impact in the classroom. Finally, the prototyping cycle repeats, to evaluate the effectiveness of this realignment.

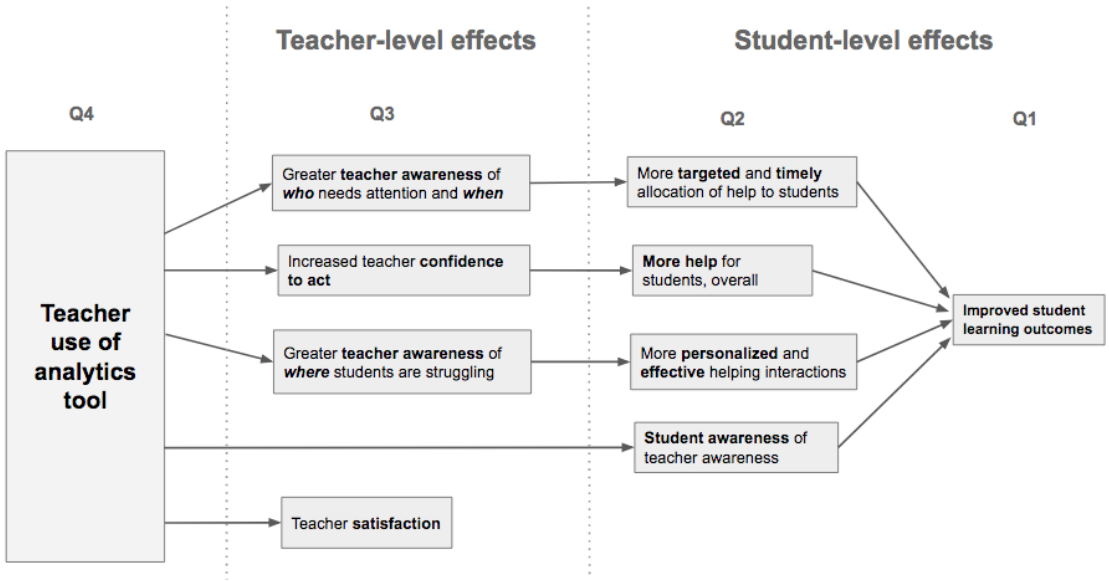

Figure 6-1. Examples of hypothesized causal paths, based on prior literature on real-time teacher analytics tools, leading from teacher use of an analytics tool to improved student learning outcomes. Causal tiers are labeled with questions from CAA (figure adapted from Holstein et al., 2018a).

Figure 6-1 shows examples of potential causal paths from a teacher's use of an analytics tool to teacher- and student-level outcomes. For these examples, I consider the context of self-paced classrooms in which students work with educational software, while a teacher uses a real-time analytics tool to decide when, with which students, and how to provide additional assistance. From left to right, the diagram shows potential influences of a teacher analytics tool (**Q4**) on the behavior of the teacher using it (**Q3**), potential impacts of resulting shifts in teacher behavior on students (**Q2**), and finally, potential impacts of these student-level effects on student learning outcomes (**Q1**).

Given that a teacher has limited time to provide one-on-one assistance, the top path in Figure 6-1 posits that if teachers were alerted to critical situations (e.g., a student exhibiting a common misconception), they would be able to more effectively allocate time to students who need their attention the most, at the right moments (e.g., see Holstein, Hong, et al., 2018; Holstein et al., 2017b; 2019a; Martinez-Maldonado et al., 2015; Tissenbaum et al., 2016). Thus, a teacher analytics tool should be designed to alert teachers of such critical situations. In contrast to the top path – which represents a hypothesis that students using educational software would learn more from additional teacher assistance in certain situations – the second path, represents the hypothesis that students would benefit from more teacher attention *in general*. Under this hypothesis, an analytics tool should be designed to encourage teachers to spend more time working with students, overall – perhaps by making teachers feel more informed, and thus increasing their overall "confidence to act" (van Leeuwen, Janssen, Erkens, & Brekelmans, 2015). The third causal path represents the hypothesis that, if the quality of a teacher's

one-one-one interactions with students were improved (e.g., more tailored to a student's specific weaknesses), this would enhance student learning with the software (see van de Pol & Elbers, 2013). Furthermore, this path posits that if teachers were made more aware of student difficulties, this would lead teachers to tailor their one-on-one interactions more closely to individual students' needs. The fourth causal path posits a direct link from a teacher's use of an awareness tool and a student-level effect. Under this hypothesis, students' mere awareness that a teacher is monitoring their activities in the software contributes to their learning, perhaps by increasing engagement (see *Chapter 2* and Stang & Roll, 2014). Finally, the bottom path represents a hypothesis that teachers' use of a particular analytics tool positively impacts their classroom experience (Rodriguez-Triana et al., 2017), but has no notable effects on student outcomes.

Despite showing a relatively small set of hypothesized paths – each specified at a high level of abstraction – Figure 6-1 illustrates the enormous breadth of the design space for teacher analytics tools. Focusing on different combinations of these paths may yield radically different tool designs.

In addition to helping guide and scope the *initial design* of a teacher analytics tool, CAA can be used to inform the refinement of an existing tool. A designer applying CAA to the refinement of an existing teacher analytics tool would begin by considering the tool's educational goals, and would then work backwards from these goals (cf. Wiggins et al., 2001) to construct one or more hypothesized causal paths leading from a teacher's use of the tool to the achievement of these goals (guided by existing data and theory where possible). By iteratively prototyping the teacher analytics tool and collecting data on relevant outcomes, the designer would evaluate whether the tool is likely to have desirable effects along each node in the path, adjusting the design as needed. To illustrate the use of CAA in practice, I next demonstrate the iterative design refinement of *Lumilo* (introduced in *Chapter 4*).

6.3 Case Study: Iterative design refinement of *Lumilo*, using Causal Alignment Analysis

In developing the initial versions of *Lumilo* (see *Chapters 4* and *5*), I decided to target its design largely towards the problem of supporting teachers in allocating their scarce time and attention to those students who need it the most (the top path in Figure 6-1), during class sessions in which students work individually with ITSs. This focus was motivated, in part, by design research with teachers, which highlighted these decisions as a major challenge in orchestrating personalized learning (e.g., see Martinez-Maldonado, et al., 2015, and *Chapters 1*, *2*, and *4* of this dissertation). In addition, this choice of focus was motivated by prior empirical results, suggesting that teachers' decisions about whom to help, and when, may be impactful (e.g., see Martinez-Maldonado et al., 2015 and *Chapter 2* of this dissertation). As I narrowed my focus

towards the development of a concrete prototype (*Chapters 4* and *5*), I increasingly focused on designing for classrooms that use *Lynnette*, a specific ITS that provides tutored problem solving practice in equation solving (Long & Aleven, 2013; 2017; Long, Holstein, & Aleven, 2018).

Taken together, the indicators presented by *Lumilo* can be taken to represent, at least in part, the phenomena that teachers expect require their attention and/or intervention. For example, four of the teachers I worked with while designing and prototyping *Lumilo* (*Chapters 1*, *4*, and *5*) argued that alerts about high local error (one of three states of *Lumilo*'s "Struggling" indicator) would require immediate intervention. Without rapid intervention, these teachers worried that repeated error-making in an ITS might "entrench" the errors, despite negative feedback from the software (see Metcalfe, 2017 for a discussion of this common teacher belief). At the same time, it is important to note that teachers also found some indicators valuable for reasons that did not directly relate to helping their students. For example, I found that positive indicators about student performance were valuable to teachers, in part, because they found them personally motivating (Holstein, Hong, et al., 2018; Holstein et al., 2019a).

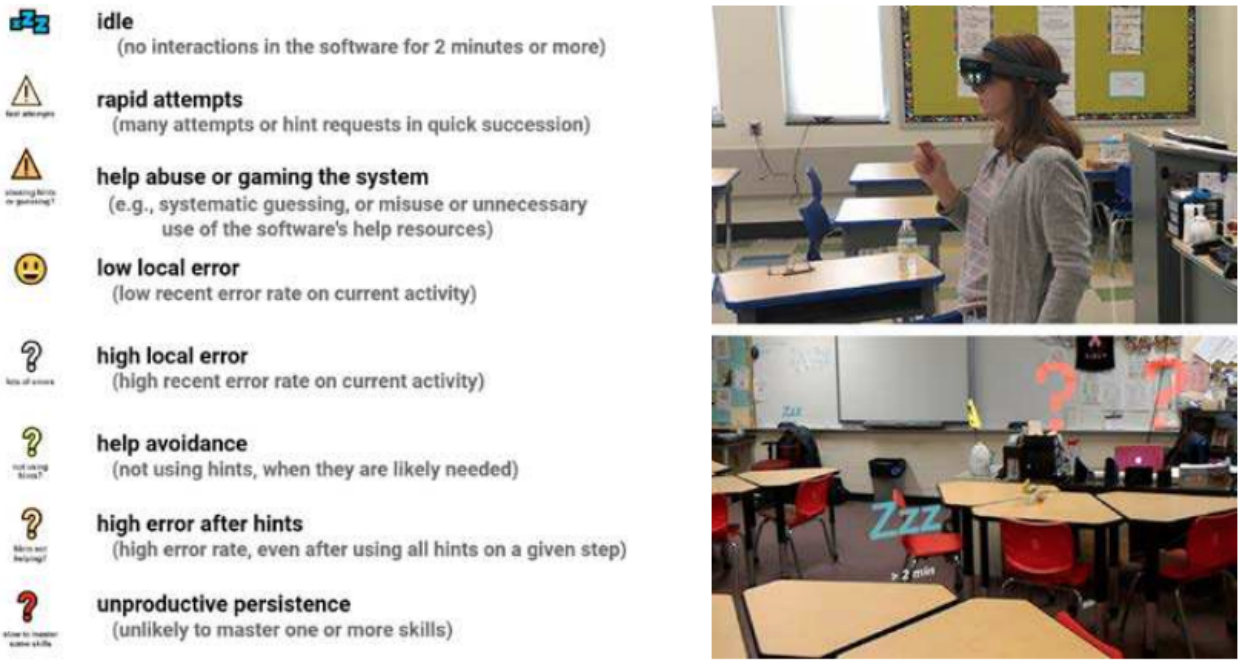

Figure 6-2. Left: Full set of student-level indicator states displayed by an early version of *Lumilo*. Top-right: Teacher using *Lumilo*. Bottom-right: Point-of-view screenshot (taken moments after the end of a class session, to preserve student privacy).

In addition to serving teachers' expressed needs and desires, however, I wanted to design a teacher analytics tool that could *measurably benefit students*. Teachers' intuitions about the most important opportunities for intervention may not always be correct (e.g., Baker, Walonoski, Heffernan, Roll, Corbett, & Koedinger, 2008; Metcalfe, 2017). Therefore, in the next phase of

design, I used Causal Alignment Analysis to iteratively refine *Lumilo*'s design, to increase its chances of having a positive impact in the classroom. With respect to the first of CAA's four guiding questions, I had defined the primary learning objectives as the set of equation-solving skills that a given version of *Lynnette* tutors (see Long, Holstein, & Aleven, 2018). In answer to CAA's second and third questions, I adopted a causal model search approach to understand the relationships between *Lumilo's* indicators and student learning outcomes on a pre- and post-test. I hypothesized that a teacher's attention during class should be directed towards student processes that have a negative influence on learning. Finally, in response to CAA's fourth question, I iteratively refined *Lumilo* to direct teachers' time and attention towards these processes, over a sequence of in-lab and classroom pilot studies. Each of these steps is discussed in turn below.

To answer CAA's second question (*"What student processes promote or hinder ..."*), I sought to better understand the relationships between student processes detected by the current prototype of *Lumilo* (the student-level indicators shown in Figure 6-2, emerging from the design process described in *Chapters 1*, *4*, and *5*) and student learning within *Lynnette*. To this end, I adopted a causal model search approach, using directed acyclic graphs (DAGs) to represent the causal structure among variables measured by *Lumilo*, and student assessment scores. I collected data from 115 middle school math students (across 7 classrooms and 4 teachers), each of whom worked with *Lynnette* for 60 minutes. In these classrooms, the teacher did not use a real-time analytics tool (Table 6-1, Study 1). In all studies, students' equation-solving skills were assessed via a pre- and post-test, administered before and after using *Lynnette*. I used two forms that were identical except for the specific numbers used in equations, and presented the forms in counterbalanced order across pre- and posttest.

I then used the PC algorithm in the Tetrad V program to search for an equivalence class of DAGs, consistent with a set of conditional independence constraints (Spirtes et al., 2000). The PC algorithm is asymptotically reliable; its primary limitations are its assumptions that no unmeasured confounders are present, and that any underlying causal relationships between variables can be modeled by linear functions. To relax the former of these assumptions, I also used the FCI algorithm, which allows for the possibility of unmeasured confounders. The FCI algorithm learns an equivalence class, represented by partial ancestral graphs (PAGs), encoding uncertainty over the nature of pairwise relationships between variables (Spirtes et al., 2000). To inform both searches, I provided background knowledge about the study design as a search constraint: I specified that the pretest was prior to any process variables, and that all process variables preceded the posttest.

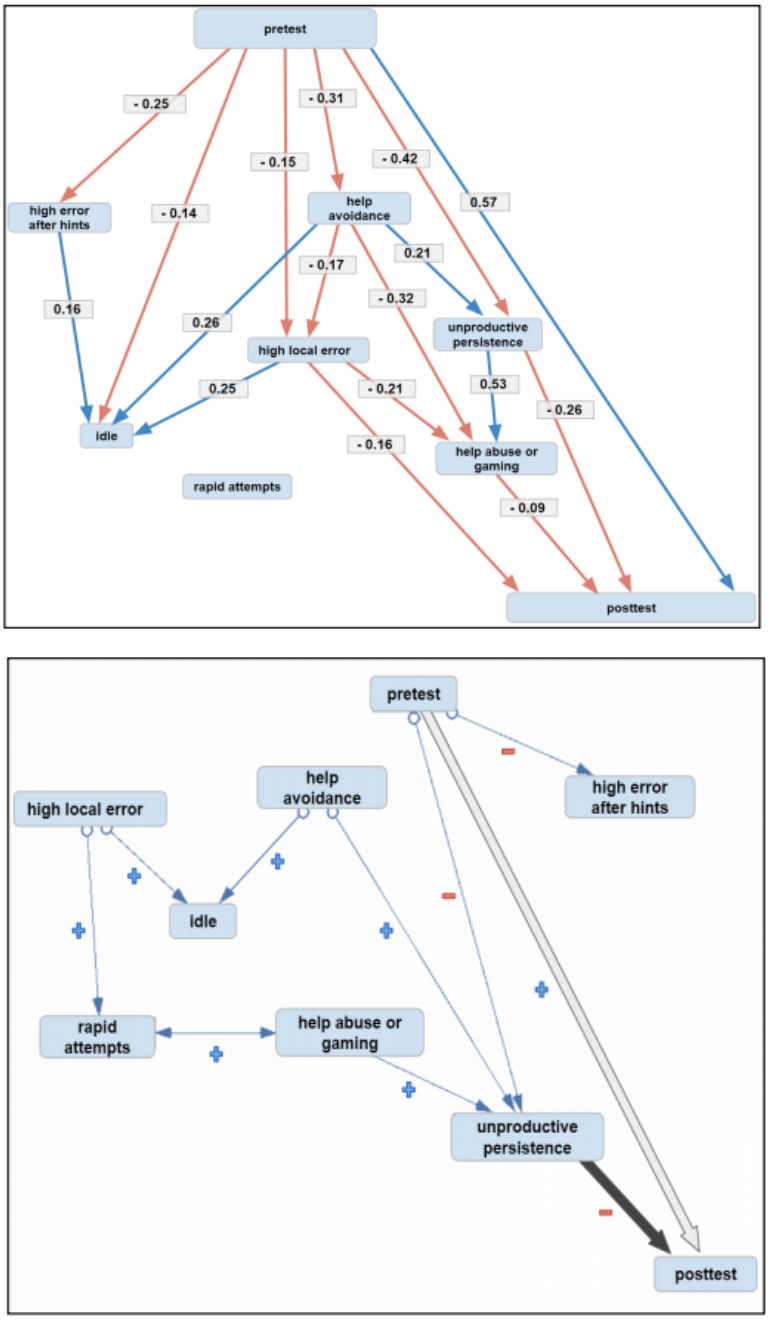

Figure 6-3. Top: model found by PC, with normalized coefficient estimates included. Bottom: PAG equivalence class found by FCI, encoding the possibility of unmeasured common causes.

Figure 6-3 (left) shows the DAG learned with the PC algorithm, including normalized coefficient estimates, to enable comparison of magnitudes. This model suggests that, of the indicators included in the initial prototype of *Lumilo*, three are potential direct causes of reduced student learning within the software: help abuse (measured by the Help Model introduced in Aleven et al., 2006; 2016) or gaming-the-system (measured by the gaming detector introduced in Baker et

al., 2008), high local error (defined by teachers during Replay Enactments prototyping (*Chapter 5*) as an error rate greater than 80%, within the last 8 student actions on the current activity), and unproductive persistence (measured by "wheel-spinning," as described in Beck & Gong, 2013; Kai et al., 2018; and Zhang et al., 2019). This model fits the data well[17] ($\chi^2 = 18.33$, df = 19, p = .50) (1). Figure 6-3 (right) shows the PAG learned with the FCI algorithm. In this figure, bidirectional links indicate the presence (and circle-origin links indicate the possibility) of unmeasured confounders. Otherwise, links indicate causal relationships. Wide links indicate no unmeasured confounders, and dark, wide links further indicate direct relationships. The PAG equivalence class found by FCI suggests that unmeasured confounders could potentially explain several of the links between *Lumilo*'s indicators. Finally, in both causal models, gaming/help-abuse, unproductive persistence, and help avoidance (as measured by the Help Model, Aleven et al., 2006; 2016) are negatively linked to student learning. The model found by FCI suggests that out of 7 negative teacher-generated indicator ideas implemented in *Lumilo*, only one is directly linked to student learning: unproductive persistence. Influences of help avoidance and gaming/help-abuse on learning may in turn be mediated through unproductive persistence.

To determine how the design of *Lumilo* might be improved (the fourth question in CAA), I first wanted to better understand how the then-current prototype of *Lumilo*[18] influenced teacher behavior, prior to deploying it in real classrooms. To this end, I conducted finer-grained analyses of data from the Replay Enactments (REs) sessions reported in *Chapter 5*.

First, I investigated how teachers' time allocation across students during REs may have been influenced by each of *Lumilo*'s student-level indicators. Teacher time allocation was measured per student by the cumulative time (in seconds) spent within a 4-ft. radius of that student (resolving ties among students by proximity, as described in *Chapter 5*), as well as time spent monitoring the student's activities via *Lumilo*'s deep-dive screens (see *Chapter 4*). Table 6-1 (Study 2) shows group-normalized correlations between detected student processes and teachers' time allocation during six REs. Real-time indicators that were not significant predictors of teacher time allocation are omitted. As shown, occurrences of four of *Lumilo*'s indicator alerts were significantly positively correlated with teachers' time allocation.

Second, to understand the degree to which *Lumilo* might have directed teachers towards students most in need of help, as per the top path in Figure 6-1, correlations between student assessment scores and teacher time allocation are also shown in Table 6-1. Given that teachers did not have

[17] In path analysis, the null hypothesis is that the estimated model is the true model. The p-value represents the probability, under the null, of observing a difference between the estimated and observed covariance matrices at least as large as the realized difference; a p-value above a given threshold (conventionally alpha = .05) implies a model cannot be rejected.

[18] For simplicity, although the prototype of *Lumilo* underwent many tens of design iterations *prior to* the activities discussed in this chapter (see *Chapters 4* and *5*), versions of *Lumilo* discussed in this chapter will henceforth be referred to as version 1, version 2, and so on.

access to assessment scores during REs, and that it is not possible to influence learning during a replayed class, I take the correlation between teacher time allocation during REs and student post-test scores (controlling for pretest) as evidence that *Lumilo* can direct teachers' time to students who would otherwise exhibit lower learning. However, this correlation was relatively small, suggesting room for improvement.

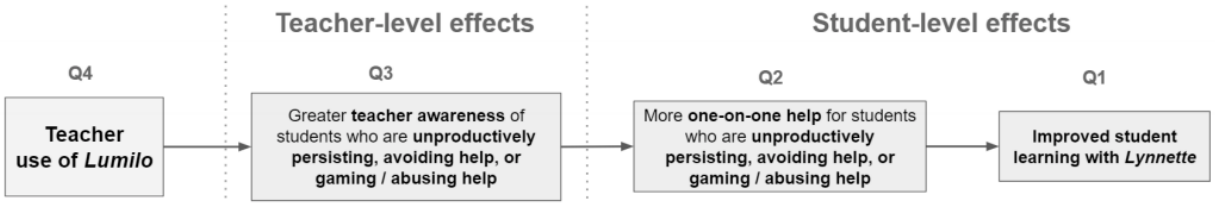

Figure 6-4. Hypothesized causal path from a teacher's use of *Lumilo* to improved student learning outcomes.

Table 6-1. Correlations between teacher time allocation, and detected student processes and test scores, * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$. Rows show a series of studies, using successive versions of the *Lumilo* prototype.

	Study Context				Process Variables (awareness tool alerts)					Assessment Scores	
Study	type	awareness support	sample (teachers, classes, students)	total time (min)	unproductive persistence	help avoidance	help abuse or gaming	rapid attempts	high local error	pretest	post | pre
1	live	none	(4, 7, 115)	60	- 0.06	**- 0.17***	- 0.09	- 0.12	0.06	0.13	- 0.02
2	RE	*Lumilo v1*	(6, 3, 90)	40	**0.25***	- 0.03	**0.44*****	**0.38*****	**0.32****	**- 0.06****	**- 0.17****
3	live	*Lumilo v2*	(1, 1, 15)	40	**0.65***	**0.61***	0.22	0.27	0.39	**- 0.84*****	0.40
4	live	*Lumilo v3*	(2, 4, 84)	60	**0.52*****	**0.16***	**0.07***	0.01	0.18	**- 0.30***	0.16

Taken together, these analyses suggested various ways the design of *Lumilo* could be improved (**Q4**), to increase its alignment with the hypothesized causal path shown in Figure 6-4. Unproductive persistence was the weakest driver of teacher attention during REs, out of the indicators correlated with teacher time allocation (as shown in Study 2 of Table 6-1), despite being the one variable directly (and negatively) related to student learning in the causal model found by FCI. To better align *Lumilo*'s design with the findings of these analyses, the design should focus more explicitly on alerting teachers to cases of unproductive persistence, by increasing the salience of this alert and others that may serve as reliable early predictors. For instance, although help avoidance is a potential cause of unproductive persistence in the PAG found by FCI (and thus potentially valuable as an early predictor), it was not a significant driver

of teacher attention. Similarly, this model suggests that less emphasis should be placed on alerting teachers to high local error or rapid attempts *in general*, and more should be placed on alerting teachers to cases that specifically constitute maladaptive help-use and/or gaming. As such, I next refined the prototype of *Lumilo* to place greater emphasis on alerts about unproductive persistence and persistent help avoidance. This included not only making the corresponding indicator symbols more visually salient than others (larger and brighter), but also drawing teachers' attention to these alerts through ambient sound notifications (see *Chapters 4* and *5*). Meanwhile, I de-emphasized other alerts, including high local error and rapid attempts, by making these indicators relatively dimmer and smaller. Furthermore, if a student was detected as unproductively persisting on one or more skills, avoiding help, or gaming/abusing-help, any other alerts for that student would be hidden at a glance (although still accessible upon a teacher's request).

I next ran two more pilot studies, in live classrooms. The first of these studies was run with one teacher in a single, 80-minute class session. In this study, students worked with *Lynnette* for 40 minutes, while the teacher used *Lumilo* (version 2) to monitor and help students. Students' domain knowledge in equation solving was measured before and after using the software, via computer-based pre- and post-tests, as in prior studies. As shown in Study 3 of Table 6-1, students who were more frequently detected as unproductively persisting or avoiding help received significantly more teacher time during this single-classroom pilot, compared with students exhibiting other behaviors tracked by *Lumilo*, suggesting that the design refinements may have had the intended effect. Furthermore, the teacher's attention during this single-classroom pilot was strongly and significantly focused towards students with lower prior domain knowledge (as measured by the pretest), and the correlation between teacher time allocation and student posttest score (controlling for pretest) was positive, despite a likely selection effect, although not statistically significant.

Following this pilot, I made minimal design refinements to Lumilo, in an effort to ensure that alerts of unproductive persistence were emphasized (as potentially more critical) over alerts of help avoidance and gaming/help-abuse. In version 3 of *Lumilo*, if a student was detected as unproductively persisting in the software on one or more skills, any other alerts for that student would be hidden.

I ran additional classroom pilots using *Lumilo* (version 3) in 4 classrooms. Students in each classroom worked with *Lynnette* for a total of 60 minutes while the teacher used *Lumilo* to monitor and help their students. As before, student domain knowledge was measured via 20-minute, computer-based pre- and post-tests. As shown in Study 4 of Table 6-1, unproductive persistence was the strongest predictor of teacher time allocation, followed by help avoidance and gaming/help-abuse. Classroom observations indicate that teachers continued to make use of all indicators presented by *Lumilo* (e.g., praising recent high performers or nudging inactive students), but tended to reserve in-depth help sessions for those students detected as

unproductively persisting. Retrospective post-interviews corroborated this observation. However, teachers also reported frequently attending to *"quick fix"* alerts for students who were physically *"en-route"* to a particular student the teacher was targeting for remediation (for further discussion of qualitative findings from classroom studies with *Lumilo*, see *Chapter 8*).

In summary, in the first phases of the design and prototyping process (*Chapters 4* and *5*), I decided to focus on the problem of supporting teachers in allocating scarce time and attention to those students who may need it most. I adopted a participatory design approach, eliciting ideas for real-time analytics that teachers considered actionable, relevant to learning, or otherwise valuable to monitor. I leveraged pre-existing student modeling techniques to provide teachers with these analytics, while iteratively prototyping and refining them with teachers to ensure their usefulness and usability. In the next phase of design, I used CAA to iteratively align *Lumilo*'s design with a hypothesized causal path (based on findings from causal modeling on student process data) from teacher tool use to improved student learning outcomes (a finer-grained instantiation of the top path in Figure 6-1, as shown in Figure 6-4). With respect to the first of CAA's four guiding questions (**Q1**), I defined students' learning objectives as the skills that *Lynnette* is intended to tutor, and assessed student learning with respect to these skills. In answer to CAA's second and third questions (**Q2** and **Q3**), I adopted a causal model search approach to discover a critical subset of *Lumilo*'s indicators, representing student processes that appear to most strongly influence learning outcomes with *Lynnette*. In turn, I hypothesized that students exhibiting these processes may benefit most from out-of-software, teacher interventions. Finally, with respect to CAA's fourth question, I iteratively refined *Lumilo*'s design – over a sequence of four pilot studies conducted in both simulated and live classrooms – to draw teachers' time and attention towards these students.

6.4 Conclusions

In this chapter, I have introduced Causal Alignment Analysis (CAA): a design framework for the data-informed, outcome-driven design and iterative improvement of teacher analytics tools, linking the design of these tools to educational goals (see item 6 under *Summary of Expected Contributions – "Causal Alignment Analysis, a framework for the data-informed, iterative design of teacher augmentation"*). I have illustrated the application and usefulness of CAA through a case study, demonstrating the iterative design alignment of a real-time teacher analytics tool (*Lumilo)* with a hypothesized causal path from teacher tool use to student learning (Figure 6-4). The resulting prototype augments teachers' awareness of student learning, metacognition, and behavior, while also measurably directing their time towards a subset of student processes that appear to have a negative influence on student learning outcomes.

While this case study may represent a step towards the design of teacher awareness tools that can measurably enhance student learning, it does not fully "close the loop" (Koedinger et al., 2013).

To support iterative design, a CAA approach favors running larger numbers of small to mid-scale studies over running a single high-powered study. As such, it may not support strong causal inference. To better understand whether and how a teacher's use of *Lumilo* influences student learning, I have also conducted a larger-scale in-vivo classroom experiment (presented in *Chapter 7)*. Analysis of data from this experiment will enable the investigation of multiple possible paths from teacher tool use to student learning (Figure 6-1), and will make it possible to begin teasing apart the distinct causal explanations that these paths represent. For example, although the analyses presented in this chapter led to the improvement of *Lumilo* with respect to the hypothesized causal path pictured in Figure 6-4, the results presented in this chapter leave open whether the final link in this path (improved student learning) will hold in practice.

While the case study presented in this paper focused on data-informed design optimization with respect to teacher attention allocation across students (the top path in Figure 6-1), there are many other causal paths along which an analytics tool might be optimized. For instance, even if teachers are made more aware of critical moments, it may not always be clear how to effectively respond. My prior design work with teachers suggests that they often desire more direct support (e.g., action recommendations) for planning and enacting effective interventions – especially in personalized learning contexts, where planning time can be very scarce (see *Chapters 1, 4,* and *5*). A promising direction for future work may be to use CAA to explore whether and how a teacher analytics tool could be designed to measurably enhance the effectiveness of teacher-student coaching interactions (e.g., through targeted teacher recommendations for *how to help* a particular student or students at a given moment).

In summary, as the fast-growing research area of teacher analytics tools matures, I hope to see the design of these tools (within and beyond the academic Learning Sciences and Learning Analytics communities) increasingly guided by educational data and theory, in addition to user feedback. Causal Alignment Analysis provides a framework for making the goals and implicit assumptions behind the design of awareness tools explicit – in turn representing these assumptions as hypotheses to be continuously tested throughout a design process. Given the complexity of designing teacher analytics tools, I expect that such data-informed design approaches will be key to ensuring that they are not only useful and usable, but also beneficial for learning.

Chapter 7
A Classroom Experiment to Investigate Student Learning Benefits of Teacher–AI Co-orchestration

This chapter is based in part on the following publications:

- Holstein, K., McLaren, B. M., & Aleven, V. (2018). Student learning benefits of a mixed-reality teacher awareness tool in AI-enhanced classrooms. In *Proceedings of the 19th International Conference on Artificial Intelligence in Education (AIED 2018)*. LNAI 10947 (pp. 154-168). Springer: Berlin.

7.1 Background and Motivation

As discussed in *Chapter 6*, the design and development of real-time teacher analytics tools is often motivated by the assumption that enhanced teacher awareness will lead to improved teaching, and ultimately, to improved student outcomes. Yet there is a paucity of evidence to support these claims, and scientific knowledge about the effects that such tools may have on teaching and learning in real educational settings is scarce (Molenaar & Knoop-van Campen, 2017; Rodríguez-Triana et al., 2017).

To investigate the hypothesis that real-time teacher–AI co-orchestration, supported by analytics from an ITS would enhance student learning, I ran an in-vivo classroom experiment with 343 middle school students (286 included in analyses, as discussed in the following sections), across 18 classrooms and 8 teachers. Among several other interesting findings, the results indicated that a teacher's use of *Lumilo* had a positive impact on student learning with an ITS, compared with both business-as-usual (where the teacher did not have any real-time support) and a second, stronger control condition in which the teacher had access to a simpler form of classroom monitoring support via mixed-reality glasses (i.e., support for peeking at a student's screen remotely), but without any advanced analytics.

Although much prior work has focused on the design, development, and evaluation of teacher analytics tools, very few studies have evaluated effects on student learning (Kelly et al., 2013; Molenaar & Knoop-van Campen, 2017; Rodríguez-Triana, et al., 2017; Xhakaj et al., 2017). Prior work has found that providing teachers with real-time notifications about student performance can direct their attention to low-performing students, resulting in local performance improvements (e.g., Martinez-Maldonado et al., 2015). Other recent work has begun systematically investigating how teachers use real-time progress and performance analytics in blended classrooms (e.g., Molenaar & Knoop-van Campen, 2017). However, the present work is

the first experimental study showing that real-time teacher analytics or teacher–AI co-orchestration can enhance students' learning outcomes (within or outside the context of AI-supported classrooms; see items 1 and 3 under *Summary of Contributions – "First broad design exploration of needs for real-time teacher analytics and orchestration support"* and *"First experimental study to demonstrate student learning benefits of real-time teacher analytics"*).

Interestingly, part of *Lumilo*'s overall effect on student learning appeared to be attributable to monitoring support alone. These findings suggest that, when evaluating the impacts of teacher-facing learning analytics tools, future research should take care to tease apart potential effects of a teacher's use of a monitoring tool (such as novelty effects or students' awareness of being monitored by their teacher), versus teachers' use of the kinds of advanced analytics and student modeling methods that are often the focus of research in learning analytics (LA), AI in education (AIED), user modeling (UM), and educational data mining (EDM) (see item 5 under *Summary of Contributions – "First investigations of relationships between teacher movement/monitoring and student behavior and learning in AI-supported classrooms"*).

7.2 Methods

In this study, I investigated the hypothesis that real-time teacher/AI co-orchestration, supported by real-time analytics from an ITS, would enhance student learning compared with both (a) business-as-usual for an ITS classroom, and (b) classroom monitoring support without advanced analytics (a stronger control than (a), as described below).

To test these hypotheses, I conducted a 3-condition experiment with 343 middle school students, across 18 classrooms, 8 teachers, and 4 public schools (each from a different school district) in a large U.S. city and surrounding areas. All participating teachers had at least 5 years of experience teaching middle school mathematics and had previously used an ITS in their classroom. The study was conducted during the first half of the students' school year, and none of the classes participating in this study had previously covered equation-solving topics beyond simple one-step linear equations (e.g., $x - 2 = 1$).

Classrooms were randomly assigned to one of three conditions, stratified by teacher. In the Glasses+Analytics condition, teachers used the full version of *Lumilo*, including all displays described above. In the business-as-usual (noGlasses) condition, teachers did not wear *Lumilo* during class, and thus did not have access to real-time analytics. I also included a third condition (Glasses) in which teachers used a reduced version of *Lumilo* with only its monitoring functionality (i.e., without any of its advanced analytics). This condition was included because prior empirical findings suggest that students' mere awareness that a teacher is monitoring their activities within an ITS may have a significant effect on student learning (e.g., by discouraging, and thus decreasing the frequency of maladaptive learning behaviors such as gaming-the-system)

(see *Chapter 2*; Holstein et al., 2017a; Stang & Roll, 2014). In the Glasses condition, teachers only retained the ability to "peek" at students' screens from any location in the classroom, using the glasses (although without the line-by-line annotations present in *Lumilo*'s "Current Problem" screen). All of *Lumilo*'s student indicators were replaced by a single, static symbol (a faint circular outline) that did not convey any information about the student's state. Further, the "Areas of Struggle" deep dive screens and the class-level displays were hidden. The goal of providing this stripped-down version of *Lumilo* (as opposed to a completely non-functional pair of glasses) was to encourage teachers to interact with the glasses, thereby minimizing differences in students' perceptions between the Glasses+Analytics and Glasses conditions. The Glasses condition bears some similarity to standard classroom monitoring tools[19], which enable teachers to peek at student screens on their own desktop or tablet display.

All teachers participated in a brief training session before the start of the study. Teachers were first familiarized with *Lynnette*, the tutoring software that students would use during the study. In the Glasses+Analytics and Glasses conditions, each teacher also participated in a brief (30-minute) training with *Lumilo* before the start of the study. In this training, teachers practiced interacting with two versions of the glasses (Glasses and Glasses+Analytics) in a simulated classroom context. At the end of this training, teachers were informed that, for each of their classes, they would be assigned to use one or the other of these two designs.

Classrooms in each of the three conditions followed the same procedure. In each class, students first received a brief introduction to *Lynnette* from their teacher. Students then worked on a computer-based pre-test for approximately 20 minutes, during which time the teacher provided no assistance. Following the pretest, students worked with the tutor for a total of 60 minutes, spread across two class sessions. In all conditions, teachers were encouraged to help their students as needed, while they worked with the tutor. Finally, students took a 20-minute computer-based post-test, again without any assistance from the teacher. The pre- and posttests focused on procedural knowledge of equation solving. Two isomorphic test forms were used for the pre- and post-test, which differed only by the specific numbers used in equations. The tests forms were assigned in counterbalanced order across pre- and post-test, and were graded automatically, with partial credit assigned for intermediate steps in a student's solution, according to *Lynnette*'s cognitive model.

In the Glasses and Glasses+Analytics conditions, I used Lumilo to automatically track a teacher's physical position within the classroom relative to each student, moment-by-moment (leveraging Lumilo's indicators as mixed-reality proximity sensors as discussed in *Chapters 4, 5,* and *6*). Given my prior observations, in Replay Enactments (*Chapter 5*) and classroom pioting (*Chapter 6*) that teachers in both of these conditions frequently provided assistance remotely

[19] For example: Chromebook Management Software for Schools, *https://www.goguardian.com/*; Hapara, *https://hapara.com/*; and LanSchool Classroom Management Software, *https://www.lenovosoftware.com/lanschool*

(i.e., conversing with a student from across the room, while monitoring her/his activity using the glasses), teacher time was also accumulated for the duration a teacher spent peeking at a student's screen via the glasses. In the noGlasses condition, since teachers did not wear *Lumilo*, time allocation was recorded via live classroom coding (using the LookWhosTalking tool[20]) of the target (student) and duration (in seconds) of each teacher visit. In addition to test scores and data on teacher time allocation, I analyzed tutor log data to investigate potential effects of condition on students' within-software behaviors.

7.3 Results

Fifty-seven students were absent for one or more days of the study and were excluded from further analyses. I analyzed the data for the remaining 286 students. Given that the sample was nested in 18 classes, 8 teachers, and 4 schools, and that the experimental intervention was applied at the class level, I used hierarchical linear modeling (HLM) to analyze student learning outcomes. 3-level models had the best fit, with students (level 1) nested in classes (level 2), and classes nested in teachers (level 3). I used class track (low, average, or high) as a level-2 covariate. Both 2-level models, (with students nested in classes) and 4-level models (with teachers nested in schools) had worse fits according to both AIC and BIC, and 4-level models indicated little variance on the school level. In the following subsections, I report r for effect size. An effect size r above 0.10 is conventionally considered small, 0.3 medium, and 0.5 large (Cohen, 1992).

Effects on Student Learning.

To compare student learning outcomes across experimental conditions, I used HLMs with test score as the dependent variable, and both test type (pretest/posttest, with pretest as the baseline value) and experimental condition as independent variables (fixed effects). For each fixed effect, I included a term for each comparison between the baseline and other levels of the variable. For comparisons between the Glasses+Analytics and noGlasses conditions, noGlasses was used as the condition baseline. Otherwise, Glasses was used as the baseline.

Across conditions, there was a significant gain between student pre- and post-test scores ($t(283) = 7.673$, $p = 2.74*10^{-13}$, $r = 0.26$, 95% CI [0.19, 0.34]), consistent with results from prior classroom studies using *Lynnette (*Long & Aleven, 2013; Long, Holstein, & Aleven, 2018; Waalkens et al., 2013*),* which showed learning gain effect size estimates ranging from r = 0.25 to r = 0.64. Figure 7-1 shows pre-post learning gains for each condition. There was a significant positive interaction between student pre/posttest and the noGlasses/Glasses+Analytics conditions ($t(283) = 5.897$, $p = 1.05*10^{-8}$, $r = 0.21$, 95% CI [0.13, 0.28]), supporting the hypothesis that

[20] *https://bitbucket.org/dadamson/lookwhostalking/*

real-time teacher/AI co-orchestration, supported by analytics from an ITS, would enhance student learning compared with business-as-usual for ITS classrooms.

Decomposing this effect, there was a significant positive interaction between student pre/posttest and the noGlasses/Glasses conditions ($t(283) = 3.386$, $p = 8.08*10^{-4}$, $r = 0.13$, 95% CI [0.02, 0.23]), with a higher learning gain slope in the Glasses condition, indicating that relatively minimal classroom monitoring support, even without advanced analytics, can positively impact learning. In addition, there was a significant positive interaction between student pre/posttest and the Glasses/Glasses+Analytics conditions ($t(283) = 2.229$, $p = 0.027$, $r = 0.11$, 95% CI [0.02, 0.20]), with a higher slope in the Glasses+Analytics condition than in the Glasses condition, supporting the hypothesis that real-time teacher analytics would enhance student learning, above and beyond any effects of monitoring support alone (i.e., without advanced analytics).

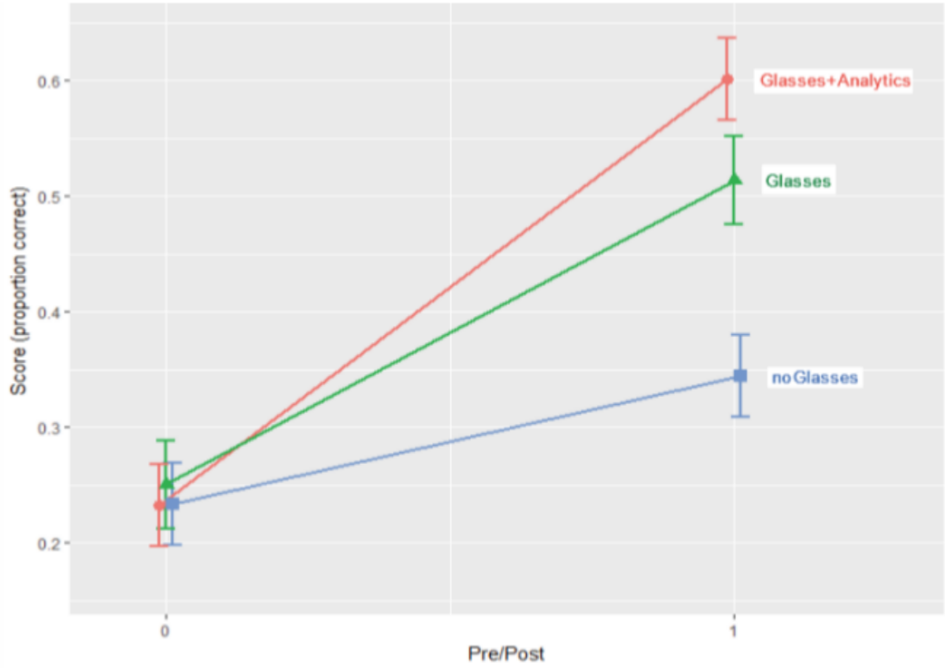

Figure 7-1. Student pre/post learning gains, by experimental conditions ("Glasses + Analytics": Teacher uses *Lumilo*; "Glasses": Teacher wears reduced version of *Lumilo*, without analytics; "noGlasses": Teacher does not wear glasses at all). Error bars indicate standard error (figure from Holstein et al., 2018b).

Aptitude-Treatment Interactions on Student Learning.

I next investigated how the effects of each condition might vary based on students' prior domain knowledge. As discussed in *Chapters 4*, *5*, and *6*, *Lumilo* was designed to help teachers quickly identify students who are currently struggling (unproductively) with the ITS, so that they could provide these students with additional, on-the-spot support. If *Lumilo* was successful in this

regard, we would expect to see an aptitude-treatment interaction, such that students coming in with lower prior domain knowledge (who are more likely to struggle) would learn more when teachers had access to *Lumilo*'s real-time analytics (see *Chapter 5* and *6*; Holstein, Hong, et al., 2018; Holstein et al., 2018a).

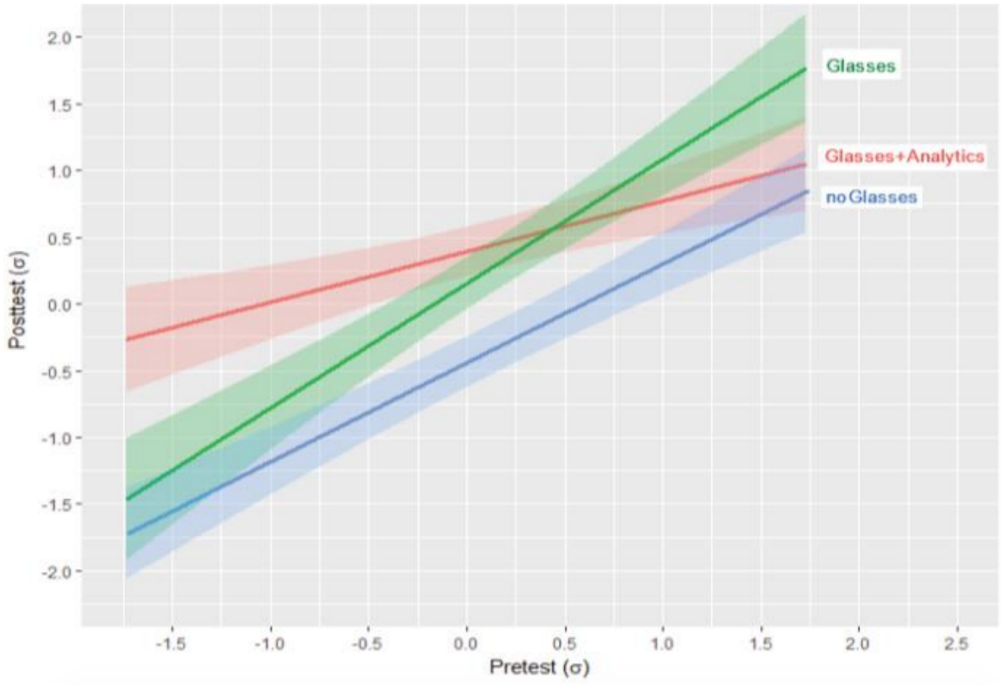

Figure 7-2. Student posttest scores plotted by pretest scores, for each experimental condition. Lines indicate condition means; shaded regions indicate standard error; overlapping shaded regions indicate overlapping standard errors (figure from Holstein et al., 2018b).

I constructed an HLM with posttest as the dependent variable and pretest and experimental condition as level-1 covariates, modeling interactions between pretest and condition. Figure 7-2 shows student posttest scores plotted by pretest scores (in standard deviation units) for each of the three conditions. As shown, students in the Glasses condition learned more overall, compared with the noGlasses condition, but the disparity in learning outcomes across students with varying prior domain knowledge remained the same. For students in the Glasses+Analytics condition, the posttest by pretest curve was flatter, with lower pretest students learning considerably more than in the other two conditions. There was no significant interaction between noGlasses/Glasses and student pretest. However, there were significant negative interactions between student pretest scores and noGlasses/Glasses+Analytics ($t(46) = -2.456$, $p = 0.018$, $r = -0.15$, 95% CI [-0.26, -0.03]) and Glasses/Glasses+Analytics ($t(164) = -2.279$, $p = 0.024$, $r = -0.16$, 95% CI [-0.27, -0.05]), suggesting that a teacher's use of real-time analytics may serve as an equalizing force in the classroom.

Effects on Teacher Time Allocation.

As an additional way of testing whether the real-time analytics provided by *Lumilo* had their intended effect, I fit an HLM with teacher time allocation, per student, as the dependent variable (i.e., to test **Q2** and **Q3** of Causal Alignment Analysis, as described in *Chapter 6*), and student pretest score, experimental condition, and their interactions as fixed effects. Figure 7-3 shows teacher time, plotted by student pretest, for each condition. As shown, in the Glasses+Analytics condition, teachers tended to allocate considerably more of their time to students with lower prior domain knowledge, compared to the other conditions. There was no significant main effect of noGlasses/Glasses on teacher time allocation ($t(211) = 0.482$, $p = 0.63$, $r = 0.03$, 95% CI [0, 0.14]), nor a significant interaction with pretest. However, there were significant main effects of noGlasses/Glasses+Analytics ($t(279) = 2.88$, $p = 4.26*10^{-3}$, $r = 0.17$, 95% CI [0.06, 0.28]) and Glasses/Glasses+Analytics ($t(278) = 2.02$, $p = 0.044$, $r = 0.12$, 95% CI [0.01, 0.23]) on teacher time allocation. In addition, there were significant negative interactions between student pretest and noGlasses/Glasses+Analytics ($t(279) = -2.88$, $p = 4.28*10^{-3}$, $r = -0.17$, 95% CI [-0.28, -0.05]) and Glasses/Glasses+Analytics ($t(275) = -3.546$, $p = 4.62*10^{-4}$, $r = -0.23$, 95% CI [-0.33, -0.11]).

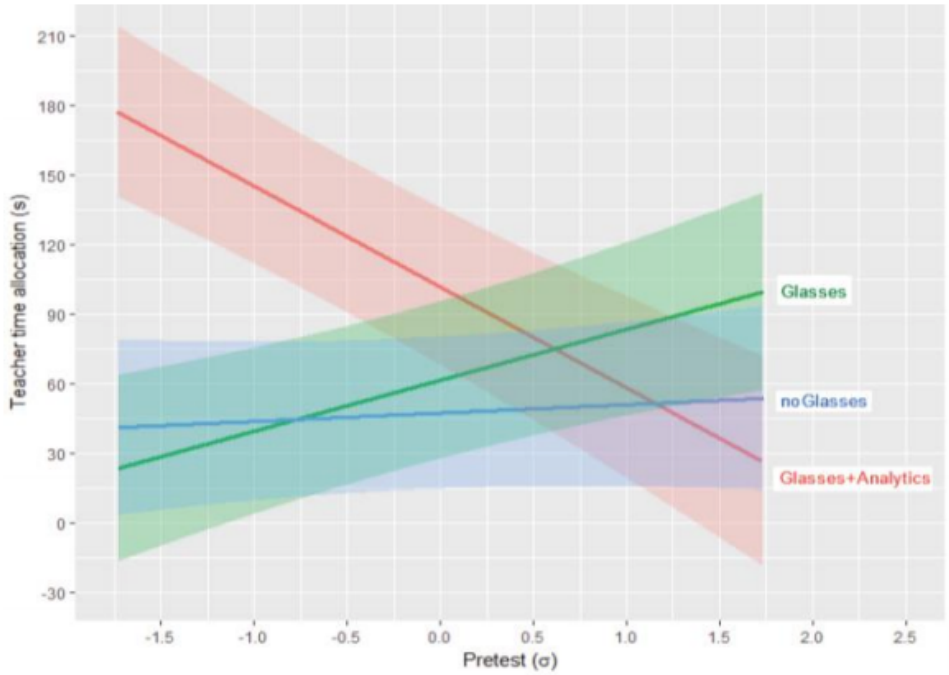

Figure 7-3. Teacher attention allocation (in seconds), plotted by pretest scores, for each experimental condition. Lines indicate condition means; shaded regions indicate standard error; overlapping shaded regions indicate overlapping standard errors (figure from Holstein et al., 2018b).

I also investigated how teachers' relative time allocation across students may have been driven by the real-time analytics presented in the Glasses+Analytics condition. Specifically, I examined whether and how teacher time allocation varied across conditions, based on the frequency with which a student exhibited each of the within-tutor behaviors/states detected by *Lumilo* (i.e., *Lumilo*'s student indicators, described in *Chapters 4* and *6*). I constructed HLMs with teacher time allocation as the dependent variable, and the frequency of student within-tutor behaviors/states, experimental condition, and their interactions as fixed effects. Row 3 of Table 7-1 shows relationships between student within-tutor behaviors/states and teacher time allocation across students, for the Glasses+Analytics vs. noGlasses (GA v. nG) comparison. As shown, teachers' time allocation across students appears to have been influenced by *Lumilo*'s real-time indicators. Compared with business-as-usual (Row 3, Table 7-1), teachers in the Glasses+Analytics condition spent significantly less time attending to students who frequently exhibited low local error, and significantly more time attending to students who frequently exhibited undesirable behaviors/states detected by *Lumilo*, such as unproductive persistence.

Table 7-1. Estimated effects of condition (rows) on teachers' allocation of time to students exhibiting each within-tutor behavior/state (columns). Cells report estimated effect sizes:
*** p<0.001, ** p<0.01, * p<0.05, ~ 0.05< p<0.07

	high local error	hint abuse or gaming	hint avoidance	high error after hints	idle	low local error	rapid attempts	unproductive persistence ("wheel-spinning")
G v. nG	n.s.	n.s.	n.s.	n.s.	n.s.	0.13 ~	n.s.	n.s.
GA v. G	**0.20 ***	**0.17 ***	**0.19 ***	**0.18 ***	**0.22 ****	**- 0.51 *****	n.s.	**0.35 *****
GA v. nG	**0.16 ****	0.10 ~	**0.14 ***	0.11 ~	**0.17 ****	**- 0.23 *****	n.s.	**0.24 *****

Rows 1 and 2 of Table 7-1 show estimates for Glasses vs. noGlasses (G v. nG) and Glasses+Analytics vs. Glasses (GA v. G), respectively. As shown, there were no significant differences in teacher time allocation due to the introduction of the glasses themselves, suggesting *Lumilo*'s overall effects on teacher time allocation may result primarily from teachers' use of the advanced analytics presented in the GA condition.

Effects of Classroom Monitoring Support and Real-time Teacher Analytics on Student-level Processes.

To investigate potential effects of experimental condition on the frequency of student within-tutor behaviors and learning states detected by *Lumilo*, I constructed HLMs with students' within-tutor behaviors/states as the dependent variable, and pretest score and experimental condition as fixed effects. Row 3 of Table 7-2 shows estimated effects of classroom condition on the frequency of student within-tutor behaviors/states, for Glasses+Analytics vs. noGlasses (GA v. nG).

Table 7-2. Estimated effects of condition (rows) on the frequency of student within-tutor behaviors/states (columns):
*** p<0.001, ** p<0.01, * p<0.05, ~ 0.05< p<0.07

	high local error	hint abuse or gaming	hint avoidance	high error after hints	idle	low local error	rapid attempts	unproductive persistence ("wheel-spinning")
G v. nG	- 0.36 **	- 0.21 **	- 0.32 **	n.s.	0.23 *	0.34 ***	n.s.	n.s.
GA v. G	- 0.12 ~	n.s.	n.s.	n.s.	n.s.	n.s.	n.s.	- 0.20 ~
GA v. nG	**- 0.47 *** **	**- 0.28 ** **	**- 0.41 *** **	**- 0.30 *** **	**0.26 * **	**0.42 * **	**- 0.34 ** **	**- 0.15 * **

Compared with business-as-usual, students in the Glasses+Analytics condition exhibited less hint avoidance or gaming / hint abuse, were less frequently detected as unproductively persisting or making rapid consecutive attempts in the tutoring software and exhibited less frequent high local error. In addition, students in the Glasses+Analytics condition were more frequently idle in the software, and more frequently exhibited low local error. Row 1 of Table 7-2 suggests that the introduction of the glasses, even without real-time teacher analytics, may have had a considerable influence on students' behavior within the software. By contrast, there were no significant differences between the Glasses+Analytics and Glasses conditions. These results suggest that, despite the ostensible positive effects of real-time teacher analytics on student learning outcomes, some of the largest effects of *Lumilo* on students' within-tutor behavior may result primarily from teachers' use of the monitoring support provided in the Glasses condition, rather than from a teachers' use of advanced analytics.

7.4 Conclusions

I conducted a three-condition classroom experiment to investigate the effects of real-time teacher/AI co-orchestration on student learning in ITS classrooms. The results indicated that a teacher's use of *Lumilo* had a positive impact on student learning with an ITS, compared with both business-as-usual (where the teacher did not have any real-time support) and a second, stronger control condition in which the teacher had access to a simpler form of classroom monitoring support via mixed-reality glasses (i.e., support for peeking at a student's screen remotely), but without any advanced analytics.

In particular, presenting teachers with real-time analytics about student learning, metacognition, and behavior at a glance had a positive impact on student learning with the ITS, above and beyond the effects of monitoring support alone (without any advanced analytics). The real-time analytics provided by *Lumilo* appear to have served as an equalizing force in the classroom. That is, whereas the use of AI in K-12 education has sometimes been shown, in prior studies, to have a "rich get richer" effect, *increasing* within-classroom achievement gaps (Holstein et al., 2018b; Holstein & Doroudi, 2018; Rau, 2015; Reich & Ito, 2017), teachers' use of *Lumilo* in the classroom had the effect of *narrowing* the gap in learning outcomes across students of varying prior ability, as measured by the pre-test, by benefiting students of lower prior ability, but without measurably affecting students of higher prior ability. This effect appears to have been mediated, in part, by an increase in teachers' time allocation towards students of lower prior ability over the course of a class session in the Glasses+Analytics condition, compared with the other two conditions.

Finer-grained analyses of teachers' moment-by-moment movement throughout their classrooms revealed that teachers in the "Glasses+Analytics" condition spent significantly more of their time, compared with the "noGlasses" condition, working with students who frequently exhibited unproductive persistence (here measured by "wheel spinning," as in Beck & Gong, 2013; Kai et al., 2017), hint avoidance (see Aleven, Roll, et al., 2016), or idle periods in the software greater than 80 seconds (see Baker, 2007); and significantly less of their time working with students who frequently exhibited a low recent error rate in the software (see Pelánek & Řihák, 2017). Teasing apart potential effects of a teacher's use of real-time analytics versus the simpler form of monitoring support provided in the "Glasses condition, I found that a teacher's use of analytics explained these associations with teachers' time allocation.

Although much prior work has focused on the design, development, and evaluation of teacher analytics tools, very few studies have evaluated effects on student learning (Kelly et al., 2013; Molenaar & Knoop-van Campen, 2017; Rodríguez-Triana, et al., 2017; Xhakaj et al., 2017). Prior work has found that providing teachers with real-time notifications about student performance can direct their attention to low-performing students, resulting in local performance improvements (e.g., Martinez-Maldonado et al., 2015). Other recent work has begun

systematically investigating how teachers use real-time progress and performance analytics in blended classrooms (e.g., Molenaar & Knoop-van Campen, 2017). However, the present work is the first experimental study showing that real-time teacher analytics can enhance students' learning outcomes (within or outside the context of AI-supported classrooms; see items 1 and 3 under *Summary of Contributions – "First broad design exploration of needs for real-time teacher analytics and orchestration support"* and *"First experimental study to demonstrate learning benefits of real-time teacher analytics"*).

Interestingly, part of *Lumilo*'s overall effect on student learning appeared to be attributable to monitoring support alone. Follow-up correlational analyses suggested that a teacher's use of the glasses, with a simpler form of monitoring support, but without advanced analytics, may have reduced students' frequency of maladaptive learning behaviors (such as gaming/hint-abuse) without significantly influencing teachers' time allocation across students. These results suggest that the observed learning benefits of monitoring support may be due to a motivational effect, resulting from students' awareness that a teacher is monitoring their activities in the software (cf. Holstein et al., 2017a; Stang & Roll, 2014), and/or due to a novelty effect. It may also be that the monitoring support provided in the Glasses condition had a positive effect on teacher behavior that is not reflected in the way they distributed their time across students (e.g., perhaps there were beneficial changes in the kinds of help teachers gave, or in their non-verbal behaviors). However, future research is needed to tease apart these alternative explanations.

Importantly, these findings suggest that, when evaluating the impacts of teacher-facing learning analytics tools, future research should take care to tease apart potential effects of a teacher's use of a monitoring tool (such as novelty effects or students' awareness of being monitored by their teacher), versus teachers' use of the kinds of advanced analytics and student modeling methods that are often the focus of research in learning analytics (LA), AI in education (AIED), user modeling (UM), and educational data mining (EDM) (see item 5 under *Summary of Expected Contributions – "First investigations of relationships between teacher movement/monitoring and student behavior and learning in AI-supported classrooms"*).

I see several exciting directions for future work. The current study involved teachers with at least five years of mathematics teaching experience. However, my prior design work with teachers indicated that less-experienced teachers may often struggle to generate effective on-the-spot help, in response to real-time analytics from an ITS (see *Chapter 4* and *Chapter 5*). Thus, a promising direction for future design research is to investigate differences in needs for real-time support across teachers with varying levels of experience. In addition, while the current study was conducted over a single week of class time, future longitudinal studies may shed light on whether and how the effects of real-time teacher analytics and monitoring support may evolve over longer-term use (cf. Molenaar & Knoop-van Campen, 2017). More broadly, an exciting direction for future work is to better understand and characterize the complementary strengths of

human and automated instruction, in order to explore how they can most effectively be combined (cf. Holstein et al., 2017b; 2019a; Ritter et al., 2016b).

In sum, the findings presented in this chapter illustrate the potential of AIED systems that integrate human and machine intelligence to support student learning. In addition, this work illustrates that the kinds of analytics *already* generated by ITSs, using student modeling techniques originally developed to support adaptive tutoring behavior, appear to provide a promising foundation for real-time teacher support tools.

Chapter 8

Lumilo Goes to School in the Big City: Classroom Observations and Feedback Sessions

This chapter is based in part on the following publications:

- Holstein, K., McLaren, B. M. & Aleven, V. (2019a). Co-designing a real-time classroom orchestration tool to support teacher–AI complementarity. *Journal of Learning Analytics (JLA)*.

8.1 Motivation

Chapters 6 and *7* focused primarily on quantitative research findings from classroom pilots and in-vivo classroom experiments. However, deploying in 30 live K-12 classrooms also provided an opportunity to hear student and teacher perspectives on the experience of using the *Lumilo* prototype in the classroom.

In this chapter, I share observations from piloting *Lumilo* "in the wild," with a focus on needs and nuances that were not captured in my earlier design research with teachers (see items 1 and 2 under *Summary of Contributions – "First broad design exploration of needs for real-time teacher analytics and orchestration support"* and *"First design exploration and prototypes of wearable, heads-up displays to support orchestration of personalized classrooms"*).

These field observations in turn serve as the foundation for more targeted, detailed investigations in *Part Four* of this thesis. *Chapter 9* further explores needs observed during these classroom pilots, through concept generation and validation exercises with both teachers and students. *Chapter 10* then compares findings from the current studies with findings from *Lumilo* pilots in classrooms that use a different ITS (Carnegie Learning's *MATHia* software).

8.2 Methods

Each class session began with the teacher introducing the glasses to the class, which was often accompanied by an invitation to laugh at the teacher's appearance while wearing the prototype (see Figure 8-1, left). For example, near the beginning of class, one teacher told students, *"Alright everyone get the giggles out now. You can laugh at me for the next five minutes. But after that it's time to work."* For these feedback sessions, audio recording was prohibited by our study's IRB and some schools, to avoid the possibility of collecting potentially identifiable data on young students. Throughout each classroom pilot session, a researcher took live notes on

classroom observations. Throughout each classroom experiment session, a researcher manually transcribed teacher (but not student) dialogue live, using the LookWhosTalking tool[21]. Teacher utterances captured with this tool were automatically time-stamped, to facilitate later triangulation between time-stamped teacher dialogue, and the real-time analytics a teacher was seeing at a given moment through the *Lumilo* prototype (as logged to DataShop). In ten (out of thirty-six) class sessions that were so noisy as to make manual, live dialogue transcription impractical (e.g., due to many students talking at once), the researcher took notes on classroom observations instead of conducting live transcriptions of teacher utterances.

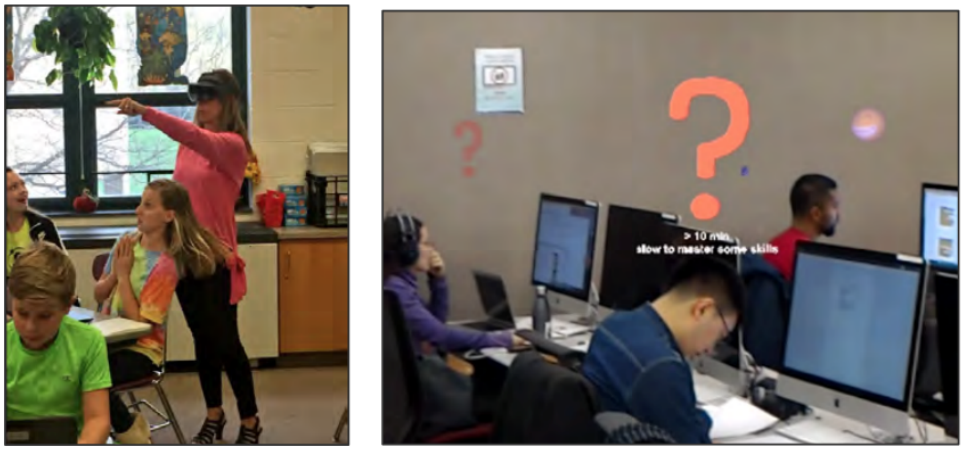

Figure 8-1. Left: A teacher using *Lumilo* in a live middle school math classroom while her students work with *Lynnette*, an ITS for linear equation solving (from Holstein et al., 2018b). Right: An illustrative point-of-view screenshot through *Lumilo*, captured during earlier prototyping sessions at our institution.

For 20 to 30 minutes at the end of each study, the teacher would invite the whole class to reflect on the experience, and provide design feedback from the student perspective. Given that audio recording was not permitted for these feedback sessions, a researcher took detailed notes on student feedback throughout each session, with the teacher serving as the primary session facilitator. At the end of the school day, teachers then participated in a loosely structured post-interview, lasting approximately 30 minutes, to share their own reflections on the experience. These teacher-only sessions were audio recorded and later transcribed. To analyze data from classroom observations and feedback sessions, we worked through classroom notes and approximately 14 hours of transcribed audio to synthesize findings using a thematic analysis approach (Hanington & Martin, 2012). Key findings and reflections are briefly summarized in the following subsection.

[21] *https://bitbucket.org/dadamson/lookwhostalking/*

8.3 Findings

From the first classroom pilot onward, teacher and student responses to Lumilo were much more positive than I would have expected for an initial venture outside of the lab. By the time I entered the classroom, I had already encountered many "surprises" in Replay Enactments sessions by testing with datasets from a (relatively) diverse range of classroom contexts, and had iterated on Lumilo's design accordingly. In some cases, teachers found that particular design features that had emerged through iterative prototyping were even more useful in live classrooms than they had anticipated during REs. For example, one teacher – who had first participated in an REs session and then piloted *Lumilo* in his own classes – was frequently observed "multitasking" during a class session: using the glasses to peek at analytics for students across the room, and interleaving quick feedback between multiple students, even in the middle of working face-to-face with a particular student. This teacher reflected that he did not feel as strong a need to multitask in the REs study, but in a live classroom where multiple students were constantly vying for the teacher's attention,

> *"[The ability to] take a student's screen with me, even if I'm over here working with another student is amazingly useful… that was well thought through."*

Teachers reported making frequent use of Lumilo's analytics to identify students who might need their help — in some cases, directly referencing these analytics in one-on-one conversations with a student:

> *"You have a smiley face [right now], but you're still having trouble with adding and subtracting variables from both sides ... That's what you need to watch, when you have variables on both sides, you need to subtract on both sides [not just one]."*

In other cases, teachers used Lumilo's alerts in ways they had not anticipated during REs think-aloud sessions. For example, in one class, a teacher noticed a "system misuse" alert over one student's head, with elaboration text indicating that this student seemed to be "Abusing hints?" However, the teacher believed that hint abuse was out of character, given what they knew about this particular student. When the teacher approached to find out how the student was doing, they learned that this student was actually colourblind and thus could not perceive the correctness feedback provided by the tutoring system (which was coded green for "correct" and red for "incorrect"). This had led the student to grow frustrated, and ostensibly, to rely on the tutoring software's "hint" function more. In a similar case, a teacher noticed that an otherwise diligent student had been idle in the software for over five minutes. The teacher approached this student and noticed that this student was playing online video games instead of doing the assigned work. The teacher asked how the student was feeling that day, which led the student to disclose that they had broken up with a significant other over the preceding weekend. In response, the teacher gave the student permission to take the day off from math, if needed.

Overall, students also reacted very positively to teachers' use of Lumilo during ITS lab sessions. At the beginning of one class session, a student said,

"I'm a little afraid. [The researcher's] gonna let [the teacher] spy on us…"

During class, however, the student appeared to warm up to the idea of their teacher using the prototype. On multiple occasions, the teacher approached the student to provide unsolicited help, based on what the teacher was seeing through the glasses. At the end of one of these student–teacher exchanges, after the student completed the majority of a problem without the teacher's assistance, the student and teacher high-fived. During the end-of-class feedback session, the same student said,

"It was awesome how [the teacher] just knew when I needed help."

In another classroom, a student – excited by the concept of becoming a *"cyborg teacher"* – exclaimed, *"I want to be a teacher when I grow up!"*

Piloting Lumilo in 30 live middle-school classrooms also revealed several critical needs that Lumilo's design did not address. For example, both students and teachers across multiple classrooms emphasized needs for additional features to support "anonymous" non-face-threatening communication between students and teachers during a class session, extending beyond "invisible hand raises" (a potential feature that was discussed in *Chapter 4*, but which was not activated in the version of *Lumilo* that was used in classrooms due to initial teacher concerns about student misuse or overuse of such functionality). In the absence of such features, students sometimes took matters into their own hands. For example, in one class session, a student appropriated the equation-entry box in *Lynnette*'s interface to write "secret" messages to their teacher, viewable through the teacher's glasses.

In addition, while teachers reported that it rarely made sense to give a whole-class lecture given the non-synchronized nature of ITS class sessions, they realized that it would be useful to have more support in dynamically deciding between small group ("pull out") interventions versus interventions with individual students. While the Lumilo's design included analytics at the individual and whole-class levels, the design did not facilitate rapid filtering of students to identify relevant student subgroups. One teacher suggested it might have been helpful

> *"[if] the class [dashboard would let] you zoom in and see which students are struggling with a particular skill."*

Finally, after using Lumilo in the classroom, multiple teachers raised needs for greater transparency and control. For example, one teacher noted that although the analytics presented by *Lumilo* provide some insight into why the ITS might be making certain decisions (e.g., relating to adaptive problem selection),

> *"When a student asks me why they have to do twenty problems in level three [of the ITS], before [moving on], but another student only has to do two problems… I should be able to answer that."*

In addition, some teachers noted that the level of transparency currently provided by Lumilo helped make them more aware of some of the limitations of the intelligent tutoring system their class was using (and associated student modelling techniques). Given this enhanced awareness, teachers reported frustration over their relative lack of control (cf. Lee & Baykal, 2017). One teacher noted that it would be helpful if they could provide feedback when Lumilo's alerts miss the mark, to customize the alerts to their needs.

Similarly, this teacher and their students agreed that it would be nice if *students* had the option to see their own student model (including skill mastery estimates, metacognitive variables, and other information that the teacher can see through *Lumilo*) and contest what it says about their knowledge or behaviour (cf. Bull & Kay, 2016), perhaps allowing the teacher to review and approve these cases individually during class.

Another teacher mentioned that when using *Lumilo*, they were seeing students struggle with the same issues over and over again, and the software's built-in hints did not seem to be helping in these cases. This teacher suggested that instead of "filling in" for the ITS by repeatedly giving different students the same feedback,

> *"It would be nice if [the ITS] could listen to what I tell [the student, and] just say that the next time a student gets stuck [with the same issue]."*

8.4 Conclusions

Building upon early design and data mining investigations presented in *Chapters 1, 2,* and *4*, the prototyping studies and classroom experiments presented in *Chapters 5* through *7* demonstrate promise for "hybrid" approaches to AI-supported education, which integrate teacher and machine intelligence to support students' learning. However, studies in live K-12 classrooms also revealed broader needs for orchestration support in these settings – among both teachers and students – extending beyond those addressed by *Lumilo*'s current design.

For example, both teachers and students expressed needs for better mechanisms to support "private" teacher–student communication during a class session (e.g., to enable students to signal help-need during class without losing face to peers). In addition, after using *Lumilo* in live K-12 classrooms, teachers began to reveal more nuanced preferences for which classroom tasks should be handled by the AI, which should be handled by the teacher, and which should be handled by students (and under which circumstances). Similarly, students began to reveal needs for greater

agency over how their personal analytics are used and interpreted than *Lumilo* (and associated ITSs) currently provides.

Building upon these and other findings, in *Part Four* of this dissertation, I involve students, as well as teachers, in the next phase of design. Through iterative concept generation and validation exercises, I work with students and teachers to better understand their respective needs and boundaries (*Chapter 9*). In a subsequent classroom study, conducted in classrooms using Carnegie Learning's widely-used *MATHia* tutoring software (*Chapter 10*), I then follow up on findings from *Chapter 9*, to further validate these findings with participants who have had recent experiences using a form of real-time wearable teacher augmentation (a modified version of *Lumilo*) to support their classroom activities.

Part Four

Preparing for Broader Use: Implications for Cyborg Teachers in the Wild

In the final chapters of my thesis on real-time, wearable teacher augmentation (RWTA), I begin to explore challenges in preparing this concept for wider-spread use. Over the past two years, there has been considerable excitement, from teachers, students, school administrators, and educational technology companies, in turning *Lumilo* into a widely-available classroom tool. My colleagues and I have recently formed an academic–industry partnership with Carnegie Learning, a major educational technology company that is interested in bringing this concept to a larger audience. Carnegie Learning's AI tutoring software *MATHia* is used by over 2,000 schools and 500,000 students each year.

It is worth noting that the Research through Design (RtD) work I have conducted up to this point has not been motivated by the possibilities of near-term productization or scaling. Had these been major, near-term objectives early on, I most likely would have taken a much more technologically conservative route after the explorations presented in *Part One* of this dissertation. For example, I may have restricted my subsequent design explorations in *Part Two* to technologies that were already relatively low cost, widely adopted, and familiar in K-12 classroom settings (none of which were properties of mixed reality heads up displays at the time; see Harrison, 2018).

My primary goals in *Parts One* through *Three* of this dissertation have been to explore and understand possible futures for AI in education (AIED), in which AIED systems are designed to augment and amplify teachers' complementary abilities, instead of necessarily trying to automate these away (Baker, 2016; Holstein et al., 2017b; Ritter et al., 2016). However, this research has since expanded beyond its initial scope and goals, and it now appears that RWTA may be headed for wider use (i.e., beyond short-term research studies in local K-12 classrooms).

Scaling up RWTA not only introduces new technical and interface design challenges, it **requires a shift to a broader framing of the design problem**. My prior work has focused on an important gap in the AIED research literature: working with K-12 teachers, I have explored how we might design to empower teachers to help their students during AI-supported class sessions. However, a system intended for broader use, beyond the context of short-term research studies, should be designed to serve the needs (and respect the boundaries) of all stakeholders within the classroom. In keeping with the ethos of empowerment central to this thesis (Kulkarni, 2019; Toyama, 2018), it is necessary to involve not just teachers but also students in the design process (cf. Prieto-Alvarez, Martinez-Maldonado, & Anderson, 2018).

While prior chapters have focused on the design of more effective teacher–AI partnerships in K-12 classrooms, the final chapters of this thesis begin to explore how we might design for mutually desirable *student–teacher–AI partnerships*. Specifically, I conceptualize RWTA as just one teacher-facing component of an integrated human–AI co-orchestration system that helps to "choreograph" interactions between teachers, students, and AI agents in the classroom (see *Conclusions, Contributions, and Future Directions* for a discussion).

As a design-oriented HCI researcher, I believe it is important to collaborate with companies to help shape the ways academic research is used – as opposed to simply handing off our research and having industry "take it from there" (Blikstein, 2018; Holstein & Doroudi, 2019) or passively critiquing product design decisions "from the sidelines" (Buckingham Shum, 2018; Holstein, Wortman Vaughan, et al., 2018; 2019; Veale et al., 2018). As an initial step towards designing a system that is "ready" to be scaled up (in collaboration with Carnegie Learning), in **Chapter 9** I investigate how human–AI co-orchestration systems can be better designed to serve the needs (and respect the boundaries of *both* teachers and students, building upon observations and feedback from classroom pilots and experiments with *Lumilo* (see *Chapter 8*).

In addition, as a step towards demonstrating the feasibility of generalizing RWTA beyond use with a single tutoring system, in **Chapter 10** I explore what new design challenges arise when generalizing RWTA for use with a new tutoring system (Carnegie Learning's *MATHia*) that is used over significantly longer timescales, covers a much broader range of curricular content, and is used in a broader range of classroom contexts.

Beyond the scope of this thesis, the work presented in *Part Four* will help prepare for the next phase of this research: a large-scale classroom experiment (using an updated and miniaturized version of *Lumilo*) with over 60 middle school classrooms that use *MATHia*, to better understand the effects of human–AI co-orchestration on student learning and other outcomes of interest.

Chapter 9

My Teacher is a Cyborg: Designing for More Desirable Student–Teacher–AI Interactions in AI-supported Classrooms

This chapter is based in part on the following publications:

- Holstein, K., McLaren, B. M., & Aleven, V. (2019b). Designing for Complementarity: Teacher and Student Needs for Orchestration Support in AI-Enhanced Classrooms. In *Proceedings of the International Conference on Artificial Intelligence in Education (AIED 2019)* (pp. 157-171). Springer, Cham.

9.1 Background and Motivation

If real-time, wearable teacher augmentation (RWTA) is to be used in actual classrooms, beyond the context of short-term research studies, it is critical that these systems be carefully designed to respect the needs and boundaries of both teachers and students (Dillenbourg & Jermann, 2010; Rummel, Walker, & Aleven, 2016; Schofield, 1997; Zimmerman & Forlizzi, 2017). Among other considerations, this requires a detailed understanding of teacher and student preferences regarding which classroom roles (and under which circumstances) to augment with AI, which roles to automate, and which to leave fully to human teachers or peers (du Boulay, Luckin, & del Soldato, 1999; Davidoff et al., 2007; Holstein et al., 2017b; 2019a; Lubars & Tan, 2019). Close involvement of both teachers and students throughout the design and prototyping process can help in understanding where particular forms of AI automation or augmentation may help more than hurt (Holstein et al., 2017b; 2019a; 2019b; Lubars & Tan, 2019).

For example, prior design research with K-12 teachers has found that there is a delicate balance between automation and respecting teachers' autonomy (see *Chapters 1*, *4*, and *5,* and Heer, 2019; Holstein et al., 2017b; Olsen, 2017; Olsen, Rummel, & Aleven, 2018; van Leeuwen et al., 2018; Lubars & Tan, 2019). Over-automation risks taking over classroom roles that teachers would prefer to perform (e.g., providing socio-emotional support) and threaten their flexibility to set their own instructional goals. On the other hand, under-automation risks burdening teachers with tasks they would rather not perform (e.g., routine grading), and may limit the degree of personalization they can feasibly achieve in the classroom (Holstein et al., 2017b; Holstein, Hong, et al., 2018; Olsen, 2017). Similarly, from students' perspectives, AI systems that attempt to over-automate "caring" tasks such as providing motivational or emotional support may fail to serve their needs (Bartneck & Forlizzi, 2004; Ritter et al., 2016b; Schofield, 1994; Watters,

2014) or may be perceived as patronizing (Holstein et al., 2019b). On the other hand, forms of teacher augmentation like *Lumilo* may risk revealing sensitive information that students would be more comfortable with an AI system knowing than their teacher (cf. Lucas, Gratch, King, & Morency, 2014), and thus be perceived as invasive, creepy, or threatening to their autonomy as learners (Holstein et al., 2019a; 2019b; Manolev, Sullivan, & Slee, 2018; Molenaar, Horvers, & Baker, 2019; Williamson, 2016; 2017). Yet prior work on human–AI co-orchestration has generally focused on the needs of K-12 teachers, but not students' perspectives, in AI-supported classrooms (Holstein et al., 2017b; Olsen, 2017; Olsen et al., 2018; van Leeuwen et al., 2018; Wetzel et al., 2018).

This chapter builds on prior findings to contribute: (1) an analysis of teacher and student feedback regarding 24 design concepts for human–AI co-orchestration systems, to understand key needs and social boundaries that such systems should be designed to address (Dillahunt, Lam, Lu, & Wheeler, 2018; Friedman et al., 2008; Zhu et al., 2018; Zimmerman & Forlizzi, 2017) (see item 1 under *Summary of Contributions – "First broad design exploration of needs for real-time teacher analytics and orchestration support"*), and (2) "Participatory Speed Dating": a new variant of the speed dating design method (Davidoff et al., 2007; Zimmerman & Forlizzi, 2017) that involves multiple stakeholders in the generation and evaluation of novel technology concepts (see item 4 under *Summary of Contributions – "Novel design and prototyping methods"*).

9.2 Methods

To better understand and validate needs uncovered in prior ethnographic and design research with K-12 students and teachers (e.g., Feng & Heffernan, 2006; Holstein et al., 2017b; 2018a; 2019a; Holstein, Hong, et al., 2018; Olsen et al., 2017; Schofield, 1997; Schofield et al., 1994), I adopted a Participatory Speed Dating approach. Speed Dating is an HCI method for rapidly exploring a wide range of possible futures with users, intended to help researchers/designers elicit unmet needs and probe the boundaries of what particular user populations will find acceptable (which otherwise often remain undiscovered until after a technology prototype has been developed and deployed) (Davidoff et al., 2007; Odom et al., 2012; Zimmerman & Forlizzi, 2017). In Speed Dating sessions, participants are presented with a number of hypothetical scenarios in rapid succession (e.g., via storyboards) while researchers observe and aim to understand participants' immediate reactions.

Speed dating can lead to the discovery of unexpected design opportunities, when unanticipated needs are uncovered or when anticipated boundaries are discovered not to exist. Importantly, speed dating can often reveal needs and opportunities that may not be observed through field observations or other design activities (Davidoff et al., 2007; Dillahunt et al., 2018; Odom et al., 2012; Zimmerman & Forlizzi, 2017). For example, Davidoff et al. found that, whereas field

observations and interview studies with parents had suggested they might appreciate smart home technologies that automate daily household tasks, a speed dating study revealed that parents strongly rejected the idea of automating certain tasks, such as waking or dressing their children in the morning. These findings led the researchers to reframe their project—away from creating smart homes that "do people's chores," towards homes that facilitate moments of bonding and connection between busy family members (Zimmerman & Forlizzi, 2017).

In this study, I adapted the Speed Dating method to enable participants from multiple stakeholder groups (K-12 teachers and students) to reflect on other stakeholders' needs and boundaries, and contribute ideas for new scenarios and technology concepts. I refer to this adaptation as multi–stakeholder "Participatory Speed Dating" (PSD). Like other Speed Dating approaches, PSD can help to bridge between broad, exploratory design phases and more focused prototyping phases (where associated costs may discourage testing a wide range of ideas) (Davidoff et al., 2007; Dow et al., 2010; Zimmerman & Forlizzi, 2017). However, drawing from approaches such as Value Sensitive and Service Design (Forlizzi & Zimmerman, 2013; Payne et al., 2008; Zhu et al., 2018), PSD emphasizes a systematic approach to balancing multiple stakeholder needs and values (Miller et al., 2007; Zhu et al., 2018). Drawing from Participatory Design (Luckin & Clark, 2011; Trischler, Pervan, Kelly, & Scott, 2018; Walsh, Foss, Yip, & Druin, 2013), in addition to having stakeholders evaluate what is undesirable about a proposed concept (potentially representing design elements that address *other* stakeholders' needs), PSD also involves them in generating alternative designs, to address conflicts among stakeholder groups.

I conducted PSD sessions one-on-one with 24 middle school teachers and students. To recruit participants, I emailed contacts at eight middle schools and advertised the study on Nextdoor, Craigslist, and through physical fliers. A total of 10 teachers and 14 students, from two large US cities, participated in the study. Sixteen sessions were conducted face-to-face at our institution, and eight were conducted via video conferencing. All participants had experience using some form of adaptive learning software in their classrooms, and 21 participants had used AI tutoring software such as McGraw Hill Education's *ALEKS* (Hagerty & Smith, 2005) or Carnegie Learning's *Cognitive Tutor* or *MATHia* (Ritter et al., 2007; Ritter, Carlson, Sandbothe, & Fancsali, 2005).

We first conducted a series of four 30-minute study sessions focused on concept generation, with two teachers and two students. In each session, participants were first introduced to the context for which they would be designing: classes in which students work with AI tutoring software while their teacher uses a real-time co-orchestration tool that helps them help their students (specifically, a set of teacher smart glasses, following my earlier design explorations; see *Chapters 4* through *8*). Participants were then shown an initial set of 11 storyboards, each created to illustrate specific classroom challenges uncovered in prior research (e.g., Feng & Heffernan, 2006; Holstein et al., 2017b; Ritter et al., 2016a; Schofield et al., 1994; Schofield, 1997), with multiple challenges hybridized (Davidoff et al., 2007) into a single storyboard in some cases. For

example, prior work suggests that teachers often struggle to balance their desire to implement personalized, mastery-based curricula with their need to keep the class relatively synchronized and "on schedule" (Holstein et al., 2017b). Given this conflict, teachers often opt to manually push students forward in the curriculum if they have failed to master current skills in the ITS by a certain date, despite awareness that this practice may be harmful to students' learning (Holstein et al., 2017b; Ritter et al., 2016a). As such, one storyboard (Figure 9-1) presented a system that helps teachers make more informed decisions about when to move students ahead (based on the predicted learning benefits of waiting a few more class periods), but without strongly suggesting a particular course of action (Holstein et al., 2017b).

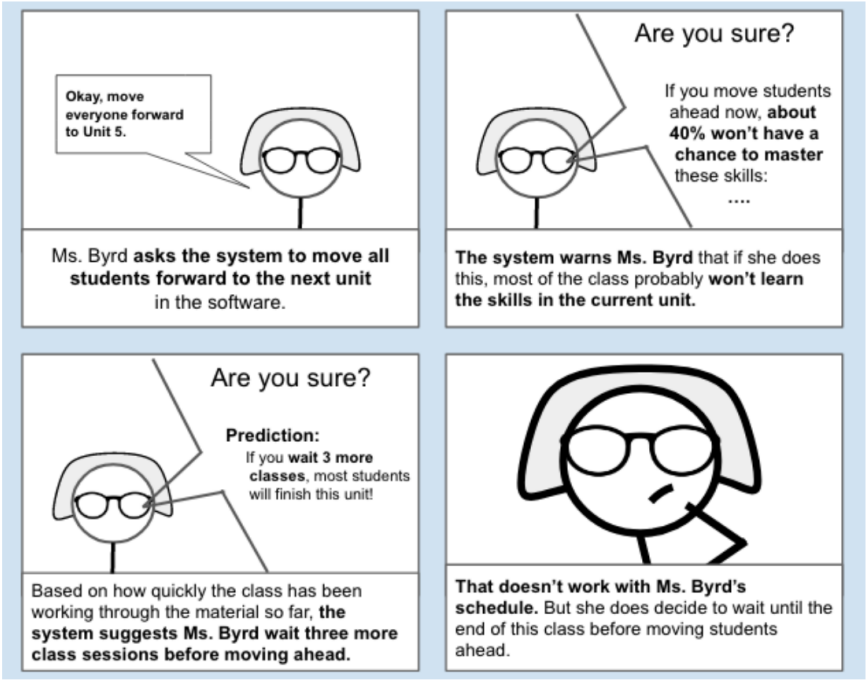

Figure 9-1. Example of a storyboard addressing challenges raised in prior research.

Each participant in these initial studies was then encouraged to generate at least one new idea for a storyboard, addressing challenges they personally face in AI-enhanced classrooms as opposed to imagined challenges of others (cf. Dillahunt et al., 2018). To inform ideation, participants also reviewed storyboards generated by other teachers and students in prior study sessions. Participants were provided with editable storyboard templates, in Google Slides[22], and were given the options to generate entirely new concepts for orchestration tool functionality (starting from a blank template) or to generate a variation on an existing concept (starting from a copy of an existing storyboard). In either case, participants generated captions for storyboard panels during the study session, using existing storyboards for reference. Immediately following each

[22] *http://slides.google.com*

session, a researcher then created simple illustrations to accompany each caption (cf. Hiniker et al., 2017).[23]

Following this concept generation phase, I conducted a series of PSD studies with an additional twelve students and eight teachers. Study sessions lasted approximately 60 minutes. In each session, storyboards were presented in randomized order. Participants were asked to read each storyboard and to describe their initial reactions immediately after reading each one. An interviewer asked follow-up and clarification questions as needed. Participants were then asked to provide an overall summary rating of the depicted technology concept as "mostly positive (I would probably want this feature in my classroom)", "mostly negative (I would probably not want this ...)", or "neutral" (cf. Dillahunt et al., 2018). After participants rated each concept, they were asked to elaborate on their reasons for this rating. Before moving on to the next concept, participants were shown notes on reactions to a given concept, thus far, from other stakeholders. Participants were prompted to share their thoughts on perspectives in conflict with their own.

	S1	S2	S3	S4	T1	T2	T3	T4	S5	S6	S7	T5	S8	T6	T7	S9	T8	S10	S11	S12	Teacher avg.	Student avg.
[A.1] Ranking Students by Need for Teacher Help	0	1	0	1	1	1	0	1	1	-1	1	1	1	1	1	0	1	0	1	1	0.88	0.50
[A.2] Explaining Ranking of Students	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	1	0	0	1	0.13	0.08
[B] Suggesting Which Students to Help and How to Help	0	1	0	0	1	1	-1	1	1	0	0	1	0	1	1	0	1	0	0	1	0.75	0.25
[C] Helping Teachers Mediate between Stu. and Student Models	0	1	1	-1	-1	0	1	-1	1	0	0	-1	1	1	-1	1	-1	-1	0	1	-0.38	0.33
[D] Predicting Time to Mastery to Support Teacher Scheduling	0	1	0	0	1	1	0	1	1	0	1	1	0	-1	1	0	1	0	0	1	0.63	0.33
[E.1] Alerting Teachers to Student Frustration, Misbehavior, ...	0	-1	-1	0	1	1	0	1	1	0	1	1	-1	1	1	1	1	1	-1	1	0.88	0.08
[E.2] Providing Automated Motivational Prompts ...	-1	-1	-1	0	-1	1	1	-1	1	0	1	-1	-1	0	0	1	0	1	-1	1	-0.13	0.00
[E.3] Allowing Stu. to Hide (All) of their Analytics from Teachers			1	-1	-1	-1	-1	-1	0	0	-1	1	1	-1	-1	-1	-1	-1	0	-1	-0.75	-0.30
[E.4] Notifying Stu. When the System has Alerted their Teacher							1	-1	1	0	1	1	0	1	0	0	0	-1	-1	-1	0.33	-0.13
[E.5] Allowing Students to Hide Emotion-related Analytics ...													1	0	-1	0	1	0	1	1	0.00	0.60
[E.6] Asking Stu. Permission before Revealing (Some) Analytics ...														0	0	1	0	1	1	1	0.00	1.00
[F.1] "Invisible Hand Raises" and Teacher Reminders	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	-1	0	1	0.88	0.75
[F.2] Suggesting Peer Tutors to Support Teachers ...	1	1	-1	-1	1	1	1	1	0	1	1	1	-1	1	0	0	0	1	0	1	0.75	0.25
[G] Providing Teacher with Suggested "Conversation Starters" ...	0	1	0	0	-1	1	1	1	1	1	0	1	0	1	-1	0	1	0	0	1	0.50	0.33
[H.1] Enabling Students to Request Not to be Helped	0	1	1	1	1	1	0	-1	1	0	1	-1	1	1	1	1	1	1	1	1	0.38	0.83
[H.2] Enabling Stu. to Ask the Whole Class Anonymous Questions	1	1	1	0	1	0	0	0	1	0	1	-1	1	1	-1	1	1	1	0	0	0.13	0.67
[H.3] Student–System Joint Control Over Selection of Peer Tutors				1	1	1	1	0	1	1	1	1	0	0	1	1	0	1	1	1	0.63	0.89
[H.4] Showing Students Potential Peer Tutors' Skill Mastery												1	1	-1	-1	-1	0	0	-1	-1	-0.25	-0.50
[I.1] Real-time Positive Feedback on Teacher Explanations.	1	1	1	0	0	1	0	1	0	1	0	1	1	1	1	0	1	1	0	1	0.75	0.58
[I.2] Real-time Negative Feedback on Teacher Explanations.	1	1	1	0	1	1	1	1	0	1	0	1	1	1	1	0	1	1	0	1	1.00	0.58
[J] Notifying Teachers about Stu. they Have Not Visited Recently	0	1	1	1	1	1	0	-1	1	0	1	1	1	1	-1	1	1	1	1	1	0.38	0.83
[K] Listening in on Teacher Help-giving to Improve AI Tutor's Hints	0	0	-1	-1	-1	1	1	1	1	0	1	1	1	1	1	1	1	0	0	1	0.75	0.25
[L] Teacher-controlled Shared Displays to Foster Competition	1	1	1	1	1	1	1	1	0	1	-1	0	1	1	1	0	1	1	1	1	0.88	0.67
[M] Allowing Parents to Monitor their Child's Behavior During Class							1	-1	0	0	0	-1	-1	1	-1	-1	0	-1	-1	0	-0.17	-0.50

Figure 9-2. Matrix showing overall ratings for all 24 concepts. Columns show participants (in order of participation, from left to right), and rows show design concepts. Concepts generated by participants are highlighted in blue. Cell colors indicate ratings as follows: Red: negative; Green: positive; Yellow: neutral; Grey: concept did not yet exist. Average ratings among teachers and students are provided in the rightmost columns.

In addition, participants were encouraged to pause the speed dating process at any point, if they felt inspired to write down an idea for a new storyboard. Each time a participant generated a new idea for a storyboard, this storyboard was included in the set shown to the next participant.

[23] Please refer to *https://tinyurl.com/Complementarity-Supplement* for the full set of storyboards and more detailed participant demographics.

However, if a participant saw an existing storyboard that they felt captured the same concept as one they had generated, the new, "duplicate" storyboard was not shown to subsequent participants (similar to the notion of "synonymous" superpowers used in the "teacher superpowers" exercise in *Chapter 1*). In cases of disagreement between stakeholder groups, generating new storyboard ideas provided an opportunity for students and teachers to try to resolve these disagreements. For example, as shown in Figure 9-2, the generation of concepts E.3 through E.6 over time represents a kind of "negotiation" between teachers and students, around issues of student privacy, transparency, and control. This phase of the study yielded a total of seven new storyboards.

9.3 Results

In the following subsections, we discuss teachers' and students' top five most and least preferred design concepts, according to the average overall ratings among those who saw a given concept (Dillahunt et al., 2018). To analyze participant feedback regarding each concept, we worked through transcriptions of approximately 19 hours of audio to synthesize findings through interpretation sessions and affinity diagramming (Beyer & Holtzblatt, 1997; Hanington & Martin, 2012). High-level themes that emerged are briefly summarized below, organized by design concept.

The concepts that were *most preferred*, on average, within each stakeholder group are presented in *Section 9.3.1*, and the *least preferred* are in *Section 9.3.2*. Within each subsection, preferences among teachers are presented first, followed by student preferences and those shared between teachers and students. As in Dillahunt et al. (2018), the goal of this presentation format is not to contribute a set of "winning" and "losing" design concepts, but instead to discuss the *underlying reasons* behind some of teachers' and students' strongest positive and negative reactions.

9.3.1 Most preferred design concepts

Most preferred among teachers.

[I.2] *Real-time Feedback on Teacher Explanations.*
Consistent with findings from my prior design research (see *Chapters 1* and *5*), the most popular concept among teachers was a system that would provide them with constructive feedback, after helping a student, on the effectiveness of their own explanations. As one teacher (Teacher 7) explained,

> *"Usually our only chance to get [fast] feedback is, you ask [...] the kids [and] they just say, 'Oh, yeah, I get it,' when they don't really get it."*

[A.1] *Ranking Students by their Need for Teacher Help.*
Another popular concept among teachers was a system that would allow them to see, at a glance,

a visual ranking of which students most need the teacher's help at a given moment (see *Chapters 1, 4,* and *5*). Teacher 5 commented,

> *"Yeah. Welcome to teaching every day [...] trying to go to those kids that are [struggling] most."*

However, several other teachers emphasized that such a ranking would be much more useful if it took into account the *kind* and *extent* of teacher help that would likely be needed to address a particular student issue, along with a teacher's individual preferences for sequencing and prioritizing help among students. For example, Teacher 1 noted,

> *"If I could see how much time it would take [to help] I would start with the kids who I could get [moving again quickly] and then I'd spend more time with the other kids. [But] if it's a kid that I know is gonna get completely frustrated [...then I] wanna [go to] that kid first no matter what. So there are other factors involved."*

This concept was also generally well received by students. As Student 7 put it,

> *"sometimes you just can't ask [for help] because you don't even know what [you're struggling with], and so it would just [be] hard to explain it to the teacher."*

At the same time, as discussed below, multiple students expressed preferences for systems that can support students in recognizing when (and with what) they need to ask the teacher for help, rather than always having the system alert the teacher on their behalf (cf. Roll et al., 2011).

[E.1] *Alerting Teachers to Student Frustration, Misbehavior, or "Streaks".*
Consistent with my prior design findings (see *Chapters 1, 4,* and *5*) teachers were enthusiastic about a concept that would allow them to see real-time analytics about student frustration, misbehavior (e.g,. off-task behavior or gaming the system; see Baker et al., 2008), or high recent performance in the software (Pelánek, R., & Řihák, 2017). They felt that having access to this information could help them make more informed decisions about whom to help first and how best to help particular students (e.g., comforting a student or offering praise). Yet students reported finding aspects of this concept upsetting. While students generally liked the idea that the system would inform the teacher when they needed help, students often perceived real-time teacher alerts about emotions like frustration as *"really creepy"* (Student 9) and teacher alerts about misbehavior as *"basically the AI ratting out the child"* (Student 3).

[L] *Teacher-controlled Shared Displays to Foster Competition.*
Finally, a popular concept among teachers was a system that would allow them to transition the classroom between different "modes," to help regulate students' motivation (cf. Alavi, 2011; Alavi & Dillenbourg, 2012; Olsen, 2017). This system would allow teachers to switch the class into a "competitive mode," in which students would be shown a leaderboard of comparable classrooms in their school district and challenged to move their class to the top. Teachers expected that such a feature could work extremely well with some groups of students, while

backfiring and potentially serving to demotivate others. As such, teachers emphasized the importance of teacher control and discretion.

Most preferred among students.

[E.6] *Asking Students' Permission before Revealing (Some) Analytics to Teachers.*
In response to one of teachers' most preferred design concepts, [E.1], students generated multiple new storyboards that preserved the idea of real-time teacher alerts, but provided students with greater control over alert policies. One of these ideas emerged as the most popular design concept among students: a system that asks students' permission, on a case-by-case basis, before presenting certain kinds of information to the teacher on a student's behalf. Students and teachers were generally in agreement that an AI system should ask students' permission before alerting teachers about affective states, such as frustration. In this scenario, if a student opted not to share affective analytics with their teacher, the system might privately suggest other ways for students to regulate their own emotions. Interestingly, one student (Student 12) suggested that if a student opted to share their affect with the teacher, the system should also ask the student to specify:

> *"How do you want the teacher to react? [...] Help you [in person]? Help you on the computer?"*

This student noted that sometimes, they just want their teacher to *"know how I'm feeling,"* but do not actually want them to take action.

[H.3] *Student–System Joint Control Over Selection of Peer Tutors.*
Whereas teachers often expressed that they know which groups of their students will not work well together, this did not align with students' perceptions of their own teachers. In contrast to teacher-generated concepts where teachers and AI worked together to match peer tutors and tutees (cf. Olsen, 2017), the second most popular concept among students was a student-generated storyboard that gave students the final say over peer matching decisions. In this storyboard, the system sends struggling students a list of suggested peer tutors, based on these students' estimated tutoring abilities (cf. Walker, Rummel, & Koedinger, 2014) and knowledge of relevant skills. Students could then send help requests to a subset of peers from this list who they would feel comfortable working with. Those invited would then have the option to reject a certain number of requests. Some students suggested that it would also be useful to have the option to accept but delay another student's invitation if they want to help but do not want to disrupt their current flow.

[H.1] *Enabling Students to Request Not to be Helped.*
Another of the most popular concepts among students was a system that, upon detecting that a student seems to be unproductively persisting in the software (Beck & Gong, 2013; Kai et al., 2018), would notify the student to suggest that they try asking their teacher or classmates for help. The system would then only notify the teacher that the student is struggling if the student

both ignored this suggestion and remained stuck after a few minutes. By contrast, some teachers expressed that they would want the system to inform them immediately in such cases. For example, Teacher 5 commented:

> *"They shouldn't just get the option to keep working on their own, because honestly it hasn't been working*."

Some students and teachers suggested a compromise. For example, Teacher 7 suggested the teachers should at least be notified that such a request has been made, so that they can use their own discretion on a case-by-case basis:

> *"The AI should inform the teacher right away [...] that it suggested [asking for help] but the kid did something else."*

[J] *Notifying Teachers of Students they Have Not Visited Recently.*
Finally, a popular concept among students was a system that would track a teachers' movement during class and occasionally highlight students they may be neglecting (cf. An et al., 2018; Holstein et al., 2017a; Echeverria et al., 2018). Several students noted that even when they are doing well on their own, they feel motivated when their teacher remembers to check in with them. Most teachers responded positively to this concept. For example, Teacher 6 noted:

> *"Sometimes you forget about the kids that work well on their own, but sometimes those kids actually need help and don't raise their hands."*

However, a few teachers perceived this system as overstepping bounds and inappropriately judging them. For example, Teacher 4 responded:

> *"It's just too much in my business now. You better be quiet and give me a break."*

Most preferred among both teachers and students.

[F.1] *"Invisible Hand Raises" and Teacher Reminders.*
A concept popular with both teachers and students was a system that would allow students to privately request help from their teacher by triggering an "invisible hand raise" that only the teacher could see. To preserve privacy, this system would also allow teachers to silently acknowledge receipt of a help request. After a few minutes, the teacher would receive a light reminder if they had not yet helped a student in their queue, since as most students and teachers agreed *"usually teachers just forget"* (Student 1). Student 7 noted:

> *"I don't actually like asking questions since I'm supposed to be, like, 'the smart one' ...which I'm not. So I like the idea of being able to ask a question without [letting] others know."*

Similarly, teachers suspected that students would request help more often if they had access to such a feature (Holstein, Hong, et al., 2018; Schofield et al., 1994).

9.3.2 Least preferred design concepts

Least Preferred among Teachers

[C] *Helping Teachers Mediate between Students and their Student Models.*
To my surprise, although prior field research (see *Chapter 8*) had suggested that teachers might find it desirable to serve as "final judges" in cases where students wished to contest their student models (e.g., skill mastery estimates) (Bull & Kay, 2016), this was one of the least popular design concepts among teachers. Students generally viewed teacher-in-the-loop mediation desirable, since as Student 9 put it,

"I feel like the teacher knows the student better, not the software."

However, teachers generally did not view this as an efficient use of their time – viewing the tracking of student knowledge growth during a class session as a relative strength of AI tutors, compared with their own abilities. As Teacher 3 noted:

"I would just trust the tutor on this one. Having worked with Cognitive Tutor and other systems, I've learned to trust that it's pretty good at saying, 'Yeah, you haven't mastered it.' "

Furthermore, some teachers expressed concerns that from a student's perspective this concept may create undesirable conflict in the classroom by, as Teacher 1 put it:

"pitting one teacher against the other, if you consider the AI as a kind of teacher"

Several teachers instead suggested having the system assign a targeted quiz if a student wants to demonstrate knowledge of particular skills, rather than necessarily involving the teacher in resolving such "disputes" (cf. Bull & Kay, 2016).

Least Preferred among Students

[E.4] *Notifying Students When the System has Automatically Alerted their Teacher.*
A teacher-generated concept intended to provide students with greater transparency into the analytics being shared about them was among those least popular with students overall. Interestingly, while students valued having more control over the information visible to their teachers, they generally did not want greater transparency into aspects of the system that were outside of their control (cf. Lee & Baykal, 2017). As Student 10 put it,

"That would make me really anxious [...] If it's not asking students' [permission], I don't think they should know about it."

Least Preferred among both Teachers and Students

[E.3] *Allowing Students to Hide (All) of their Analytics from Teachers.*
The least popular concept among teachers, and the third least popular among students, was a privacy feature that would enable individual students to prevent their AI tutor from sharing real-time analytics with their teacher. This was a student-generated concept intended to mitigate the "creepiness" of having their teacher "surveil" students' activities in real-time. Yet as discussed in *Section 9.3.1*, overall students felt that it should only be possible for students to hide certain kinds of analytics (e.g., inferred emotional states). As Student 4 put it,

> *"if the AI sees a student is really, really struggling [...] I don't think there should be that blanket option."*

[H.4] *Showing Students Potential Peer Tutors' Skill Mastery.*
Consistent with prior research (e.g., Holstein, et al., 2019a), teachers and students responded negatively to a student-generated concept that made individual students' skill mastery visible to peers. While this concept was intended to help students make informed choices about whom to request as a peer tutor, most teachers and students perceived that the risk of teasing among students outweighed the potential benefits. Rather than sharing student skill mastery information, students and teachers suggested that the system should do the work of curating only viable peer matches, while still supporting student choice within these curated sets.

[M] *Allowing Parents to Monitor their Child's Behavior During Class.*
Somewhat surprisingly, Teacher 3 generated the concept of a remote monitoring system that would allow parents (cf. Broderick, O'Connor, Mulcahy, Heffernan, & Heffernan, 2011; Williamson, 2017) to

> *"see exactly what [their child is] doing at any moment in time [so that] if a kid's misbehaving, their parent can see the teacher's trying [their] best."*

While this concept resonated with one other teacher, student and teacher feedback on this concept generally revealed an attitude that to create a safe classroom environment, as Student 11 put it,

> *"we have to [be able to] trust that data from the classroom stays in the classroom."*

Teachers shared concerns that data from their classrooms might be interpreted out of context by administrators. As Teacher 5 shared:

> *"I don't ever want to be judged as a teacher [because] I couldn't make it to every student, if every kid's stuck that day. [But] using that data [as a teacher] is very useful."*

Students shared fears that, depending on the data shared, parents or even future employers might use classroom data against them.

[E.2] *Providing Automated Motivational Prompts to Frustrated Students.*
Finally, among the concepts least popular with both teachers and students was a system that automatically provides students with motivational prompts when it detects they are getting frustrated (see Baker et al., 2012; D'Mello, Picard, & Graesser, 2007; Woolf, Burleson, Arroyo, Dragon, Cooper, & Picard, 2009). Although teachers generally liked the idea of incorporating gamification elements to motivate students (cf. Long, Aman, & Aleven, 2015; Williamson, 2017), providing motivational messages in response to automatically detected affective states was perceived as, in Teacher 1's words, *"trying to [do] the teacher's job."* Teacher 5 emphasized that

> *"[this kind of] message being able to come from the teacher [usually] means a lot more than [coming from] computer programs for students"*

Similarly, several students indicated strongly that they would prefer these kinds of messages to come from an actual person, if at all (cf. Bartneck & Forlizzi, 2004; Huber, Lammer, Weiss, & Vincze, 2014). Student 8 said,

> *"I would just get more annoyed if the AI tried something like that."*

Similarly, Student 11 suggested:

> *"No emotional responses, please. That feels just [...] not genuine. If it's from the AI it should be more analytical, like just [stick to] facts."*

9.4 Conclusions

If new AI systems are to be well-received in K-12 classrooms, it is critical that they support the needs and respect the boundaries of both teachers and students. In this chapter, I have introduced "participatory speed dating" (PSD): a variant of the speed dating design method that involves multiple stakeholders in the iterative generation and evaluation of new technology concepts (see item 4 under *Summary of Contributions – "Novel design and prototyping methods"*). Using PSD, I sampled student and teacher feedback on 24 design concepts for systems that integrate human and AI instruction—an important but underexplored area of AIED research (see item 1 under *Summary of Contributions – "First broad design exploration of needs for real-time teacher analytics and orchestration support"*).

Overall, I found that teachers and students aligned on needs for "hidden" student–teacher communication channels during class, which enable students to signal help-need or other sensitive information without losing face to their peers. More broadly, both teachers and students expressed nuanced needs for student privacy in the classroom, where it is possible to have "too little," "too much," or the wrong forms of within-classroom privacy (cf. Mulligan & King, 2011; Wong & Mulligan, 2019). However, students and teachers did not always perceive the same needs. As discussed in *Section 9.3.1*, some of students' highest rated concepts related to privacy

and control were unpopular among teachers. Additional disagreements arose when teachers and students had different expectations of the roles of teachers versus AI agents and peer tutors in the classroom.

Interestingly, while students' expressed desires for transparency, privacy, and control over classroom AI systems extend beyond what is provided by existing systems (Broderick et al., 2011; Bull & Kay, 2016; Holstein et al., 2018b; Wetzel et al., 2018), these desires are *also* more nuanced than is commonly captured in theoretical work on risks and challenges in the design of classroom AI systems (e.g., Bulger, 2016; Watters, 2014; Williamson, 2016; 2017). For example, I found that while students were uncomfortable with AI systems sharing certain kinds of personal analytics with their teacher without permission (e.g., real-time alerts of student frustration), they rejected design concepts that grant students *full* control over these systems' sharing policies. These findings suggest an important role for empirical, design research approaches to complement critical and policy-oriented research on AI in education (cf. Lee & Baykal, 2017; Mulligan & King, 2011; Wong & Mulligan, 2019) (see item 1 under *Summary of Contributions – "First broad design exploration of needs for real-time teacher analytics and orchestration support"*).

In sum, the present work provides tools and and early insights to guide the design of more effective and desirable human–AI partnerships for K-12 education. Findings from the current study are further explored in the context of classrooms using Carnegie Learning's *MATHia* software in *Chapter 10*. Future research should further investigate student and teacher needs uncovered in the present work via rapid prototyping in live K-12 classrooms. While design research methods such as Participatory Speed Dating are critical in guiding the initial development of novel prototypes, many important insights surface only through deployment of functional systems in actual, social classroom contexts (Holstein et al., 2019a; Odom et al., 2012; Schofield et al., 1994). An exciting challenge for future research is to develop methods that extend the advantages of participatory design approaches (e.g., Mitchell, Ross, May, Sims, & Parker, 2016; Trischler et al., 2018; Walsh et al., 2013; Zhu et al., 2018) to later stages of the AIED and Learning Analytics design cycle (see *Chapter 5* for a discussion). Given the complexity of data-driven AI systems (Dove et al., 2017; Holstein et al., 2019a; Holstein, Wortman, Vaughan, et al., 2019; Zhu et al., 2018), fundamentally new kinds of design and prototyping methods may be needed to enable non-technical stakeholders to remain meaningfully involved in shaping such systems, even as prototypes achieve higher fidelity.

Chapter 10

Towards Generalizing across Tutoring Systems: Piloting Lumilo in Carnegie Learning Classrooms

10.1 Background and Motivation

Although the concept of real-time, wearable teacher augmentation is intended as a general approach, *Lumilo* has thus far been used with a single tutoring system, *Lynnette* (see Figure 10-1; Holstein et al., 2018a; 2018b; 2019a; Long, Holstein, & Aleven, 2018; Waalkens, Aleven, & Taatgen, 2013), running within a particular infrastructure for ITS authoring and deployment (CT+A, an extension of the CTAT/TutorShop architecture (Holstein, Yu, et al., 2018)). In this chapter, I begin to explore how RWTA might be generalized for use with a broader range of tutoring systems and classroom contexts. Specifically, I explore what challenges arise in adapting the design of *Lumilo* to work with an ITS for which it was not originally designed: Carnegie Learning's *MATHia* system (see Figure 10-2).

Despite growing awareness in the learning analytics, AI in education, and educational data mining communities of the difficulty of transferring student modeling methods and learning analytics tools across different educational software systems, it remains rare to see demonstrations of generalization across systems, or explorations of challenges that arise when trying to generalize across systems (Baker, 2019; Holstein, Yu, et al., 2018; but see: Paquette, Baker, de Carvalho, & Ocumpaugh, 2015; Paquette et al., 2018). Student modeling and learning analytics methods – and tools that utilize these methods – are often designed and developed for use with *specific* educational software systems. However, these methods and tools do not always generalize for use with other software systems or classroom contexts (Karumbaiah, Ocumpaugh, & Baker, 2019; Ocumpaugh, et al., 2014; Paquette et al., 2015; 2018).

Generalization challenges may arise not only when transferring designs between different classes of educational software, but even when transferring designs across relatively similar systems. For example, when transferring a machine-learned detector of a particular student behavior, such as gaming-the-system, from one ITS (the system for which it was originally designed) for use with a different ITS, the detector may be significantly less accurate in the new system. Detectors may fail to generalize even when transferring between systems teaching similar content, such as two different tutors for equation solving in middle school algebra (Paquette et al., 2015; 2018). Furthermore, even if a set of detectors or analytics generalizes well across ITSs, the design of an associated teacher-facing tool that relies upon these detectors/analytics may fail to generalize (e.g., since the nuances of different ITSs' designs may raise different orchestration challenges for the teacher).

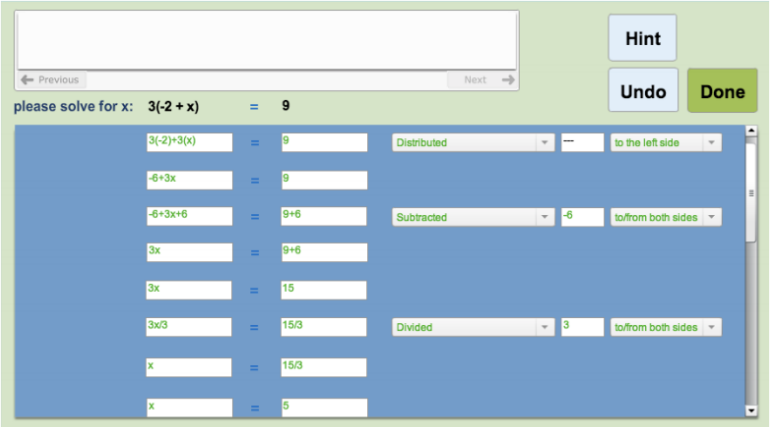

Figure 10-1. Screenshot of a problem in *Lynnette*.

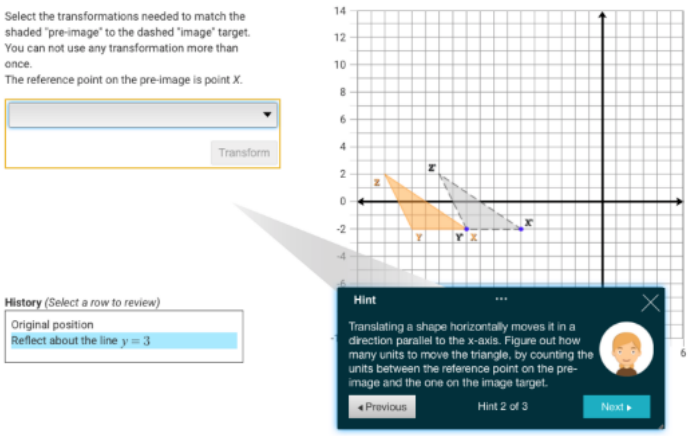

Figure 10-2. Screenshot of a problem in *MATHia*.

In the case of teacher-facing analytics tools, interface design and algorithmic design considerations intersect (Baumer, 2017; Dennerlein, et al., 2018; Holstein, Hong, et al., 2018; Holstein et al., 2019a; Zhu, et al., 2018). For example, a given interface design may only behave in a reasonable way under certain assumptions about algorithmic behavior, which may break down when switching contexts – the true positive rate for a given detector may vary across different ITSs, causing a teacher tool to provide a manageable number of notifications in classrooms that use one ITS, but an overwhelming number of notifications in classrooms that use a different ITS (Holstein, Hong, et al., 2018; Holstein et al., 2019a).

MATHia raises a number of potential design challenges for teacher support tools that are not encountered with *Lynnette* (see Table 10-1). For example, whereas *Lynnette* covers curricular content on the scale of days or weeks, *MATHia* covers content on the scale of months or years. This may mean that *MATHia* classrooms have the potential to be significantly less synchronized than *Lynnette* classrooms (i.e., students may be more "spread out" across the curriculum at any

given moment; see Ritter et al., 2016a for a discussion). Thus, forms of co-orchestration support that work well in *Lynnette* classrooms (e.g., system support for class-level or group-level teacher interventions that assume a certain degree of classroom synchrony) may be less useful in *MATHia* classrooms. As discussed in *Chapters 1* and *9*, prior work suggests that teachers often manually "override" mastery learning in ITS classrooms in order to increase synchrony and keep classes "on schedule", and there is reason to believe that such manual overrides may be harmful to students' learning (Ritter et al., 2016a). The teachers I have worked with tend to be aware that this behavior may be harmful to students, yet they persist in this behavior both due to external, systemic pressures to keep classes on schedule, and because less synchronized classrooms pose many orchestration challenges for teachers (see *Chapters 1* and *9*, and Holstein et al., 2017b; 2019a; 2019b).

More broadly, while the specific choices of real-time analytics (e.g., particular detectors of help avoidance, gaming-the-system, and unproductive persistence) and parameter settings (e.g., alert thresholds) in *Lumilo* have been iteratively shaped with teachers, much of this shaping occurred in the context of prototyping studies using *Lynnette*. It remains to be seen how well these analytics generalize for use in *MATHia* classrooms (cf. Ocumpaugh, et al., 2014; Paquette, et al., 2015; 2018).

Table 10-1. Anticipated challenges in adapting the design of *Lumilo* to work with *MATHia*.

Lynnette	MATHia
Covers on the order of **days - weeks** of curricular content, and is used over relatively short timespans.	Covers on the order of **months - years** of curricular content, and is often used continuously throughout the school year.
Includes a **smaller range of problem types** (all tutor problems follow a similar format: line-by-line equation solving).	Includes a **broader range of problem types** (tutor problems span a wide range of formats and topics).
Problems include **less context** that the teacher needs to catch up on, in the moment (students are presented with an algebraic equation and asked to solve it).	Some problem types involve **substantial context** (e.g., a given problem may present students with detailed, multi-step word problems and interactive graphical representations).
Lumilo's real-time analytics and parameter settings have been **tuned** for *Lynnette* classrooms (see *Chapters 4 through 6*).	*Lumilo*'s real-time analytics and parameter settings have **not been tuned** for *MATHia* classrooms.

Unlike *Lynnette*, where all tutor problems follow a similar format (line-by-line equation solving, as shown in Figure 10-1), *MATHia* includes a much broader range of problem types, such as detailed word problems and interactive graphing problems. This broader range may necessitate

the design of new ways for *Lumilo* to efficiently debrief teachers on students' current activities and areas of struggle, while also providing necessary context for interpretation. This may be especially important given that students in *MATHia* classrooms may be working on a very wide range of topics and problem types at any given time – requiring the teacher to frequently context-switch when moving from student to student (as discussed in *Chapter 4*).

While Table 10-1 presents my *initial, broad hypotheses* about factors that may require *Lumilo*'s design to be adapted for a *MATHia* context, it was not clear a priori whether all of these would present equal challenges, nor what exactly may be needed to overcome the generalization challenges that do arise.

Thus, in this chapter, I present a classroom pilot and technology probe study (Hutchinson et al., 2003) conducted in collaboration with Carnegie Learning – using an initial, functional prototype of a *Lumilo–MATHia* integration – that aims at understanding the extent to which the current design of *Lumilo* may generalize (or fail to generalize) to classrooms that use *MATHia*. I identify areas of alignment and disconnect with my prior design research findings, highlighting (1) findings that confirm needs from my prior research, and which were represented in the design of *Lumilo* previously used with *Lynnette*; (2) findings that confirm needs from my prior research, yet were *not* represented in the version of *Lumilo* previously used with *Lynnette*; and (3) findings that point to different design requirements for real-time teacher augmentation in *MATHia* versus *Lynnette* classrooms. Based on findings from this study, in *Section 10.4*, I present an updated version of Table 10-1, pointing to specific directions for future design.

This work represents a rare exploration of challenges that arise in adapting the interface and algorithm design of a learning analytics tool to work with an educational software system for which it was not originally designed (see item 2 under *Summary of Contributions – First design exploration and prototypes of wearable, heads-up displays to support orchestration of personalized classrooms*).

10.2 Methods

As a first step towards generalizing real-time, wearable teacher augmentation, I developed a new, minimal version of *Lumilo* that would work with Carnegie Learning's *MATHia* software. Our newly expanded research team – consisting of researchers at both Carnegie Learning and Carnegie Mellon University – first worked to develop the technical infrastructure required to integrate *Lumilo* and *MATHia*. In many ways, the extended infrastructure that emerged from these efforts resembled the extended CTAT/TutorShop Analytics (CT+A) architecture presented in *Chapter 3* (Holstein, Yu, et al., 2018). However, building atop Carnegie Learning's existing, company-scale technical infrastructure raised many new challenges that required the Carnegie Learning team to diverge from CT+A's architectural design.

To study issues around the transferability of our existing designs and design knowledge to classroom contexts that use *MATHia*, my collaborators and I conducted a technology probe study in live middle school classrooms. Hutchinson et al. conceptualize a technology probe as an instrument used to "combine the social science goal of collecting information about the use and the users of the technology in a real-world setting, the engineering goal of field-testing the technology, and the design goal of inspiring users and designers to think of new kinds of technology to support their needs" (Hutchinson et al., 2003; Quintana et al., 2016). Accordingly, this classroom study served as an opportunity to (1) conduct *technical field tests* of an early *Lumilo–MATHia* integration (a particularly important goal for the Carnegie Learning team), (2) conduct *classroom observations* of a teacher's use of *Lumilo* in *MATHia* classrooms (a key goal for both teams), and (3) to provide teachers with the necessary context to provide rich, experientially-grounded *design feedback and ideas* for future versions of *Lumilo–MATHia* (the primary focus of the present chapter).

To balance these three goals, without devoting too much time upfront to implementing features that may not generalize to a *MATHia* context, I worked with the Carnegie Learning team to develop a reduced-functionality version of *Lumilo* for use with *MATHia.* The three main differences between the version of *Lumilo* used in prior classroom studies (with *Lynnette*) and the initial prototype of the *Lumilo–MATHia* integration are illustrated in Figures 10-3 and 10-4. First, the concreteness and specificity of information presented in the "deep dive" screens was heavily reduced; second, only a subset of *Lumilo's* original real-time indicators were re-implemented; and third, the granularity/specificity of information presented with the included indicators was reduced.

Features of the original *Lumilo* prototype were prioritized for re-implementation based, first and foremost, on the implementation time and effort that would be required prior to running the (pre-scheduled) classroom study. Within these practical constraints, features were then prioritized based on a combination of 1) the degree to which teachers reported favoring particular features or combinations of features in prior classroom studies, and 2) the frequency with which teachers were observed making use of particular features in prior classroom studies.

We conducted study sessions with four teachers and 138 students, across five 6th-grade math classrooms. These teachers and students had already been using *MATHia* in the classroom for a full year by the time we introduced *Lumilo*. School- and teacher-level demographic information is provided in Table 10-2.

Each class worked with *MATHia* for two class periods, for a total of 160 minutes (including student login time, announcements at the beginning of a class session, etc). Students in each class worked on two *MATHia* sections (or "workspaces"): one for circle geometry and one for working with ratios. Notably, neither of these topics are covered by *Lynnette*, which is focused on algebraic equation solving. As in prior classroom studies with *Lumilo* and *Lynnette*, teachers

participated in a brief (approximately 30 minute) training before using *Lumilo–MATHia* in their classrooms. However, unlike prior studies, this training was conducted without the use of Replay Enactments (given technical challenges in implementing screen-based replay functionality for *MATHia*). This was replaced with a brief familiarization phase for each teacher, conducted in their first few minutes wearing the glasses in a live classroom.

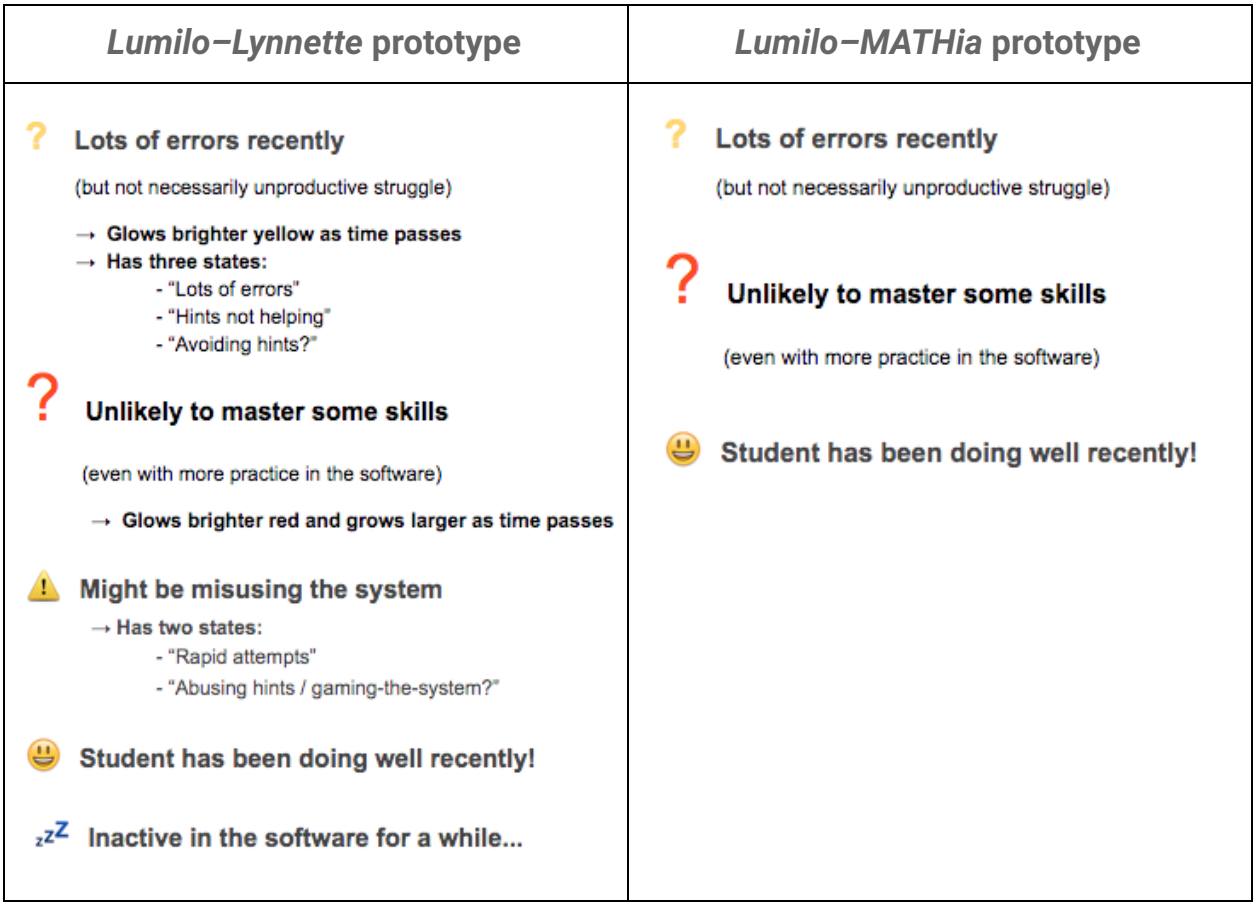

Figure 10-3. Illustration of key differences (regarding information visible *at a glance*) between the version of *Lumilo* previously deployed in in-vivo classroom experiments with *Lynnette* (left), and the newly-created, minimal version used with *MATHia* in the current study (right). The *Lumilo–MATHia* prototype has fewer indicators, and provides less granular information for each indicator.

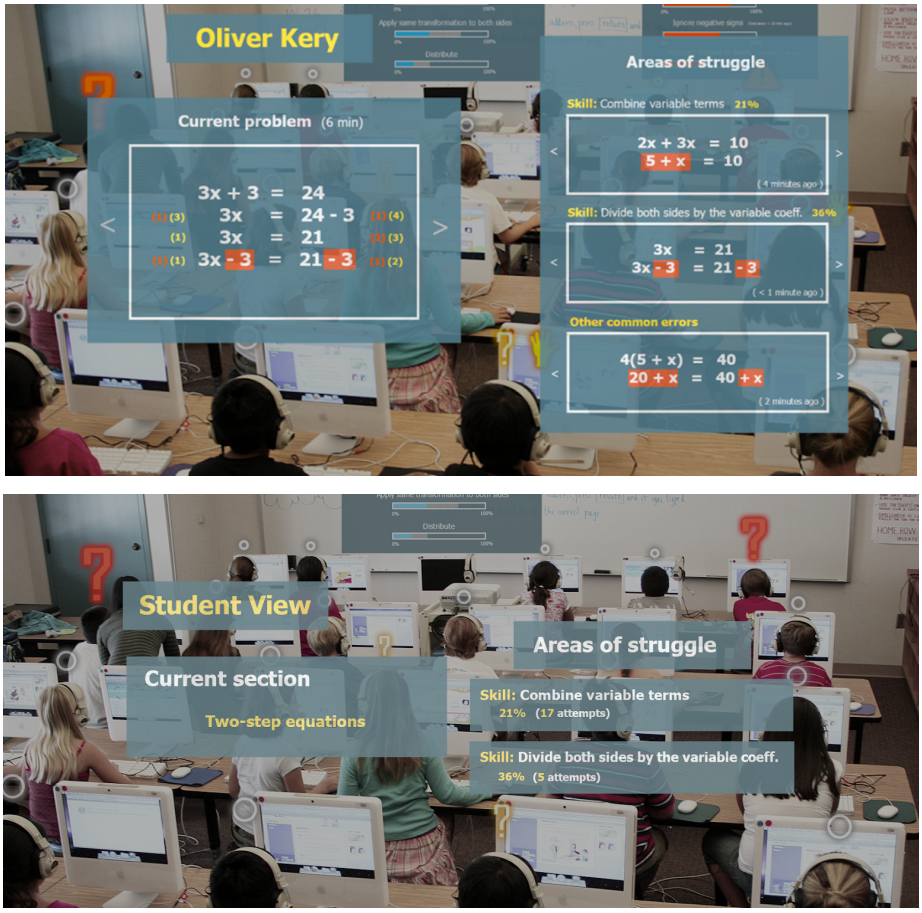

Figure 10-4. Illustration of key differences in the design of the "deep dive" screens between the previous version of *Lumilo* (top), and the minimal *Lumilo–MATHia* prototype used in the current study (bottom). The *Lumilo–MATHia* prototype displays only the current broad topic (i.e., the name of a section, or collection of problems) that a student is working on (e.g., "Two-step equations"), rather than showing an annotated live view of the student's activities on their current problem. In addition, this prototype provides high-level analytics about a student's lowest skills (their skill mastery percentage and number of attempts), but does not provide concrete examples of recent errors that student has made on a skill. Note that due to privacy concerns in the school where we piloted, student names were not made visible to this version of *Lumilo–MATHia*. Thus, the label "Student View" was displayed in place of student names within the deep dive screens. Note: student names shown in this figure are fabricated.

Table 10-2. Demographic information for participating schools.

School	Region	Free/reduced price lunch	# of participating teachers	# of participating teachers with ≤ 2 years' experience
O	Rural	51%	4	0

Given the primary goals of this study, we did not administer pre- and post-assessments for students. Rather, students began working with *MATHia* from the start of a class session. During class, members of our research team conducted informal classroom observations. Teachers were asked to help their students during each class session, but were also invited to visit researchers in the back of the classroom at any time during the study, to ask questions or provide feedback on the experience of using the *Lumilo–MATHia* prototype.

Immediately following the final class session, researchers conducted a 75-minute post-interview and workshop session with all four teachers. Teachers were first reminded that the version of *Lumilo–MATHia* that they used in their classrooms is a rough prototype of a future technology, and that the design remains extremely malleable. They were then asked to share general reflections on the experience with the group, as well as initial thoughts on ways future versions could be improved. Teachers were then asked a sequence of increasingly targeted questions. Teachers were asked whether they found the real-time indicators presented by *Lumilo–MATHia* helpful. If so, a researcher further probed to understand how teachers had used particular indicators during a class session. Teachers were then asked to generate ideas for other kinds of information that may have been helpful to have in real-time, but which were not represented in the current prototype. After teachers provided their initial responses, they were shown ideas that had emerged in our prior design research with teachers and students (including, but not limited to, indicators presented by earlier versions of *Lumilo*; see *Chapters 1*, *4*, *5*, *8*, and *9*). This process was then repeated for classroom orchestration functionality more broadly, moving beyond the framing of real-time information that is presented to the teacher. Finally, teachers were presented with storyboards from the participatory speed dating study reported in *Chapter 9*, as an additional means of identifying any key needs uncovered in our prior research that may not generalize to *MATHia* classrooms.

10.3 Findings

To analyze data from classroom observations and feedback sessions, I worked through classroom observation notes and 75 minutes of audio to identify areas of alignment or disconnect with my prior design research findings. Key findings and reflections are briefly summarized below, with an emphasis on 1) findings that confirm needs from my prior research, and were represented in the design of *Lumilo* previously used with *Lynnette*; 2) findings that confirm needs from my prior research, yet were *not* represented in the version of *Lumilo* previously used with *Lynnette*; and 3) findings that point to different design requirements for real-time teacher augmentation in *MATHia* versus *Lynnette* classrooms.

Notes on generalization challenges our team observed even before entering classrooms (i.e., during development and log-replay-based testing in the lab) are discussed first, followed by

insights from classroom observations and the after class interview/workshop session with teachers.

Observations before entering classrooms

[B1] *Poor transferability of Lumilo's "unproductive persistence" detector for real-time teacher support*

During development and internal, iterative testing – using replayed classroom log data collected from a range of classrooms that use *MATHia* – we observed that *Lumilo* tended to detect unproductive persistence much later in *MATHia* classrooms than in *Lynnette* classrooms. In a *Lynnette* classroom, *Lumilo* would typically alert a teacher about unproductive persistence on a particular skill within around 10 minutes of the student's first practice opportunity on that skill. However, in a *MATHia* classroom, a teacher might not be alerted for closer to 40 minutes, or the teacher might not be alerted at all, depending on the skill in question.

This is due to the implementation of *Lumilo's* original detector of unproductive persistence (defined as a phenomenon in which an AI tutor is *failing to help the student learn*, on one or more skills; see Holstein, 2018), which relies on the operationalization proposed by Beck & Gong (Beck & Gong, 2013; Kai et al., 2018; Zhang et al., 2019). Under this operationalization, a student is considered to be unproductively persisting on a skill if they fail to reach a mastery criterion (e.g, getting M steps correct in a row on steps tagged with that skill, or achieving a certain probability of mastery on that skill under a probabilistic student model such as Bayesian Knowledge Tracing) within the first N attempts (where the parameter N is conventionally set to 10; see Beck & Gong, 2013; Kai et al., 2018; Zhang et al., 2019). For example, compared with the *MATHia* units used in the current study*, Lynnette* provides many more practice opportunities for a given skill within a relatively short time frame. Thus, unproductive persistence on a given skill can potentially be detected earlier in *Lynnette* than in *MATHia*, under *Lumilo*'s original detector algorithm, since students tend to reach N=10 practice opportunities sooner. Furthermore, *MATHia*'s existing problem selection policy is designed such that the software may in certain cases "give up" on tutoring a particular skill before a student has mastered the skill *and* before a student has reached N=10 practice opportunities (Zhang et al., 2019), in favor of moving the student on to other material.

This policy of "giving up" on tutoring a given skill and instead moving on to others can be viewed as the tutoring system recognizing that it may have reached its own pedagogical limitations for the given skill and student – sparing the student from prolonged, unproductive practice with this skill (Beck & Gong, 2013; Holstein, 2018; Holstein, Hong, et al., 2018; Kai et al., 2018; Käser et al., 2016; Zhang et al., 2019). However, when a teacher, peers, or other educational resources are available – as is often the case when ITSs are used in classrooms – it may be desirable for the tutoring system and/or the student to instead draw upon these external

resources for help in such scenarios (Beck & Gong, 2013; Holstein, 2018; Holstein et al., 2017b; 2019a; 2019b; Holstein, Hong et al., 2018; Kai et al., 2018; Zhang et al., 2019).

A consequence of these differences between *Lynnette* and *MATHia* is that in *MATHia* classrooms, teachers may tend to be alerted to students who need their help later than would be ideal for a timely intervention. Indeed, even in *Lynnette* classrooms, teachers expressed desires for earlier detection of unproductive persistence (see *Chapters 5* and *8*, and Holstein, Hong et al., 2018; Holstein et al., 2019a; Zhang et al., 2019). As a temporary solution (given limited time in preparing for the current study) we adjusted the value of the parameter N from 10 to 6 – effectively increasing the potential for early detection, while simultaneously increasing the potential for false positives. However, longer term, it may be best to move away from the use of a Beck & Gong family detector of unproductive persistence (Kai et al., 2018; Zhang et al., 2019) to a detector that is more tightly coupled with and tailored to *MATHia*'s problem selection policy and curriculum design. This could, for example, ensure that it is impossible, or at least unlikely, for the tutoring system to "give up" on tutoring a given skill without first requesting the teacher's assistance. Our collaborators at Carnegie Learning are currently exploring this route as part of our ongoing research.

[B2] *Transferability of Lumilo's "hint abuse" detector*

While developing the initial prototype of *Lumilo–MATHia*, it became clear that directly transferring the detector of "hint abuse" from *Lumilo–Lynnette*, without modification, may not be a viable option. Unlike *Lynnette*, *MATHia* includes built-in, student-facing functionality designed to discourage hint abuse: a delay is enforced in between hint requests to discourage students from rapidly clicking through hints. However, the behavior of rapidly clicking through hints (e.g., milliseconds elapsed between consecutive hint requests) is a feature that *Lumilo*'s detector of hint abuse relies upon (Aleven et al., 2016). Thus, *MATHia*'s hint request "speed bump" feature, designed to discourage hint abuse, may also serve to hinder detection of students who persist in abusing hints (by simply waiting out the enforced delays, and immediately skipping ahead to the next hint) – at least if the existing detector is used without modification. As discussed below, real-time detection of hint abuse remains a key teacher need in *MATHia* classrooms, so the question of whether and how the Help Model (Aleven et al., 2016) may need to be adapted for use with *MATHia* is an important challenge for future work on this project.

Observations in the classroom study and post-workshop session that align with prior research and were addressed in previously deployed versions of *Lumilo*

Many of our findings from prior design research were validated in the current study, in the context of *MATHia* classrooms. Given that these and similar findings have been discussed in previous chapters, this and the following subsection presents key needs that re-emerged during the current study very briefly.

[A1] *Preference for heads-up, private, spatially distributed displays*

Consistent with prior findings from our group and others (e.g., An et al., 2019; d'Anjou et al., 2019; Holstein et al., 2017b; 2019a; Holstein, Hong, et al., 2018), teachers expressed a strong preference for heads-up, private, spatially distributed displays, compared with handheld interfaces or other wearables such as smart watches. However, as before, teachers emphasized that a version of the tool intended for regular use would need to be lighter-weight and less bulky than the existing HoloLens 1 based prototype (Holstein et al., 2019; Holstein, Hong, et al., 2018). The following snippet of teachers' conversation during the post-workshop presents a sample of teachers' reflections on the experience:

> **Teacher 1:** *"...I was impressed with how easy it was to just kinda scan and see, like, 'Okay, these people are smiley faces, these people are question marks.' I mean, if they were lightweight glasses... it was really nice to have the whole class directly in front of your... instead of having to be carrying around a screen..."*
>
> **Teacher 2:** *"Because it's right above your head. Like, even if it was a list of kids [...] you'd still have to go and kind of search for that kid, where here, it's right in front of you like, 'Oh, this kid has the red question mark.' "*
>
> **Teacher 1:** *"Yeah. I like that just being able to stand there and still see everything, like, not have to look down at something and then look back up. I was just still seeing their faces, I was still seeing their computers, I was still seeing if somebody walked into the room, but I could still help them at the same time."*
>
> **Teacher 4:** "*Yeah, and then you accidentally set it down somewhere and then you walked away and then you were like, 'Where was that,' right? [With this] it's always with you."*
>
> **Teacher 1:** "*Yeah, it's just there. You just scan and... If somebody calls and you set down your iPad, then you forget that you were looking at it and you go back to something. If the glasses are on, you're automatically like, 'Hey. Oh, that's right, you have a red question mark. I need to go over there.' "*

[A2] *Very useful to have positive, not just negative, information in real-time*

Consistent with prior findings from our group and others (e.g., Holstein et al., 2019a; Holstein, Hong, et al., 2018; Martinez-Maldonado et al., 2014), teachers noted that they appreciated being able to see positive information about their students – such as the smiley faces that appear when a student has recently been on a "streak" – rather than only seeing negative information. These teachers shared that mobile- and tablet-based real-time analytics tools they had used previously tended to present only negative information (e.g., about students who seemed to be inactive or making a lot of errors). As Teacher 1 noted:

"...it was nice for that positive feedback, too, for those kids to, like, get some sort of comment to say, 'You're doing a good job right now.' Because we would typically not give that comment out because we would just be going to all the kids that were having issues."

[A3] *Desires for real-time detection of system misuse and off-task behavior*

In line with our prior findings (e.g., Holstein et al., 2019a; 2019b; Holstein, Hong, et al., 2018), teachers expressed desires for real-time indicators about student misuse of the software (e.g., hint abuse or gaming the system) and off-task behavior. These are features that were removed from the prototype of *Lumilo–MATHia* that these teachers had used in the classroom.

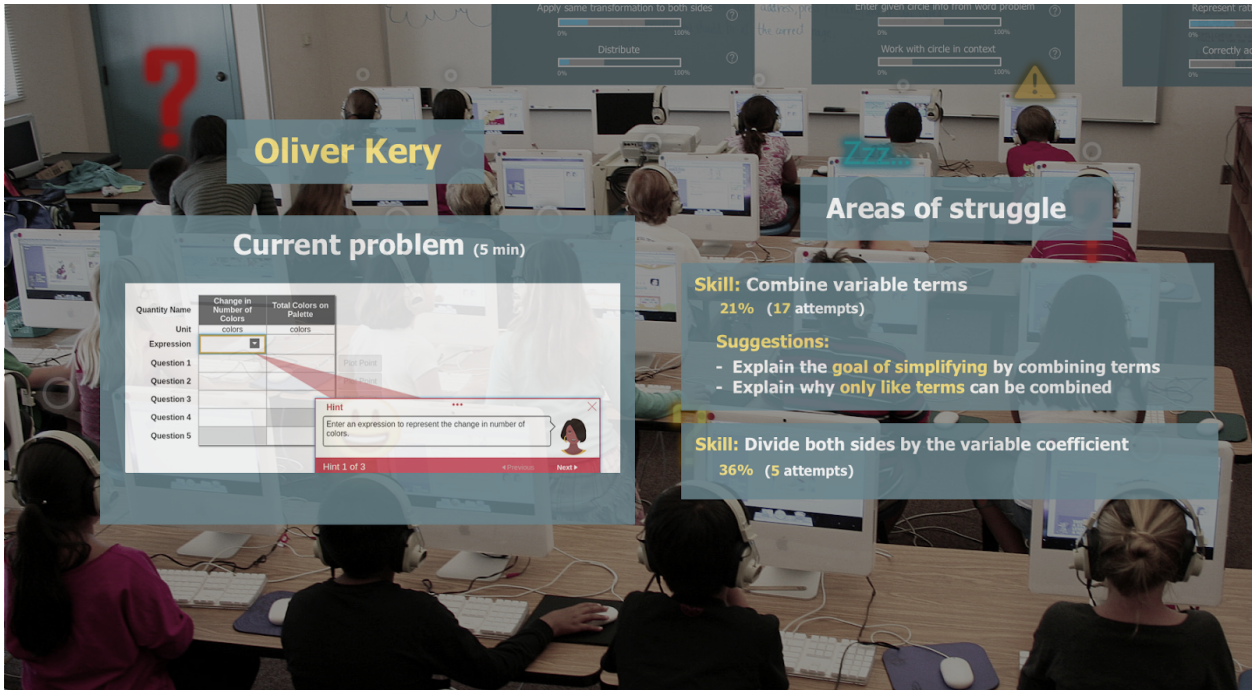

Figure 10-5. Early illustration of *just one possible re-design* for the deep-dive screens in a future version of *Lumilo–MATHia* (a thorough exploration of how such features can best be designed is left for future work). The teacher can peek at an individual student's current activities via a live feed of their tutor interface in the "Current problem" screen, and can view areas where a given student is struggling in the "Areas of struggle" screen. This screen provides specific action recommendations or "suggestions" for how to help the student in cases where these are available. Note: student names shown in this figure are fabricated.

[A4] *Desire for ability to remotely "peek" at student screens in real-time*

Consistent with prior findings from our group and others (e.g., Holstein et al., 2019a; 2019b; Holstein, Hong, et al., 2018; VanLehn et al., 2019; Wetzel et al., 2018), all teachers in the post-workshop expressed a strong desire in the ability to remotely "peek" at a live feed of

students' screens from any location in the classroom (as illustrated in Figure 10-5) – a feature that was removed from the prototype version of *Lumilo–MATHia* that they had used in the classroom.

[A5] *Desire for specific, concrete, and rapidly actionable information about student difficulties*
Consistent with previous findings from our group and others (e.g., Bull & Kay, 2016; Holstein et al., 2019a; Holstein, Hong, et al., 2018), teachers expressed a desire to see specific, concrete, and rapidly actionable information of student difficulties during a class session (e.g., raw examples of student errors). However, teachers were concerned that for certain problem types in *MATHia*, it may be challenging to design brief, readily actionable representations of student errors. During the post-workshop, teachers discussed the possibility of presenting real-time action recommendations in these cases (as illustrated in Figure 10-5). Such action recommendations would essentially acknowledge that even experienced teachers can be situationally impaired due to factors such as heavy time pressure (Sears, Lin, Jacko, & Xiao, 2003; Wobbrock, Kane, Gajos, Harada, & Froehlich, 2011). These action recommendations would offload the tasks of interpreting student errors and deciding upon appropriate responses (e.g., by teacher-sourcing context-dependent action recommendations from experienced teachers who are not under such time pressure; cf. Heffernan et al., 2016; Wang, Talluri, Rosé, & Koedinger, 2019).

[A6] *Desires for greater support in prioritization (while still enabling teacher discretion)*
In line with findings from our prior research (e.g., Holstein et al., 2017b; 2019a; 2019b; Holstein, Hong, et al., 2018), teachers expressed various needs for greater support in prioritizing help among students – especially in cases where many students may need their help at the same time. Given that the detector of unproductive persistence used in this study achieved earlier detection at the cost of presenting teachers with more false positives, teachers reported feeling overwhelmed at times by the number of students who appeared to need their immediate attention (cf. Holstein et al., 2019b; Holstein, Hong, et al., 2018).

During the post-workshop, teachers suggested that to help with prioritization, it would have helped to be able to see, at a glance, how long an indicator had been active for a given student (cf. Alavi & Dillenbourg, 2012; Holstein et al., 2019a; 2019b; Holstein, Hong, et al., 2018). This is a feature that existed in prior versions of *Lumilo* used in *Lynnette* classrooms, but was removed from the rough prototype version of *Lumilo–MATHia* that these teachers had used in these *MATHia* classrooms. As teacher 4 noted:

> *"I guess I'd kinda like to see like a timer or something, though, not just the red [question mark], but [to see that the student has] been here for four minutes or for eight minutes..."*

In line with findings from Holstein et al. (2019a; 2019b), these teachers also suggested that the task of prioritizing help among students would be made easier if the tool were able to indicate, at a glance, what kind of help each student needed from the teacher (rather than only indicating

what a student may be struggling with). This notion of organizing information based on what a teacher should do to help, rather than based on specific difficulties students are facing, connects with teachers' expressed needs for real-time *action recommendations* (discussed further in [D3]).

Observations in the classroom study and post-workshop session that align with prior research, but were *not* addressed in previously deployed versions of *Lumilo*

We also observed teacher needs, in both the classroom study and the post-workshop session, that re-emerged from our prior research, yet which are not addressed (at least, not directly) in previously deployed versions of *Lumilo*.

[P1] *Desire for private teacher–student communication channels*
In line with findings from our prior research (e.g., Holstein et al., 2019a; 2019b; Holstein, Hong, et al., 2018), teachers expressed strong interest in having private teacher–student communication channels during a class session such as "invisible hand raises," and the ability for teachers to privately acknowledge these invisible hand raises (as illustrated in Figure 10-6, and discussed in Holstein et al., 2019a; 2019b; Holstein, Hong, et al., 2018).

[P2] *Desire for real-time feedback on teacher explanations*
In line with findings from our prior research (e.g., Holstein et al., 2017b; 2018b; 2019a; 2019b; Holstein, Hong, et al., 2018), teachers expressed interest in receiving near real-time feedback on the effectiveness of their own explanations during a class session.

[P3] *Desires for additional support in monitoring and regulating student motivation*
In line with findings from our prior research (e.g., Holstein et al., 2017a; 2017b; 2018a; 2018b; 2019a; 2019b; Holstein, Hong, et al., 2018), teachers expressed interest in receiving more real-time support in monitoring and regulating *student motivation* than is offered by existing versions of either *Lumilo–Lynnette* or *Lumilo–MATHia.* For example, as in Holstein et al. (2017b; 2019a) and Holstein, Hong, et al., 2018, to help inform an appropriate intervention, teachers were interested in the ability to distinguish whether a given student has been making consistent errors while putting in reasonable effort, or whether these errors may simply result from carelessness. In addition, as in Holstein et al. (2019b), teachers were interested in mechanisms that would allow them to dynamically switch the class between different social "modes" (e.g., between individual work, whole-class competitions, team-based competitions, or whole-class collaborative work), in order to enhance student engagement.

Figure 10-6. Early illustration of *just one possible re-design* for *Lumilo–MATHia*'s student-level indicators (a thorough exploration of how such features can best be designed is left for future work). In this image, a student has pressed an "Ask teacher for help" button in the tutoring software, triggering an "invisible hand raise" that only the teacher is able to see. When the teacher gazes at the student's main indicator (which is in the default state) or the student's invisible raised hand icon, a menu of additional options appears. By clicking directly on the indicator, rather than on a particular menu option, the teacher can still pull up the deep dive screen. However, the teacher also has the option to 1) privately acknowledge the hand raise and signal to the student that the teacher will be with them soon (by clicking the 'thumbs up' icon at the bottom, which is visible to the teacher only when a student is in 'hand raise' status); 2) ask for a recommendation for how to help the student (by clicking on the light bulb icon) – potentially a screen displaying a combination of what the AI believes the student needs, and what the student has self-reported needing help with (via a prompt in the tutor interface); or 3) send the student a quick message, selected from a set of very brief automated suggestions.

[P4] *Mixed feelings about monitoring student affect*

In line with findings from our prior research (e.g., Holstein et al., 2017b; 2019a; 2019b), teachers were interested in the prospect of automated support in monitoring their student's emotional states during a class session. Yet at the same time, teachers emphasized that they did not view this as a high priority feature. As Teacher 1 put it:

> *"If you're designing it, I don't know if that would be, like, one of your top priorities, I guess. I would think that that would be like, "Cool, if we have extra time, we'll just add that one in there," but there's probably fifteen other things that I would put on before that."*

When teachers were told that students participating in a prior speed dating study (Holstein et al., 2019b) wanted the option to hide real-time analytics about affect from their teacher, teachers were sensitive to the notion of affect analytics being particularly sensitive. For example, Teacher 2 said:

"I could agree with the emotion one but not to hide anything else [...] because some [students], probably, just don't want you to know, which is fine."

[P5] *Desires for selective shared awareness*

In line with findings from our prior research (e.g., Holstein et al., 2019a; 2019b; Holstein, Hong, et al., 2018), teachers expressed interest in allowing students to see a selected subset of their personal analytics (cf. Bull & Kay, 2016). However, teachers emphasized that they would want the ability to control this feature, turning it on or off on certain days, at certain times during a class session, or even for some students in a class but not others (Holstein et al., 2019a; 2019b; Holstein, Hong, et al., 2018). The following snippet of teachers' conversation reveals some of the factors they envision using to make such decisions:

Teacher 1: *"Yeah. If I feel like some of them, if they were able to see [their own real-time analytics], it would tick them off to no end."*
Teacher 3: ***"****I don't know if you could..."*
Teacher 1: "*Turn it on and off?"*
Teacher 3: *"Yeah, select it and...*
Teacher 2: *"Or for certain students..."*
Teacher 3: *"Yeah."*
Teacher 1: *"Because there are probably classes that are not as..."*
Teacher 3: *"High-strung."*
Teacher 4: *"Yeah, and crazy."*
Teacher 3: ***"****And students that aren't as high-strung, you know, just... yeah. For some, it might be motivation, for others, it could be nothing."*

[P6] *Desires for real-time teacher customization and control options*

In line with findings from our prior research (e.g., Holstein et al., 2017b; 2018b; 2019a; 2019b; Holstein, Hong, et al., 2018), teachers expressed desires for greater customizability and control – beyond awareness and decision support – in AI-enhanced classrooms. For example, as in Holstein et al. (2017b; 2019a) and Holstein, Hong, et al. (2018), teachers proposed the ability to remotely control students' current activities in the software, such as by "freezing" all students' screens, as Teacher 4 put it:

"What about, like, if you could just click the... like, it freezes everybody out for a moment? [...] Because sometimes when you're talking, they're so engaged [in the software] that they're not listening to a thing that you're saying."

In addition, as in Holstein et al. (2017b; 2019b), teachers shared experiences where the design of the software students used on ITS lab days did not align well with the textbooks students used on other days – creating unnecessary confusion. Teacher 1 noted that it would be helpful to be able to customize the terminology used in an ITS's hints and instructions:

> *"There were a couple of things we talked about this year that terminology was different [in MATHia]. When we'd go over [it], we'd say, 'Oh, well, this means proportion.' And then if the software could [also] hear that and then change it to our common terminology or even just, like, what our book uses, you know? [For example with] least value and greatest value, minimum and maximum [...] between even what the PSSAs use versus what the software uses versus what our book uses, there's like three sets of terms for the same thing, and if we haven't exposed them to it [...] that's what's confusing them."*

Similarly, Teacher 2 noted a disconnect between the input format students were taught in their textbooks, versus those they were asked to use in certain modules of the software:

> *"Our [students] were always hung up on the fractions, the way the fractions were set up, like finding a common denominator. They couldn't follow that format."*

Major differences observed between *MATHia* and *Lynnette* contexts

In addition the areas of alignment detailed above, we observed some key differences between *MATHia* and *Lynnette* contexts, pointing to different design requirements.

[D1] *Greater needs for support in handling classroom non-synchrony*
Lumilo was originally designed to support teachers in more effectively co-orchestrating non-synchronous classrooms. However, the current study revealed that further support is needed to support teachers in handling the greater degree of class asynchrony typical of *MATHia* classrooms (cf. Ritter et al., 2016a).

As discussed in section 10.1, *MATHia* contains a much broader range of content than *Lynnette*. In *MATHia*, tutored problems are divided into "workspaces." Each workspace may contain a diverse range of problems, united by a common broad topic. By analogy, the breadth of content covered by *Lynnette* is roughly equivalent to a single workspace in *MATHia* (focusing on equation solving). Furthermore, workspaces in *MATHia* are structured such that there is typically little, if any, overlap in skills across workspaces.

In the present study, to simulate students being spread across multiple workspaces – albeit to a lesser degree than may be typical of *MATHia* classrooms late in the school year (Ritter et al., 2016a) – students were given access to two workspaces: one that focuses on circle geometry and one that focuses on working with ratios. On the first day of the study, the majority of students worked on the circle geometry workspace. However, on the second day of the study, classes worked on a mixture of the circle geometry workspace and the ratio workspace. Some teachers asked their classes to begin with the ratio workspace on the second day rather than the circle geometry workspace, given that the latter appeared to be overly challenging for many students.

Even when a majority of students were working on this second workspace, *Lumilo–MATHia* persisted in showing teachers contextually-irrelevant information about the first workspace, given that students tended to struggle most with skills within this workspace. This had not presented as an issue in *Lynnette* classrooms given that different sections of *Lynnette* tended to build upon one another, with considerable overlap in the skills being tutored across sections. In addition, during periods of a class session in which students were heavily spread across the two workspaces, teachers often wanted to check on students' aggregate performance *within each workspace* that the class was currently working on. However, *Lumilo–MATHia*'s class-level display combined skills across all workspaces the class had been working on, meaning that skills from some workspaces (on which students tended to struggle more) drowned out skills from other workspaces.

To address these issues, teachers noted that it would be useful to have the ability to display summaries at the *workspace-level* (as illustrated in Figure 10-7) rather than at the *class-level* (which they had rarely found useful, as in *Lynnette* classrooms; see *Chapter 8*). As Teacher 3 said:

> *"It'd be nice if there was [a label] for [each workspace]. I knew I was having trouble when I would turn around to look at the back screen, like, all the information that it gave was on a module that only two students were on. So out of a class, it was like, 'Okay, who's working on [what]?' "*

Teachers also noted that it would be very important to be able to customize and/or automatically adapt which workspace-level summaries were visible or hidden from them at any given time, to avoid seeing a large amount of information that is irrelevant to their current tasks and goals. As Teacher 1 noted:

> *"...we have so many kids that will [all be] in a different area [...] You probably have 20 different [workspaces] that they are working on around [the same] time."*

[D2] *More pronounced desires for support in orchestrating group-level help sessions*
Perhaps due to lower levels of synchrony in *MATHia* versus *Lynnette* classrooms, we observed strong desires for support in orchestrating group-level interventions (cf. Holstein et al., 2019a). Teachers noticed during class that, even if a *majority* of the class was not struggling with similar issues, they were often providing the same help to multiple students throughout a class session.

As Teacher 3 noted:

> *"I had noticed... as I was going today, I had answered the same problem a hundred times and I could say it out loud [to the whole class], and then some still did answer, 'I don't know what to do here,' and it might help saying, '[You four,] hey, you guys are gonna need to know this now.' "*

Figure 10-7. Early illustration of *just one possible re-design* for the class-level summary displays in *Lumilo–MATHia* (a thorough exploration of how such features can best be designed is left for future work). Multiple summary screens are shown at the front of the classroom, divided by workspace in *MATHia*. Given the potential for students to be spread across a large number of workspaces in *MATHia*, details are made visible for only a subset of currently-active workspaces by default, based on (the system's knowledge of) a teacher's goals for the day. A teacher can further customize and control which summary screens are visible throughout the course of a class session. The name of the relevant workspace is shown at the top of each summary screen, together with the number of students who are currently active in that workspace. Only low-mastery, widely-practiced skills that are *relevant to a particular workspace* are shown in the corresponding summary screen. Within each screen, teachers can click on an encircled question mark to the right of each skill name (e.g., "Compute circle area (forward)") to receive more detailed information about the meaning of that skill, potentially accompanied by concrete examples of errors students have been making.

Figure 10-8. Early illustration of *just one possible design* for a commonly requested interaction with the class-level summary displays in *Lumilo* (a thorough exploration of how such features can best be designed is left for future work). If a teacher either clicks or sustains gaze on the grey portion of a class-level skill bar in a particular workspace's summary display (representing the proportion of currently active students who have practiced but not mastered a skill), the students who comprise this grey bar are revealed (both in the form of a list and by spatially highlighting students). In addition, other actionable information may be presented, such as suggestions for how to help these students. In the example shown, the tool suggests reviewing the circle area formula and/or reviewing what a radius is. Teachers can click on an encircled question mark to the right of each suggestion to receive more detailed information about 1) why they are seeing this suggestion, and 2) how to implement this suggestion. Note: any student names shown in this figure are fabricated.

Teachers noted that it would be helpful if the design of *Lumilo–MATHia* were to explicitly draw their attention to opportunities for group-level interventions – for example, by identifying specific students who are struggling within a given skill in a particular workspace, or by providing direct recommendations for ways they might help specific groups of students (as illustrated in Figure 10-8). For example, Teacher 1 suggested:

> *"Doing kind of, like, a really quick mini-lesson, I mean, I think that would be helpful. [...] You could just pull [up] those [recommendations], 'Hey, here's the five kids that aren't getting this right now. This is what you need to [do].' "*

In addition to providing group-level information and recommendations in real-time, teachers generated other ways a tool could help them orchestrate these group-level interactions. For example, Teachers 1 and 3 envisioned the ability to instantly share an example of an error that one student had made during class (in de-identified form) with other students who had recently been exhibiting similar errors (cf. Holstein et al., 2019a; Holstein, Hong, et al., 2018):

> ***Teacher 1:*** *" 'Look at this right now. This is you.' "*
>
> ***Teacher 3: "****Kinda pull their attention to it rather than [just] finding out [later]..."*

[D3] *More pronounced desires for real-time recommendations (for use at a teacher's discretion)* Finally, in response to a mockup of *Lumilo–Lynnette's* deep-dive screens, teachers noted that *MATHia* problems can involve much more context for the teacher to catch up on when approaching a student, compared with *Lynnette*. Given this difference, teachers suspected that having the option to access explicit action recommendations in real-time, in addition to other displays, could be particularly helpful in *MATHia* classrooms. Teacher 1's suggestion in [D2] is one example of a group-level recommendation the system might provide, to tell the teacher specifically what they "need to [do]," and with which group of students (as illustrated in Figure 10-8 and Figure 10-9).

Teachers expected that such action recommendations would be particularly helpful for those newer to using *MATHia* in the classroom (as they themselves had been one year before, when their school began using *MATHia* in classrooms). However, these teachers expected that even for teachers with more experience, receiving direct action recommendations could occasionally be more useful than only seeing raw examples of student work. For example, Teachers 3 and 4 discussed moments where using AI tutors in the classroom made them feel like "idiot[s]:"

> **Teacher 3: "***Well, and I know sometimes with MATHia, too, when I walked up on them, I need a minute to kinda see what they're doing. I guess maybe if I had it as, like, 'This is where they're at, this is kind of the direction that they're headed in,' sort of the preface while you're... Like, because there's been times on MATHia I'm like..."*
>
> **Teacher 4: "***And then you feel like an idiot because..."*

Teacher 3: ***"****...I don't know exactly what [the software] want[s]...*"

However, these teachers also stressed that they would likely find it bothersome to have proactive recommendations popping up regularly, suggesting instead that such a feature should instead be available upon a teacher's request.

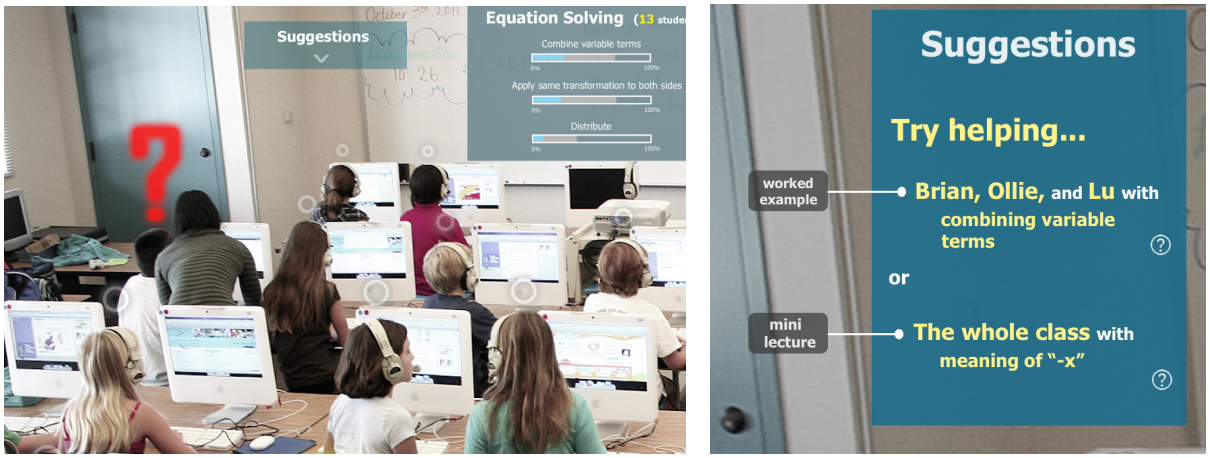

Figure 10-9. Early illustration of *just one possible design* for a new potential feature for *Lumilo–MATHia*: a general "Suggestions" screen that provides on-demand suggestions for teacher actions at any point during a class session (a thorough exploration of how such features can best be designed is left for future work). In this example, when the teacher glances at the "Suggestions" screen, it expands in response to the teacher's gaze to show two brief suggestions for good uses of the teacher's time. The first suggestion is to pull aside three specific students and to go over a worked example for "combining variable terms." The second suggestion is to give a quick whole-class lecture on the meaning of a negative 'x.' Teachers can click on an encircled question mark to the right of each suggestion to receive more detailed information about 1) why they are seeing this suggestion, and 2) how to implement this suggestion. Note: student names shown in this figure are fabricated.

10.4 Conclusions and Next Steps

In sum, piloting in *MATHia* classrooms revealed many alignments with prior findings, but also pointed to important challenges ahead in adapting the design of *Lumilo* to work with *MATHia*. Table 10-3 presents an updated version of Table 10-1, illustrating key differences observed between *Lynnette* and *MATHia* contexts. More broadly, the rows of Table 10-3 represent some of the challenges involved in scaling up real-time, wearable teacher augmentation to a broader range of curricular content and classroom contexts, as well as longer periods of use.

Table 10-3. Anticipated challenges in adapting the design of *Lumilo* to work with *MATHia*; an updated version of Table 10-1, following observations from the Spring 2019 classroom pilots and design workshop.

Lynnette	MATHia	Relevant observations during Spring 2019 classroom study?
Covers on the order of **days - weeks** of curricular content, and is used over relatively short timespans.	Covers on the order of **months - years** of curricular content, and is often used continuously throughout the school year.	**No.** This study was too brief to observe unique challenges that arise over months to years of use. However, the following two rows relate to this issue.
Includes a **smaller range of problem types** (all tutor problems follow a similar format: line-by-line equation solving).	Includes a **broader range of problem types** (tutor problems span a wide range of formats and topics).	**Yes.** Given *MATHia*'s broader range, some problem types and formats presented by *MATHia* were unfamiliar to teachers. In these areas, teachers lacked relevant pedagogical content knowledge needed to quickly help their students. Relatedly, the presence of a broader range of problem types seemed to contribute to the amount of context a teacher needed to catch up on, when approaching a given student (see row 4).
Considerable overlap in skill content throughout a student's trajectory in the software (i.e., when moving between distinct "problem sets").	Students may move between contexts ("workspaces") with **little if any overlap in skill content**.	**Yes.** This impacted the usefulness of the class-level display in *Lumilo–MATHia.* Findings pointed to needs for greater context-awareness, organization of information by workspace, and greater support in orchestrating group-level interventions during class.
Problems include **less context** that the teacher needs to catch up on, in the moment (students are presented with an algebraic equation and asked to solve it).	Some problem types involve **substantial context** that a teacher needs to catch up on, in the moment (e.g., a given problem may present students with detailed, multi-step word problems and interactive graphical representations).	**Yes.** in the post-workshop, teachers anticipated that providing raw error examples may be less useful than providing direct action recommendations in these cases.
Lumilo's real-time analytics and parameter settings have been **tuned** for *Lynnette* classrooms (see *Chapters 4 through 6*).	*Lumilo*'s real-time analytics and parameter settings have **not been tuned** for *MATHia* classrooms.	**Yes.** For example, *MATHia*'s design (e.g., its problem selection policy, and the distribution of skills across steps, problems, and workspaces) appears to have rendered *Lumilo*'s existing detector of unproductive persistence less useful.

The work presented in this chapter represents a rare exploration of challenges that arise in adapting the interface and algorithm design of a learning analytics tool to work with an educational software system for which it was not originally designed (see item 2 under *Summary of Contributions – First design exploration and prototypes of wearable, heads-up displays to support orchestration of personalized classrooms*). Despite growing awareness in the learning analytics, AI in education, and educational data mining communities of the difficulty of transferring student modeling methods and learning analytics tools across different educational software systems, it remains rare to see demonstrations of generalization across systems, or explorations of challenges that arise when trying to generalize across systems (see *Chapter 3*; Holstein, Yu, et al., 2018; Paquette et al., 2015; 2018).

Below, I provide a high-level summary of next steps for this research – based on needs, challenges, and tensions identified in both *Chapters 9* and *10* – to prepare real-time, wearable teacher augmentation for larger-scale, longer-term use (a key goal of our ongoing collaboration with Carnegie Learning).

This summary is not intended to be exhaustive. Indeed, as discussed in the *Conclusions, Contributions, and Future Directions* section, I believe there are multiple lifetimes worth of research to be done in this space. Rather, this summary represents just a small sampling of next steps that I believe to be particularly high priority in the (relatively) near-term, especially within the context of our ongoing academic–industry partnership. Broader conclusions and directions for future work are presented in *Conclusions, Contributions, and Future Directions.*

High-level summary of next steps

Designing to support more impactful teacher interventions

- Further explore (1) how real-time action recommendations or "Suggestions" can best be presented to teachers (see *Chapters 1, 5, 9*, and *10*); and (2) how to generate such recommendations (e.g., through the design of teacher-sourcing mechanisms (cf. Heffernan et al., 2016; Wang et al., 2019).
- Explore how best to support teachers in opportunistically and adaptively orchestrating group-level interventions during a class session (e.g., helping teachers identify small groups of students who are experiencing similar difficulties for a targeted mini lesson) (see *Chapters 1*, *5*, *8*, *9*, and *10*).
- Explore how to better support teachers in monitoring and regulating student motivation during AI-supported class sessions (e.g., by allowing teachers to dynamically switch classes between various individual, competitive, and collaborative "modes"; see *Chapters 1, 4, 9*, and *10*)).

- Explore how teachers might be better supported in inferring potential causal impacts of their own actions during AI-supported class sessions (e.g., learning from feedback on the effectiveness of their own on-the-spot explanations; see *Chapters 1*, *4*, *5*, *9*, and *10*).

Designing for effective balances of teacher, student, and AI control and regulation

- Further explore whether and how best to design private teacher–student communication channels (see *Chapters 1*, *4, 5, 9,* and *10*), and investigate how the presence of particular forms of private teacher–student signaling (e.g., "invisible hand raises", private teacher acknowledgements, the ability for students to request not to be helped) impacts classroom dynamics (e.g., student help-seeking behaviors) (cf. Schofield et al., 1994).
- Further explore 1) which information should be shown to which stakeholders in the classroom, and under which circumstances; and 2) the design of mechanisms to support teacher and student control over real-time information sharing in the classroom (see *Chapters 4*, *9*, and *10*).

Moving beyond a research prototype

- Move to lighter-weight hardware, as this becomes increasingly technically and economically feasible (Bohn, 2019; Harrison, 2018; Robertson, 2019; see *Chapters 4* and *10*).

Designing orchestration support for less-synchronized classroom contexts

- Explore how best to present class-level or group-level information in situations where students are spread across a large number of divergent activities (e.g., 20 or more workspaces in *MATHia*), without overwhelming teachers or counteracting the usefulness of these displays (*Chapter 10*).

Adapting to diverse classroom contexts and pedagogical goals

- Further explore how best to design detector algorithms that work well in *MATHia contexts*. In addition to addressing the challenges described in this chapter, this step may involve running analyses to better understand how these detector algorithms behave across the broad range of classroom, cultural, and socio-economic contexts in which *MATHia* is used, and designing mechanisms for contextual adaptivity and adaptability where helpful (see *Chapters 5* and *10*; Baker, 2019; Holstein & Doroudi, 2019; Holstein, Wortman Vaughan et al., 2019; Karumbaiah et al., 2019; Ocumpaugh et al., 2014; Ogan et al., 2015).
- Explore how best to make *Lumilo–MATHia*'s design more adaptable and/or adaptive (see *Chapters 5*, *8*, and *10*) – for example, by enabling teachers to see class-level information relevant to their specific instructional goals *for a given day*, and ensuring that teachers only see information relevant to workspaces students are actively working on.

- Explore ways to provide teachers with greater ability to customize and control the behavior of AI tutoring software for use in their classrooms (cf. Holstein et al., 2019a; 2019b), while also ensuring that the instructional effectiveness of the software is maintained (see *Chapters 1*, *9*, and *10*; Ritter et al., 2016a).

Conclusions, Contributions, and Future Directions

Conclusions

In this dissertation, I have begun to explore how AI and human teachers might best support one another, leveraging one another's complementary strengths to achieve outcomes greater than either could achieve alone.

I have approached this work from both an *empowerment* and an *efficiency* perspective (see Kulkarni et al., 2019). From an *empowerment* perspective, I have begun to explore how educational AI systems might be better designed to support and extend teachers' abilities to personalize instruction, and help them fulfill the roles they aspire to play during AI-supported class sessions (e.g., see *Chapters 1, 4, 5, 8,* and *9*; Aiken & Epstein, 2000; Feng & Heffernan, 2007; Holstein et al., 2017b; 2019a; 2019b; Holstein, Hong et al., 2018; Mavrikis et al., 2016; Yacef, 2002). From an *efficiency* perspective, I have begun to explore how human/AI systems can be designed to measurably benefit students' learning, by making more effective use of existing classroom resources (compared with human teachers or AI tutors working in a less-integrated fashion) (e.g., see *Chapter 2*, *5, 6, 7*, and *9;* Baker et al., 2016; Fancsali et al., 2018; Holstein et al., 2017a; 2018a; 2018b; Holstein, 2018; Kamar, 2016; Ritter et al. 2016b).

In *Part One,* I presented findings from initial needfinding studies with K-12 teachers who have used AI systems as part of their regular instruction (*Chapter 1*). In addition, I presented exploratory data analyses of teacher–student interactions in AI-supported classrooms and their relationships with students' learning (*Chapter 2*), and introduced *CTAT/TutorShop Analytics (CT+A)* an extended technical architecture for intelligent tutoring system (ITS) development and deployment that can support the prototyping of real-time analytics tools for use with ITSs (*Chapter 3*).

In *Part Two*, I presented an iterative prototyping process with K-12 teachers, yielding new prototyping methods and the development of a new form of real-time teacher augmentation: a prototype of mixed reality smart glasses for teachers called *Lumilo* (*Chapters 4* and *5*).

Part Three focused on the evaluation of real-time teacher augmentation in live classroom settings. I presented and demonstrated a design framework for the iterative, data-informed design and evaluation of real-time teacher augmentation (*Chapter 6*), culminating in an in-vivo classroom experiment that evaluated *Lumilo*'s impacts on teacher and student behavior and students' learning (*Chapters 7* and *8*).

Finally, in *Part Four* I began to explore how the concepts embodied by *Lumilo* might be prepared for wider use, through design studies (*Chapters 9* and *10*) and classroom piloting with teachers and students (*Chapter 10*). Beyond the scope of this dissertation, the explorations presented in *Part Four* will help prepare for the next phase of this research: a large-scale classroom experiment (using an updated and miniaturized version of *Lumilo*) with over 60

middle school classrooms that use commercial AI tutoring software (Carnegie Learning's *MATHia* ITS), to better understand the effects of teacher–AI co-orchestration on student learning and other classroom outcomes.

To conclude, I will first present high-level *methodological reflections and recommendations* based on my experiences designing real-time teacher augmentation with and for K-12 teachers. It is my hope that these reflections will be helpful to others who wish to involve non-technical stakeholders in the design of learning analytics (LA), educational AI (AIED), and related systems (see Holstein et al., 2019a for a discussion). Finally, in the next sections, I will present a detailed overview of this dissertation's seven main contributions, followed by a brief discussion of some broad directions for future work.

Methodological reflections and recommendations

Although recent work in the field of Learning Analytics (LA) encourages stakeholder involvement at every stage of design and development — from early, generative design phases through piloting and evaluation in real-world educational contexts — demonstrations of end-to-end co-design processes for LA and educational AI (AIED) systems remain very rare in the literature (Holstein et al., 2019a). Furthermore, existing user-centred design workflows and frameworks (e.g., Dollinger & Lodge, 2018; Martinez-Maldonado et al., 2016) provide limited methodological guidance regarding *how* to effectively involve non-technical stakeholders at each phase of an LA/AIED design process (Holstein et al., 2019a).

Below, I present some general reflections and recommendations for future LA/AIED co-design efforts, reflecting on "lessons learned" and practices I have found valuable in my dissertation work. In particular, I focus on specific practices that go beyond those explicitly highlighted in prior work on the user-centered design of LA/AIED systems. I expect that several of these recommendations will generalize to the design of other data-driven algorithmic systems, beyond the context of educational technologies. For example, the approach taken in Holstein, Wortman Vaughan, et al. (2019) followed several of these recommendations, towards designing more effective tools to help machine learning practitioners assess/address algorithmic bias and unfairness in widely-used systems.

1. **Begin with stakeholder needs, not analytics or visualizations.**
   In designing any tool, it is useful to begin with an understanding of the stakeholder needs a tool might address, and the tasks and experiences it might support. Yet design processes for LA tools often appear to begin by identifying technical solutions (e.g., particular data sources, analytic methods, and visualizations), and then searching for opportunities to apply these solutions (Rodriguez-Triana et al., 2017). In the early phases of my design research (see *Chapter 1*), I explicitly avoided discussing particular solutions with teachers, to avoid limiting these conversations by teachers'

conceptions of what is technologically possible. Instead, I found it much more generative to explore teachers' current *challenges* and *aspirations* through probes like the "superpowers" exercise and through directed storytelling around teachers' lived experiences (Beyer & Holtzblatt, 1997; Evenson, 2006). Findings from such design exercises subsequently enabled "matchmaking" between specific teacher information needs (e.g., desires to receive real-time updates about specific student constructs) and current technical possibilities (e.g., existing analytic and student modelling methods intended to measure these constructs).

In some cases, beginning from stakeholder needs led me away from the use of more abstract data visualizations that are common in existing "learning analytics dashboards" (and which might otherwise have been a default, assumed solution) — such as plots, graphs, and charts — towards the use of concrete, grounded representations of student data such as raw examples of student errors (cf. Bull & Kay, 2016)

2. **Regularly link analytics to action throughout the design process.**
   Although many learning analytics tools are designed to support awareness or reflection (An et al., 2019; Rodriguez-Triana et al., 2017), the end goal of this enhanced awareness or support for reflection is commonly to support more informed decision-making and action (Holstein et al., 2018a; Schoenfeld, 2010). Throughout my design process, I found that it was critical to regularly link particular analytics to the *decisions and actions* they might inform. For example, in early prototyping studies, I found that prompting teachers to reflect on what real-time decisions a particular information display might inform often led them to notice ways in which the display could be made more useful, usable, and/or trustworthy (Holstein, 2018; Holstein et al., 2019; Holstein, Hong, et al., 2018). In many cases, teachers would initially find particular visualizations interesting and appealing, yet would change their minds about the desirability of these visualizations when prompted to reflect upon *how they might actually use* these visualizations to inform their classroom practice.

3. **Prototype specific *user tasks* and *usage scenarios* early and often.**
   In line with the previous recommendation, I have found it very useful to simulate specific user tasks and usage scenarios for an LA/AIED tool (e.g., by having stakeholders participate in role-playing and bodystorming exercises) as early and often as possible throughout the design and prototyping process (cf. Odom et al., 2012; Zimmerman & Forlizzi, 2017). Such simulation exercises (combined with methods like think-alouds and cognitive task analyses) can help to surface information needs crucial for particular tasks or usage scenarios, but which users may not otherwise perceive or report "out of context" (cf. Beyer & Holtzblatt, 1997; Crandall et al., 2006). For example, since different usage scenarios for a teacher analytics tool can involve very different types of tasks and

decisions (e.g., planning a lesson versus identifying students who need help right now), different constraints (e.g., more or less time pressure), and different affordances for action, simulating specific usage scenarios can sometimes reveal that radically different designs are needed to support different scenarios.

4. **Prototype the behavior of LA/AIED tools using diverse real-world datasets.**
   Finally, since the behavior of LA/AIED systems can depend heavily on nuances of particular data-generating contexts, in combination with particular analytic methods or algorithms, I have found that it can be very informative to run prototyping sessions using datasets from a range of contexts. For example, by replaying data collected from real classrooms across multiple school contexts and performance levels in Replay Enactments sessions, I was able to anticipate various context-specific design challenges before entering live classrooms (see *Chapter 5*; Dove et al., 2017; Holstein et al., 2019a; Holstein, Wortman Vaughan, et al., 2019; and a brief discussion of challenges in "Global Design" in Zimmerman & Forlizzi, 2019).

Summary of Contributions

This thesis makes a total of 7 main contributions to the areas of human–computer and human–AI interaction (**HCI/HAI**), design (**DES**), and learning sciences and technologies (**LS&T**). The broad category "LS&T" encompasses such subareas as AI in education, learning analytics, the learning sciences.

Contributions are organized below by groupings of relevant fields. For each listed contribution, relevant thesis chapters and my related prior publications are provided, together with a brief summary of the significance of the contribution with respect to prior literature.

Following Wobbrock and Kientz's high-level taxonomy of research contribution types in HCI (Wobbrock & Kientz, 2016), contributions are categorized by each contribution's primary type (out of "Empirical", "Artifact", "Methodological", "Theoretical", "Dataset", "Survey", and "Opinion"). Secondary contribution types are also listed where applicable. I have further divided the "Empirical" category into two subcategories (although note that these are not mutually exclusive): "Design research" and "Classroom experiments and data mining".

Contributions to the areas of Human–Computer / Human–AI Interaction (HCI/HAI), Design (DES), and Learning Sciences & Technologies (LS&T):

1. **First broad design exploration of needs for real-time teacher analytics and orchestration support:**
   This dissertation presents the first broad exploration in the literature of teachers' needs for real-time analytics and orchestration support in AI-supported, personalized classrooms. As AI increasingly enters K-12 classrooms, it is important to understand the evolving roles and aspirations of K-12 teachers, and how AI systems can best be designed to support these roles. Through these explorations, the present work is also among the first to explore the notion of human–AI co-orchestration, in which a human teacher and students work together with AI agents to make complex, yet powerful classroom learning scenarios feasible.
   More broadly, the design explorations presented in this dissertation represent a case study of the design of real-time AI augmentation for workers in a "caring profession" (K-12 teaching) which may defy full automation.
   - **Main contribution type(s):** Design Research & Theoretical
   - **Most relevant chapters:** *Chapters 1, 4,* and *9*
   - **Related publications:** Holstein et al., 2017b; 2019a; 2019b; Holstein, Hong, et al., 2018.
   - **Significance:**

- In recent years, many real-time analytics tools have been designed and developed to aid teachers in orchestrating complex technology-enhanced learning scenarios (e.g., van Alphen & Bakker, 2016; Martinez-Maldonado, Clayphan, Yacef, & Kay, 2016; Matuk, Gerard, Lim-Breitbart, & Linn, 2016; Mavrikis, Gutierrez-Santos, & Poulovassilis, 2016). However, design decisions about which analytics to present to teachers often appear to be driven more by the availability of data or pre-existing analytics measures than by considerations of teachers' real-time information needs (Rodriguez-Triana et al., 2017). To the best of my knowledge, **no prior literature has conducted broad needfinding studies – untethered from specific, pre-existing prototypes – to understand teachers' needs and desires for real-time analytics** (Holstein et al., 2017b; 2019a; 2019b; Holstein, Hong, et al., 2018).
- **Many existing real-time orchestration tools have been designed with the assumption that a class of students progresses through instructional activities in a relatively synchronized manner** (cf. van Leeuwen, 2015; but see Olsen, 2017). Understanding how best to support teachers in orchestrating highly- differentiated, non-synchronous classrooms, such as those using AI tutoring systems, remains an important and challenging research problem. Orchestration support for such classrooms must alleviate the implementation challenges that these classrooms raise for the teacher (e.g., Alphen & Bakker, 2016; Bingham, Pane, Steiner, & Hamilton, 2018; Holstein et al., 2017b; Holstein, Hong, et al., 2018).
- **Most existing real-time orchestration tools have been developed to support instructors in university contexts** (e.g., Alavi, Dillenbourg, & Kaplan, 2009; Martinez-Maldonado et al., 2015; Rodriguez-Triana et al., 2017). Teachers working with younger students (e.g., in middle school classrooms) may face unique challenges. For example, when working with younger students, it may be important for teachers and peers to play a more proactive role in helping to regulate students' learning, help-seeking, and motivation (Aleven, Roll, McLaren, & Koedinger, 2016; Holstein et al., 2017a; 2019a; 2019b; Molenaar, Horvers, & Baker, 2019; Nelson-Le Gall, 1981; Zimmerman, 2008).
- **Most prior work on real-time orchestration tools has focused on teacher support, taking a teacher-centered view of classroom orchestration** (e.g., Alphen & Bakker, 2016; Martinez-Maldonado et al., 2015; Rodriguez-Triana et al., 2017). While some prior work has introduced the notion of "co-orchestration" (e.g., Muñoz-Cristóbal et al., 2013; Prieto, 2012; Sharples, 2013), this work has tended to focus on the sharing of

responsibility for orchestration across multiple human agents in the classroom (such as when a teacher "offloads" help-giving by initiating a peer-tutoring interaction). This dissertation is **among the first to explore the notion of *human–AI* co-orchestration** (Holstein et al., 2017b; 2018b; 2019a; 2019b; Holstein, 2018), in which a human teacher and students work together with AI agents to make complex, yet powerful classroom learning scenarios feasible (cf. Forlizzi & Zimmerman, 2013; McLaren et al., 2010; Olsen, 2017; Olsen et al., 2018; Prieto, 2012; Wetzel et al., 2018).

- **As AI increasingly enters K-12 classrooms, it is important to understand the evolving roles and aspirations of K-12 teachers, and how AI systems can best be designed to support these roles** (Holstein et al., 2017b; 2019a; 2019b; Huber et al., 2014; van Leeuwen et al., 2018; Olsen, 2017; Olsen et al., 2018; Toyama, 2017). When designing human–AI co-orchestration systems, it is critical to understand which teacher and student needs these systems might serve, and what social boundaries these systems should avoid crossing (cf. Davidoff et al., 2007; Forlizzi & Zimmerman, 2013; Zimmerman & Forlizzi, 2017). *Chapter 9* presents an initial exploration of teacher and student beliefs about desirable human/AI role divisions in AI-supported classrooms: which aspects should be handled by AI agents versus teachers or peers, and under which circumstances.
- **More broadly**, the design explorations presented in this dissertation represent a **case study of the design of real-time AI augmentation for workers in a "caring profession"** (K-12 teaching) which may defy full automation. This is a critical yet relatively underexplored area in the HCI and HAI literatures, and relates to the topics of several recent workshops at HCI conferences such as CHI, DIS, and CSCW: "Designing for Everyday Care in Communities" (Toombs, Dow, et al., 2018), "Sociotechnical Systems of Care" (Toombs, Devendorf, et al., 2018), and "Where is the Human? Bridging the Gap Between AI and HCI" (Inkpen, De Choudhury, Chancellor, Veale, & Baumer, 2019). The present work presents a relevant case study in the context of co-located K-12 teaching, a profession where full automation may remain infeasible and/or undesirable (Baker, 2016; Duckworth, Graham, & Osborne, 2019; Frey & Osborne, 2013; Holstein et al., 2017b; 2019a; 2019b; Lubars & Tan, 2019).

2. **First design exploration and prototypes of wearable, heads-up displays to support orchestration of personalized classrooms:**
   The first design exploration in the literature of the use of wearable, heads-up displays to support teachers in orchestrating personalized classrooms, yielding *Lumilo*, a

classroom-tested prototype of mixed reality smart glasses for teachers.
More broadly, the design explorations presented in this dissertation represent a case study of the use of head-mounted displays in a real-world social space (K-12 classrooms).

- **Main contribution type(s):** Design Research & Artifact
- **Most relevant chapters:** *Chapters 4, 5, 7, 8,* and *10*
- **Related publications:** Holstein et al., 2017b; 2018b; 2019a; Holstein, Hong, et al., 2018.
- **Significance:**
  - **Recent work has begun to investigate the potential of emerging wearable technologies for real-time teacher support** (e.g. Quintana, Quintana, Madeira, & Slotta, 2016; Zarraonandia, Aedo, Díaz, & Montero, 2013). Such technologies hold great promise to enhance teacher awareness, while allowing teachers to keep their heads up and eyes focused on their classroom – acknowledging the highly active role teachers play in personalized classrooms (see *Chapter 1*; Holstein et al., 2017a; 2017b; Quintana, et al., 2016, Schofield et al., 1994).
  - **Prior work suggests that teachers may prefer wearables over handheld devices for use in personalized classrooms** (e.g., Quintana, et al., 2016). **However, this work has not involved the human-centered design and evaluation of an actual wearable orchestration tool.** Furthermore, while prior work has tested the use of smart glasses to help students provide live feedback to their instructors in university lecture contexts (Zarraonandia, et al., 2013), the present work represents the first exploration of the affordances of smart glasses to support teachers in orchestrating personalized classroom sessions, yielding *Lumilo*, a classroom-tested prototype of mixed reality smart glasses for teachers (see *Chapter 4*).
  - **More broadly**, the design explorations presented in this dissertation represent **a case study of the use of head-mounted displays in a real-world social space** (K-12 classrooms). The design of wearable, heads-up displays for use in actual social spaces remains relatively underexplored in the HCI literature. This relates to the topic of recent workshops at HCI conferences, such as "Challenges Using Head-Mounted Displays in Shared and Social Spaces" (Gugenheimer, Mai, McGill, Williamson, Steinicke, Perlin, 2019). The present work presents a relevant case study in the context of co-located K-12 teaching, where the wearer of a head-mounted display is nearly constantly involved in brief social engagements (with students).
  - The research presented in *Chapter 10* expands upon this work, presenting a **rare exploration of challenges that arise in adapting and generalizing the interface and algorithm design** of a learning analytics tool (*Lumilo*) to work

with an educational software system for which it was not originally designed. Despite growing awareness in the learning analytics, AI in education, and educational data mining communities of the difficulty of transferring student modeling methods and learning analytics tools across different educational software systems, it remains rare to see demonstrations of generalization across systems, or explorations of challenges that arise when trying to generalize across systems (see *Chapter 3*; Baker, 2019; Holstein, Yu, et al., 2018; Paquette et al., 2015; 2018).

3. **First experimental study to demonstrate student learning benefits of real-time teacher analytics:**
   The first experimental study to demonstrate student learning benefits (on a pre- and post-test) of real-time teacher analytics or teacher–AI co-orchestration.
   - **Main contribution type(s):** Classroom experiments and data mining
   - **Most relevant chapter:** *Chapter 7*
   - **Related publications:** Holstein et al., 2018b.
   - **Significance:**
     - Although much prior work has focused on the design, development, and evaluation of teacher analytics tools, very few studies have evaluated effects on student learning (Kelly et al., 2013; Molenaar & Knoop-van Campen, 2017; Rodríguez-Triana, et al., 2017; Xhakaj, et al., 2017). This dissertation presents **the first experimental study to demonstrate that real-time teacher analytics can enhance students' learning outcomes (within or outside the areas of AI-supported education and intelligent tutoring systems).**
     - These experimental findings demonstrate potential for AIED systems that integrate human and machine intelligence to support students' learning (cf. Baker, 2016; Kamar, 2016; Ritter et al., 2016; Yacef, 2002).

4. **Novel design and prototyping methods:**
   Novel design and prototyping methods to support the co-design, experience prototyping, and evaluation of data-driven AI systems, and case studies exploring how these methods can be applied to involve non-technical stakeholders in the design of such systems. These methods include Replay Enactments, Participatory Speed Dating, and the use of spatial classroom replay visualizations to inform design.
   - **Main contribution type(s):** Methodological
   - **Most relevant chapters:** *Chapters 2*, *5*, *6*, and *9*
   - **Related publications:** Holstein et al., 2017a; 2017b; 2018a; 2019a; 2019b; Holstein, Hong, et al., 2018

- **Significance:**
  - Recent work in HCI/HAI has highlighted **needs for new kinds of prototyping methods to address unique challenges that arise in prototyping data-driven AI systems** (e.g., Dennerlein et al., 2018; Dove et al., 2017; Doshi-Velez & Kim, 2017; Helms, et al., 2018; Holstein, Wortman Vaughan, et al., 2018; Yang, Sciuto, et al., 2018).
  - This dissertation **introduces and demonstrates replay-based prototyping methods that use authentic data and (imperfect) algorithms** to reveal important nuances that other methods – such as Wizard of Oz studies (Lovejoy, 2018; Odom et al., 2012) – may be ill-suited to surface, (e.g., UX impacts of a prototype's false positives and negatives (Dove et al., 2017) or issues that arise only in particular data-generating contexts).
  - In addition, recent work in HCI/HAI and Learning Analytics has begun exploring **how non-technical stakeholders can be meaningfully involved in shaping the behavior complex, data-driven AI systems** – a central open challenge for the UX design of data-driven AI systems (e.g., Baumer, 2017; Chen & Zhu, 2019; Dennerlein et al., 2018; Kyung Lee et al., 2018; Holstein, Hong, et al., 2018; Prieto-Alvarez, et al., 2018; Zhu & Terveen, 2018). This dissertation explores this question in the context of AI-supported K-12 classrooms, introducing methods and strategies for effectively involving K-12 teachers and students in the design of data-driven AI systems (Holstein, Hong, et al., 2018; Holstein et al., 2019a; 2019b).

Contributions to the area of Learning Sciences and Technologies (LS&T):

5. **First investigations of relationships between teacher movement/monitoring and student behavior and learning in AI-supported classrooms:**
   This dissertation presents the first investigations in the literature of relationships between teachers' physical movement and classroom monitoring behaviors, and students' behaviors and learning outcomes, in AI-supported, personalized classrooms. Findings from this research indicate that, when evaluating the impacts of teacher-facing learning analytics tools, future research should take care to tease apart potential effects of a teacher's use of a monitoring tool versus teachers' use of learning analytics.
   - **Main contribution type(s):** Classroom experiments and data mining
   - **Most relevant chapters:** *Chapters 2, 4*, *6,* and *7*
   - **Related publications:** Holstein et al., 2017a; 2018a; 2018b; 2019a; Holstein, Hong et al., 2018.
   - **Significance:**

- While prior work has investigated associations between teachers' physical movement and monitoring behaviors in co-located classrooms (e.g., looking over students' shoulders as they work) and students' behaviors and learning outcomes, using observational data (e.g., Stang & Roll, 2014), the present work is the first to investigate such associations in the context of AI-supported, personalized classrooms – **a context where student behavior has been extensively studied using software log data alone, without data on out-of-software interactions such as those that occur between students and teachers** (see also: Miller, et al., 2015).
- Findings from an in-vivo classroom study (Holstein et al., 2018b) suggest that, **when evaluating the impacts of teacher-facing learning analytics tools, future research should take care to tease apart potential effects of a teacher's use of a monitoring tool** (such as novelty effects or students' awareness of being monitored by their teacher), **versus teachers' use of the kinds of advanced analytics and student modeling methods that are often the focus of research** in learning analytics (LA), AI in education (AIED), user modeling (UM), and educational data mining (EDM).
- I developed and utilized a new logging library for use with *Lumilo*, which appropriates the HoloLens 1's spatial mapping capabilities as a means of **automatically logging teachers' actions in a physical classroom space over the course of a class session** (i.e., to automate much of the manual coding process described in *Chapter 2*). For example, using a "mixed reality sensor" approach (see *Chapter 4*), *Lumilo* can record time-stamped logs of a teacher's physical proximity to a given student in the class moment-by-moment, as well as the teacher's absolute location in the classroom, their proximity to pre-specified landmarks (such as the teacher's desk or whiteboard), the target of a teacher's gaze, and all teacher interactions within the tool interface.
- **Unlike most prior work on physical teaching analytics** (e.g., An et al., 2019; Echeverria et al., 2018; Martinez-Maldonado, 2019; Martinez-Maldonado et al., 2018; but see Prieto et al., 2016), ***Lumilo's* mixed reality sensor approach uses an "inside out" approach to teacher tracking, and thus does not require instrumenting the classroom space with external sensors or "beacons".** Rather, this approach relies entirely on the HoloLens 1's built in sensors and spatial mapping algorithms for tracking of teachers' behavior.

6. **Causal Alignment Analysis (CAA), a framework for the data-informed, iterative design of teacher augmentation:**
   This dissertation presents Causal Alignment Analysis (CAA), a framework for the data-informed, iterative design of teacher augmentation (e.g., real-time awareness and orchestration tools), which links the design of such technologies to educational goals; and a case study illustrating CAA's application and utility.
   - **Main contribution type(s):** Methodological & Theoretical
   - **Most relevant chapters:** *Chapter 6*
   - **Related publications:** Holstein et al., 2018a.
   - **Significance:**
     - While existing design workflows such as LATUX (Martinez-Maldonado, Pardo, et al., 2016) support the user-centered design of teacher analytics tools based on teacher feedback, **there is a lack of standard methodology for the outcome-driven improvement of such tools, to achieve targeted educational goals** (cf. Molenaar & Campen, 2017; Xhakaj et al., 2017). Furthermore, **justifications for design decision**s (e.g., what information to present in a teacher dashboard) **are rarely reported in the literature** (Rodríguez-Triana et al., 2017). CAA is a framework to guide the outcome-driven design of teacher analytics tools, and to help structure reporting of justifications for design decisions.

7. **CTAT/TutorShop Analytics (CT+A), an extended technical architecture for ITS development that supports "extensible student modeling":**
   CTAT/TutorShop Analytics (CT+A), an extended technical architecture for ITS development that supports "extensible student modeling": the sharing, re-mixing, use, and prototyping of advanced student modeling techniques.
   - **Main contribution type(s):** Artifact
   - **Most relevant chapters:** *Chapter 3*
   - **Related publications:** Holstein, Yu, et al., 2018.
   - **Significance:**
     - **Authoring tools for intelligent tutoring systems (ITSs) rarely support extensible student modeling.** For example, prior to the present work, CTAT/Tutorshop, an authoring environment for cognitive tutors and example-tracing tutors that has been used to build many dozens of ITSs (Aleven et al., 2016), supported only student models comprising a set of BKT mastery probabilities for knowledge components (KCs) within the authored tutors. An author could not add other types of variables to the student model (e.g., to track the student's affective or motivational state, or

metacognition) or easily experiment with different methods for updating or using the student model.

- The present work **aims to lower the barriers to the sharing, re-use, and re-mixing of advanced student modeling methods across researchers and research groups, with the goal of accelerating progress within a *cumulative science* of student modeling** (cf. Desmarais & Baker, 2012; Sotillare et al., 2018). *CT+A* is already being used to share student modeling techniques across research groups, for use in live ITSs or for offline analyses (e.g., Holstein, Yu, et al., 2018; Paquette et al., 2018).

Future Directions

Perhaps the most important "meta-contributions" of this dissertation are: **(1)** This work has helped to lay a foundation for a broad research program around the design of "hybrid" systems that combine complementary strengths of human and AI instruction in-the-moment (Baker, 2016; Fancsali et al., 2018; Holstein et al., 2017b; 2018a; 2018b; 2019a; 2019b; Holstein, 2018; Molenaar et al., 2019; Ritter et al., 2016b; Wetzel et al., 2018; Yacef, 2002); and **(2)** This work has helped to advance conversations around the nature and roles of design research and co-design in the areas of AI-supported Education (AIED) and Learning Analytics (LA) (e.g., Buckingham Shum, Ferguson, Martinez-Maldonado, 2019; Holstein et al., 2017b; 2019a; 2019b; Prieto-Alvarez et al., 2018). My hope for these fields is that we will continue to expand beyond designing "AI systems" or "learning analytics systems", towards designing effective human/AI *partnerships* and *service systems* (Morelli, 2003; Vargo, Maglio, & Akaka, 2008). This means understanding and designing for the broader contexts in which these AIED/LA systems are embedded – viewing the value of these systems as *co-created* in action (Payne et al., 2008; Prahalad and Ramaswamy, 2004; Vargo et al., 2008) amongst various human and AI stakeholders[24].

I believe there are multiple lifetimes worth of important research yet to be done within these areas, and it is my hope that findings presented throughout this dissertation will help to inform future work. The directions explored in this dissertation have already proven to be highly generative. For example, the design research presented in this dissertation has inspired at least six major, awarded research grants so far (some of which I have worked on, and some of which were written by colleagues) which will explore the design and evaluation of new forms of real-time teacher augmentation, for use in a broader range of classroom contexts than explored in the present work.

As mentioned in *Part Four* of this dissertation, one of my own next steps will be to gain a deeper understanding of the impacts of teacher–AI co-orchestration via a larger-scale, longer-term classroom experiment (using an updated and miniaturized version of *Lumilo*). Specific directions for future research that tie-in to this project are presented at the end of *Chapter 10*, in *Section 10.4*.

In the remainder of this section, I will share just a few promising broad directions for future design research on real-time teacher augmentation. Given the breadth of design explorations and findings presented in this dissertation, this section will necessarily be high-level and non-exhaustive.

[24] Where an AI's "stakes" in these human/AI systems may be taken to represent those of the learning scientists, instructional designers, and/or educational technologists involved in its design and development (cf. Buckingham Shum, 2018; Harpstead, 2017).

Over the course of my PhD research, I gradually narrowed my focus from an exploration of teachers' broader challenges and needs in AI-supported classrooms *(Chapters 1* and *4)* to the development of a specific tool, *Lumilo* (*Chapters 4* through *6*). In the process of narrowing, many promising design directions for real-time teacher augmentation were, at least temporarily, left behind (see *Chapters 1* and *9*). For example, from *Chapter 4* onward, I largely narrowed the scope of my investigations to the design of real-time support for teacher *awareness and decision-making* in AI-supported classrooms, as opposed to *system customization and control*.

Longer term, I envision exploring a much broader design space for real-time teacher augmentation (e.g., see Tables 4 through 6). Among other directions, this may involve designing forms of real-time teacher augmentation for use in a broader range of personalized learning environments, including heavily-collaborative learning settings and open-ended, project-based learning contexts. Advances in multimodal learning analytics may reduce the dependence of tools like *Lumilo* on data streams from educational software (Echeverria et al., 2018; Martinez-Maldonado, Echeverria, et al., 2018), enabling rich real-time analytics in contexts where students are highly physically mobile and/or are "untethered" from computer screens.

Relatively early in my design research with teachers, teacher autonomy emerged as a central issue in AI-supported classrooms. On the one hand, teachers expressed desires for more direct forms of decision support than are offered by existing teacher analytics tools (e.g., real-time action recommendations or greater assistance in extracting meaning from presented analytics) – especially in usage scenarios where they are under heavy time-pressure. At the same time, teachers also often expressed strong discomfort with AI systems that they perceived to be "telling them what to do" or inappropriately "judging" their behavior. While I began to explore these issues in the design of *Lumilo*, it remains an open question how real-time teacher augmentation can best be designed to balance teacher autonomy with this desire for real-time decision support. As future work delves further into the space of real-time teacher *decision-support* (e.g., Holstein et al., 2019b; VanLehn et al., 2019), I anticipate that this tension will be brought to the fore. A central challenge in navigating this balance is determining when and how an AI system should respectfully "push back" on a teacher—for example, in cases where the system is confident that certain of a teacher's instructional goals or strategies are likely to be detrimental to students' learning (Holstein et al., 2019b; Ritter et al., 2016a).

A promising and important direction for future research is to explore what effective *bidirectional communication* between a human teacher and an AI system might look like, with respect to outcomes such as teacher trust, teachers' sense of autonomy, and the instructional effectiveness of a combined teacher-AI system (cf. Holstein, 2018; Holstein et al., 2019b). My design research and classroom studies suggest that teachers' ability to interpret inferences and recommendations made by AI systems can be key not only in facilitating teacher trust in the system, but also in empowering teachers to second-guess the AI if deemed necessary. In general, however, relatively

little is known about the effects of different forms of AI interpretability and control options on users' trust, sense of autonomy, and decision-making (e.g., their ability to *productively* second-guess an AI system) (Doshi-Velez & Kim, 2017; Lipton, 2016; Poursabzi-Sangdeh, Goldstein, Hofman, Vaughan, & Wallach, 2018; Wang, Yang, Abdul, & Lim, 2019). Recent experimental work has shown that decision-makers' ability to understand AI systems' decision-making, combined with their abilities to customize and control/override systems based on this understanding may influence trust in complex ways—with heightened interpretability potentially having negative effects on trust, when provided in isolation, but a positive effect when coupled with increased user control over system behavior (e.g., Lee & Baykal, 2017). I expect that investigating a broader design space of mechanisms for teachers and AI to communicate with one another, through design studies and behavioral experiments, will ultimately help pave the way for more effective and desirable partnerships between human teachers and AI systems in the classroom.

As discussed in *Part Four* of this dissertation, many of the design questions discussed above – surrounding autonomy, interpretability, trust, and system control and customization – extend to students in addition to teachers. An important direction for future work is to explore the broader design space of *teacher–student–AI* co-orchestration – building upon findings from the present work on supports for *teacher–AI* co-orchestration, as well as prior findings from work on supports for *student–AI* joint control and regulation of learning (e.g., Bull & Kay, 2016; Long & Aleven, 2013; 2014; Roll et al., 2011). Many open questions remain regarding how best to balance between teacher, student, and AI regulation of learning during a class session. If not carefully designed, real-time teacher augmentation may risk threatening students' autonomy in AI-supported classrooms and/or hampering students' development into effective self-regulated learners (see *Chapter 9*).

Further questions arise when considering how to design for effective sharing of regulation *over successive class sessions*, or even *over successive school years*, as students' abilities to regulate their own learning develop and relationships between teachers, students, and AI agents evolve over time. This longitudinal frame points to an additional set of challenges for future work on real-time teacher augmentation and human/AI co-orchestration systems. As discussed in *Chapter 10*, whereas the bulk of my dissertation work has studied and supported teacher–student–AI interactions over relatively short timescales (i.e., one day to one week, as in *Chapter 2* and *Chapters 4* through *9*), future work should delve into complexities that may arise only over much longer timescales.

Finally, to help guide future research on systems that support human/AI co-orchestration, I present notes on a few broad ways (certainly not an exhaustive list) in which future systems might be designed to leverage complementary abilities of teachers, students, and AI systems to improve classroom instruction. These are divided into three tables (Tables 4 through 6):

"Perception" (i.e., abilities to make certain inferences), **"Perception → Action"** (i.e., mappings from perception to action), and **"Action"** (i.e., the set of actions that each agent can perform).

An exciting and important direction for future design, learning sciences, and human–computer/AI interaction research is to better understand and characterize the complementary strengths of human and automated instruction, in order to explore how they can most effectively be combined (cf. Fancsali, et al., 2018; Holstein et al., 2017a; 2017b; 2018b; 2019a; 2019b; Ritter et al., 2016b).

Table 4. Opportunities to design for human/AI complementarity in perception

	Augmenting teachers	Augmenting AI systems	Augmenting students
Extending sensing capabilities	**Help teacher sense actionable pedagogically-relevant, information to which the AI system and or students have unique access** (e.g., real-time information presented by *Lumilo* – see *Chapter 1* and *Chapters 4* through *7*) **Design to preserve teacher ability to sense actionable, pedagogically-relevant information to which they already have unique access** (e.g., through the use of heads up displays, or other peripheral interfaces – see *Chapters 1, 4, 8,* and *10;* Alavi & Dillenbourg, 2012; An et al., 2019; Bakker et al., 2016; d'Anjou et al., 2019)	**Help AI sense actionable, pedagogically-relevant information to which the teacher and/or students have unique access (e.g., via real-time polling of the teacher or students)** (future direction – see *Chapters 8 and 9*)	**Help students sense actionable information (for self-regulated learning and peer tutoring) to which the AI system and/or the teacher has unique access** (e.g., see Long & Aleven, 2013; Roll, Aleven, McLaren, & Koedinger, 2011; Walker, Rummel, & Koedinger, 2014) **Design to preserve student ability to sense actionable, pedagogically-relevant information to which they already have unique access** (future direction)
Addressing biases and limitations in diagnostic attention	**Design to guide teachers' attention towards situations that most require their further assessment from the teacher** (e.g., real-time information presented by *Lumilo* – see *Chapter 1* and *Chapters 4 through 8*) **Design to support teacher reflection on their own diagnostic monitoring behavior** (future direction – see *Chapter 2* and *9*)	**Design to help teachers and students identify and suggest learning- and teaching-related constructs that the AI system should monitor (or features to which it should attend) but does not currently** (future direction)	**Design to guide students' attention towards features of their own or peers' learning behaviors that most require their attention** (e.g., Walker, Rummel, & Koedinger, 2014) **Design to support student reflection on their own diagnostic monitoring behavior** (future direction)

Addressing biases and limitations in perception	**Design for teacher "surprise" – prioritize presenting information that conflicts with existing teacher biases (e.g., incorrect preconceptions about particular students)** (future direction – see *Chapter 1*) **Design to scaffold teachers in productively interpreting and reflecting upon the information available to them, in-the-moment** (cf. *Chapters 1, 4, and 5;* Echeverria et al., 2018; Gerritsen, Zimmerman, & Ogan, 2018) **Present AI and/or students with enough (and the right kinds of information) to enable productive second guessing of teacher inferences** (future direction – see *Chapters 1* and *9*)	**Design to help teachers notice cases where system design biases may be negatively impacting certain groups of students** (e.g., via detection of unproductive persistence, as in *Lumilo*) **Present teachers and students with enough (and the right kinds of information) to enable productive second guessing of AI inferences** (e.g., *Lumilo's* presentation of concrete examples of errors students have made, *in addition to* more heavily interpreted information such as inferred student states and skill categories; see also: Bull & Kay, 2016) **Design to help teachers and students correct or mitigate undesirable perceptual biases in AI systems (e.g., through tuning and customization of student modeling algorithms / analytics)** (future direction – see *Chapters 1, 5, and 8;* Rodríguez-Triana, Prieto, Martínez-Monés, Asensio-Pérez, & Dimitriadis, 2018)	**Design for student "surprise" – prioritize presenting information that conflicts with existing student biases (e.g., incorrect preconceptions about particular students)** (future direction) **Design to scaffold students in productively reflecting upon and interpreting the information available to them, in-the-moment** (future direction) **Present AI and/or teacher with enough (and the right kinds of) information) to enable productive second guessing of student inferences** (e.g., see Bull & Kay, 2016)

Table 5. Opportunities to design for human/AI complementarity in linking between perception and action

	Augmenting teachers	Augmenting AI systems	Augmenting students
Enhancing ability to adapt instruction based on relevant features	**Design to help teacher sense actionable pedagogically- relevant, information to which the AI has unique access** (e.g., real-time information presented by *Lumilo* – see *Chapter 1* and *Chapters 4* through *7*) **Automatically suggest effective ways for a teacher to adapt instruction based on relevant features** (future direction – see *Chapters 1*, *5*, and *8* through *10*) **Design to preserve teacher ability to sense actionable,**	**Design to help teachers and students identify and suggest learning- and teaching-related constructs that the AI system should monitor (or features to which it should attend) but does not currently** (future direction) **Design to help teachers and students suggest effective ways of adapting instruction based on these features** (future direction – see *Chapters 8 and 9*)	**Design to help students sense actionable information (for self-regulated learning and peer tutoring) to which the AI system and/or the teacher has unique access** (e.g., see Long & Aleven, 2013; Roll, Aleven, McLaren, & Koedinger, 2011; Walker, Rummel, & Koedinger, 2014) **Automatically suggest effective ways for a student to adapt instruction based on relevant features** (e.g., see Walker, Rummel, & Koedinger, 2014) **Preserve student ability to**

	pedagogically-relevant information to which they already have unique access (e.g., through the use of heads up displays, or other peripheral interfaces – see *Chapters 1, 4, 8,* and *10;* Alavi & Dillenbourg, 2012; An et al., 2019; Bakker et al., 2016; d'Anjou et al., 2019)		sense actionable, pedagogically-relevant information to which they already have unique access (future direction)
Addressing biases and limitations in decision-making	Design to guide teachers' attention towards situations that most require further intervention from the teacher (e.g., real-time information presented by *Lumilo* – see *Chapter 1* and *Chapters 4* through *7*) Automatically suggest effective ways for a teacher to respond to particular situations (future direction – see *Chapters 1, 5,* and *8* through *10*) Nudge teachers away from potentially harmful practices (future direction – see *Chapters 1, 9,* and *10*) Present AI and/or students with enough (and the right kinds of information) to enable productive second guessing of teacher decisions (future direction – see *Chapters 1* and *9*)	Design to help teachers notice cases where system design biases may be negatively impacting certain groups of students (e.g., via detection of unproductive persistence, as in *Lumilo*) Present teachers and students with enough (and the right kinds of information) to enable productive second guessing of AI decisions (future direction – see Bull & Kay, 2016) Design to help teachers and/or students correct or mitigate the impacts of undesirable pedagogical biases (or undesirable impacts of perceptual biases) in AI systems (e.g., through customization and control over AI tutors' pedagogical policies) (future direction – see *Chapters 1, 5, 8, 9,* and *10*)	Design to guide students' attention towards situations that most require further intervention from them (e.g., see Long & Aleven, 2013; Roll, Aleven, McLaren, & Koedinger, 2011; Walker, Rummel, & Koedinger, 2014) Design to nudge students away from potentially harmful practices (e.g., see Roll, Aleven, McLaren, & Koedinger, 2011; Walker, Rummel, & Koedinger, 2014) Automatically suggest effective ways for a student to respond (e.g., in a peer tutoring scenario) to particular situations (e.g., see Walker, Rummel, & Koedinger, 2014) Present AI and/or teacher with enough (and the right kinds of) information) to enable productive second guessing of student decisions (e.g., Bull & Kay, 2016)

Table 6. Designing for human/AI complementarity in action

	Augmenting teachers	Augmenting AI systems	Augmenting students
Aptitude and availability of actions	Design to support teachers in providing more effective help (future direction – see *Chapters 9 and 10*) Adaptively, dynamically delegate tasks and roles to the AI and students, where possible, in cases where the AI or students are	Design to support teachers in customizing or creating new actions for the AI tutor (e.g., enable AI systems to adaptively deliver teacher-written hints and feedback) (future direction – see *Chapters 1, 8,* and *9*) Adaptively, dynamically delegate tasks and roles to the	Design to support students in effectively regulating their own learning (e.g., see Long & Aleven, 2013; Roll et al., 2011) Design to support students in providing more effective peer tutoring support (e.g., see Walker et al., 2014)

	expected to be more effective (future direction – see *Chapters 1, 8,* and *9*)	teachers and students, in cases where the AI tutor may have reached its own pedagogical limitations, or cases where teachers or students are expected to be more effective (future direction – see *Chapters 1, 4,* and *9*; Ritter et al., 2016b; Fancsali, et al., 2018)	
Scalability and capacity	Design to support teachers in scaling their efforts (e.g., by enabling AI systems to adaptively deliver teacher-written hints and feedback) (future direction – see *Chapters 1, 8,* and *9;* and Wang et al., 2019) Adaptively, dynamically delegate tasks and roles to the AI and students, where possible, to free up time for tasks the teacher is uniquely capable of performing (future direction – see *Chapters 1, 8,* and *9*; Ritter et al., 2016b; Fancsali, et al., 2018)		Design to support students in scaling their peer tutoring efforts (e.g., by enabling AI systems to adaptively deliver student- written hints and feedback) (cf. Williams et al., 2016)

References

Aguilar, S. J. (2018). Examining the relationship between comparative and self-focused academic data visualizations in at-risk college students' academic motivation. *Journal of Research on Technology in Education*, 50(1), 84-103.

Alavi, H. S. (2011). Ambient awareness for the orchestration of collaborative problem solving. Unpublished doctoral dissertation, EPFL.

Alavi, H. S., & Dillenbourg, P. (2012). An ambient awareness tool for supporting supervised collaborative problem solving. *IEEE Transactions on Learning Technologies (TLT)*, 5(3), 264-274.

Alavi, H. S., Dillenbourg, P., & Kaplan, F. (2009). Distributed awareness for class orchestration. In *European Conference on Technology Enhanced Learning (EC-TEL 2009),* (pp. 211-225). Springer, Berlin, Heidelberg.

Aleven, V., Mclaren, B., Roll, I., & Koedinger, K. (2006). Toward meta-cognitive tutoring: A model of help seeking with a Cognitive Tutor. *International Journal of Artificial Intelligence in Education (IJAIED)*, 16(2), 101-128.

Aleven, V., McLaren, B. M., Sewall, J., Van Velsen, M., Popescu, O., Demi, S., Ringenberg, M., & Koedinger, K. R. (2016). Example-Tracing tutors: Intelligent tutor development for non-programmers. *International Journal of Artificial Intelligence in Education (IJAIED)*, 26(1), 224-269.

Aleven, V., McLaughlin, E. A., Glenn, R. A., & Koedinger, K. R. (2016). Instruction based on adaptive learning technologies. *Handbook of Research on Learning and Instruction*. Routledge.

Aleven, V., Xhakaj, F., Holstein, K., & McLaren, B. M. (2016). Developing a Teacher Dashboard For Use with Intelligent Tutoring Systems. In *IWTA@ EC-TEL* (pp. 15-23).

Aleven, V., Roll, I., McLaren, B. M., & Koedinger, K. R. (2016). Help helps, but only so much: Research on help seeking with intelligent tutoring systems. *International Journal of Artificial Intelligence in Education (IJAIED)*, 26(1), 205-223.

Aiken, R. M., & Epstein, R. G. (2000). Ethical guidelines for AI in education: Starting a conversation. *International Journal of Artificial Intelligence in Education (IJAIED)*, 11, 163-176.

Alkhatib, A., and Bernstein, M. 2019. Street–Level Algorithms: A Theory at the Gaps Between Policy and Decisions. In *CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2019)*, May 4–9, 2019, Glasgow, Scotland Uk. ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3290605.3300760

van Alphen, E., & Bakker, S. (2016). Lernanto: using an ambient display during differentiated instruction. In *CHI 2016 Conference on Human Factors in Computing Systems (CHI 2016)*, 7-12 May 2016, San Jose, California, US. ACM.

Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P.N., Inkpen, K., Teevan, J., Kikin-Gil, R., & Horvitz, E. 2019. Guidelines for Human-AI Interaction. In *CHI 2019 Conference on Human Factors in Computing Systems Proceedings (CHI 2019),* May 4–9, 2019, Glasgow, Scotland UK. ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3290605.3300233

An, P., Bakker, S., Ordanovski, S., Taconis, R., & Eggen, B. (2018). ClassBeacons: Designing Distributed Visualization of Teachers' Physical Proximity in the Classroom. In *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction (TEI 2018),* (pp. 357-367). ACM.

An, P., Bakker, S., Ordanovski, S., Taconis, R., Paffen, C. L., & Eggen, B. (2019). Unobtrusively Enhancing Reflection-in-Action of Teachers through Spatially Distributed Ambient Information. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI 2019)* (p. 91). ACM.

Anderson, J. R. (2009). How can the human mind occur in the physical universe?. Oxford University Press.

Baker, C., Saxe, R., & Tenenbaum, J. B. (2006). Bayesian models of human action understanding. In *Advances in Neural Information Processing Systems (NeurIPS 2006)* (pp. 99-106).

Baker, R. S. (2007). Modeling and understanding students' off-task behavior in intelligent tutoring systems. In *CHI 2007 Conference on Human Factors in Computing Systems Proceedings (CHI 2007),* (pp. 1059-1068). ACM.

Baker, R. S. (2011). Gaming the system: A retrospective look. *Philippine Computing Journal*, 6(2), 9-13.

Baker, R. S. (2016). Stupid tutoring systems, intelligent humans. *International Journal of Artificial Intelligence in Education (IJAIED)*, 26(2), 600-614.

Baker, R. S. (2019). Challenges for the Future of Educational Data Mining: The Baker Learning Analytics Prizes. *Journal of Educational Data Mining (JEDM)*, 11(1), 1-17.

Baker, R. D., Corbett, A. T., Koedinger, K. R., Evenson, S., Roll, I., Wagner, A. Z., Naim, M., Raspat, J., Baker, D. J., & Beck, J. E. (2006). Adapting to when students game an intelligent tutoring system. In *International Conference on Intelligent Tutoring Systems (ITS 2006),* (pp. 392-401). Springer, Berlin, Heidelberg.

Baker, R. S., Corbett, A. T., Koedinger, K. R., & Wagner, A. Z. (2004). Off-task behavior in the cognitive tutor classroom: when students game the system. In *Proceedings of the 2004 CHI Conference on Human Factors in Computing Systems (CHI 2004)*, 383-390. ACM.

Baker, R.S.J.d., Corbett, A.T., Roll, I., Koedinger, K.R. Aleven, V., Cocea, M., Hershkovitz, A., de Carvalho, A.M.J.B., Mitrovic, A., Mathews, M. (2013). Modeling and Studying Gaming the System with Educational Data Mining. In *Azevedo, R., & Aleven, V. (Eds.) International Handbook of Metacognition and Learning Technologies*, 97-116. New York, NY: Springer.

Baker, R.S.J.d, Gowda, S.M., Wixon, M., Kalka, J., Wagner, A.Z., Salvi, A., Aleven, V., Kusbit, G., Ocumpaugh, J., & Rossi, L. (2012). Towards Sensor-Free Affect Detection in Cognitive Tutor Algebra. In *Proceedings of the 2012 International Conference on Educational Data Mining (EDM 2012)*, (pp. 126-133).

Bakker, S., Hausen, D., & Selker, T. (Eds.). (2016). Peripheral Interaction: Challenges and Opportunities for HCI in the Periphery of Attention. Springer.

Bartneck, C., & Forlizzi, J. (2004). A design-centred framework for social human-robot interaction. In *Proceedings of the 13th IEEE International Workshop on Robot and Human Interactive Communication (IEEE Catalog No. 04TH8759)* (pp. 591-594). IEEE.

Baumer, E. P. S. (2017). Toward human-centered algorithm design. *Big Data & Society*, 4(2), 2053951717718854.

Beck, J. E., & Gong, Y. (2013). Wheel-spinning: Students who fail to master a skill. In *International Conference on Artificial Intelligence in Education (AIED 2013),* (pp. 431-440). Springer, Berlin, Heidelberg.

Beyer, H., & Holtzblatt, K. (1997). Contextual design: Defining customer-centered systems. Elsevier.

Bingham, A. J., Pane, J. F., Steiner, E. D., & Hamilton, L. S. (2018). Ahead of the curve: Implementation challenges in personalized learning school models. *Educational Policy*, 32(3), 454-489.

Blikstein, P. (2018). Time to Make Hard Choices for AI in Education. Keynote talk at the *2018 International Conference on Artificial Intelligence in Education (AIED 2018)*.

Blessing, S. B., Aleven, V., Gilbert, S. B., Heffernan, N. T., Matsuda, N., & Mitrovic, A. (2015). Authoring example-based tutors for procedural tasks. *Design Recommendations for Intelligent Tutoring Systems*, 3, 71-93.

Bohn, D. (2019). Microsoft's HoloLens 2: A $3,600 mixed reality headset for the factory, not the living room. *Verge*. Retrieved on July 11, 2019 from https://www.theverge.com/2019/2/24/18235460/microsoft-hololens-2-price-specs-mixed-reality-ar-vr-business-work-features-mwc-2019

Borko, H., Roberts, S. A., & Shavelson, R. (2008). Teachers' decision making: From Alan J. Bishop to today. In *Critical Issues in Mathematics Education*, 37-67. Springer US.

du Boulay, B., Luckin, R., & del Soldato, T. (1999). The plausibility problem: Human teaching tactics in the 'hands' of a machine. In *Proceedings of the 1999 International Conference on Artificial Intelligence in Education (AIED 1999)*. (pp. 225-232). IOS Press Amsterdam.

Broderick, Z., O'Connor, C., Mulcahy, C., Heffernan, N., & Heffernan, C. (2011). Increasing parent engagement in student learning using an intelligent tutoring system. *Journal of Interactive Learning Research (JILR)*, 22(4), 523-550.

Buckingham Shum, S. (2018). Transitioning Education's Knowledge Infrastructure. Keynote at the *2018 International Conference of the Learning Sciences (ICLS 2018)*. *ISLS.*

Buckingham Shum, S., Ferguson, R., Martinez-Maldonado, R. (2019). Human-Centered Learning Analytics. In *Journal of Learning Analytics (JLA)*. SoLAR.

Bull, S., & Kay, J. (2016). SMILI☺: A framework for interfaces to learning data in Open Learner Models, learning analytics and related fields. *International Journal of Artificial Intelligence in Education (IJAIED)*, 26(1), 293-331.

Bulger, M. (2016). Personalized learning: The conversations we're not having. *Data and Society*, 22.

Cairns, P., & Cox, A. L. (Eds.). (2008). Research methods for human-computer interaction (Vol. 12). Cambridge: Cambridge University Press.

Cen, H., Koedinger, K., & Junker, B. (2006). Learning factors analysis – A general method for cognitive model evaluation and improvement. In *Proceedings of the 2006 International Conference on Intelligent Tutoring Systems (ITS 2006)*, (pp. 164-175). Springer, Berlin, Heidelberg.

Chen, B., & Zhu, H. (2019). Towards value-sensitive learning analytics design. In *Proceedings of the Ninth International Learning Analytics & Knowledge Conference (LAK 2019).* ACM.

Clow, D. (2012). The learning analytics cycle. In *Proceedings of the Second International Learning Analytics & Knowledge Conference (LAK 2012)*. ACM.

Cohen, J. (1992). A power primer. *Psychological Bulletin*, 112(1), pp. 155-159.

Corbett, A. T., & Anderson, J. R. (1995). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-adapted Interaction (UMUAI 1995)*, 4(4), 253-278.

Corbett, A. T., Koedinger, K. R., & Hadley, W. H. (2001). Cognitive Tutors: From the research classroom to all classrooms. *Technology Enhanced Learning: Opportunities for Change*, 235-263.

Cramer, H., Holstein, K., Wortman Vaughan, J., Daumé III, H., Dudík, M., Wallach, H., Reddy, S., & Garcia-Gathright, J. (2019). Challenges of incorporating algorithmic fairness into industry practice. Tutorial at the *ACM Conference on Fairness, Accountability, and Transparency (FAT* 2019)*. ACM.

Crandall, B., Klein, G., Klein, G. A., & Hoffman, R. R. (2006). Working minds: A practitioner's guide to cognitive task analysis. MIT Press.

d'Anjou, B., Bakker, S., An, P., & Bekker, T. (2019). How Peripheral Data Visualisation Systems Support Secondary School Teachers during VLE-Supported Lessons. In *Proceedings of the 2019 on Designing Interactive Systems Conference (DIS 2019)* (pp. 859-870). ACM.

Davidoff, S., Lee, M. K., Dey, A. K., & Zimmerman, J. (2007). Rapidly exploring application design through speed dating. In *International Conference on Ubiquitous Computing (UbiComp 2007)* (pp. 429-446). Springer, Berlin, Heidelberg.

Davies, N., Langheinrich, M., Maes, P., & Rekimoto, J. (2018). Augmenting Humans. *IEEE Pervasive Computing,* 17(2), 9-10.

Dennerlein, S., Kowald, D., Pammer-Schindler, V., Lex, E., & Ley, T. (2018). Simulation-based co-creation of algorithms. In *CEUR Workshop Proceedings (Vol. 2190)*. RWTH Aachen.

Desmarais, M. C., & Baker, R. S. (2012). A review of recent advances in learner and skill modeling in intelligent learning environments. *User Modeling and User-Adapted Interaction (UMUAI, 2012)*, 22(1-2), 9-38.

Diana, N., Eagle, M., Stamper, J., Grover, S., Bienkowski, M., & Basu, S. (2017). An instructor dashboard for real-time analytics in interactive programming assignments. In *Proceedings of the 2017 International Learning Analytics & Knowledge Conference (LAK 2017)* (pp. 272-279). ACM.

Dillahunt, T. R., Lam, J., Lu, A., & Wheeler, E. (2018). Designing Future Employment Applications for Underserved Job Seekers: A Speed Dating Study. In *Proceedings of the 2018 Designing Interactive Systems Conference (DIS 2018),* (pp. 33-44). ACM.

Dillenbourg, P., & Jermann, P. (2010). Technology for classroom orchestration. In *New Science of Learning* (pp. 525-552). Springer, New York, NY.

Dillenbourg, P., Prieto, L. P., & Olsen, J. K. (2018). Classroom orchestration. In *International Handbook of the Learning Sciences* (pp. 180-190). Routledge.

D'Mello, S. K., Lehman, B., & Graesser, A. (2011). A motivationally supportive affect-sensitive AutoTutor. In *New Perspectives on Affect and Learning Technologies* (pp. 113-126). Springer, New York, NY.

D'Mello, S., Picard, R. W., & Graesser, A. (2007). Toward an affect-sensitive AutoTutor. *IEEE Intelligent Systems*, 22(4), 53-61.

Doroudi, S., Aleven, V., & Brunskill, E. (2017). Robust evaluation matrix: Towards a more principled offline exploration of instructional policies. In *Proceedings of the Fourth (2017) ACM Conference on Learning@ Scale (L@S 2017),* (pp. 3-12). ACM.

Doroudi, S., Holstein, K., Aleven, V., & Brunskill, E. (2015). Towards Understanding How to Leverage Sense-making, Induction/refinement, and Fluency to Improve Robust Learning. In *Proceedings of the Eighth International Conference on Educational Data Mining. (EDM 2015)*. IEDMS.

Doroudi, S., Holstein, K., Aleven, V., & Brunskill, E. (2016). Sequence Matters, But How Exactly? A Method for Evaluating Activity Sequences from Data. In *Proceedings of the Ninth International Conference on Educational Data Mining (EDM 2016)*. (pp. 70-77). IEDMS.

Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint* arXiv:1702.08608.

Dove, G., Halskov, K., Forlizzi, J., & Zimmerman, J. (2017). UX design innovation: Challenges for working with machine learning as a design material. In *CHI 2017 Conference on Human Factors in Computing Systems Proceedings (CHI 2017),* (pp. 278-288). ACM.

Duckworth, P., Graham, L., & Osborne, M. A. (2019). Inferring Work Task Automatability from AI Expert Evidence. In *AAAI/ACM Conference on Artificial Intelligence, Ethics, and Society (AIES 2019)*. AAAI.

Echeverria, V., Martinez-Maldonado, R., Granda, R., Chiluiza, K., Conati, C., & Shum, S. B. (2018). Driving data storytelling from learning design. In *Proceedings of the 2018 International Conference on Learning Analytics and Knowledge (LAK 2018)*, (pp. 131-140). ACM.

Echeverria, V., Martinez-Maldonado, R., Power, T., Hayes, C., & Shum, S. B. (2018). Where Is the Nurse? Towards Automatically Visualising Meaningful Team Movement in Healthcare Education. In

International Conference on Artificial Intelligence in Education (AIED 2018), (pp. 74-78). Springer, Cham.

Evenson, S. (2006). Directed storytelling: Interpreting experience for design. *Design Studies: Theory and research in graphic design*, 231-240.

Fancsali, S. (2014). Causal discovery with models: behavior, affect, and learning in cognitive tutor algebra. In *Proceedings of the 2014 Conference on Educational Data Mining (EDM 2014)* (pp. 28-35).

Fancsali, S. E., Ritter, S., Stamper, J., & Nixon, T. (2013). Toward "hyper-personalized" Cognitive Tutors. In *AIED 2013 Workshops Proceedings (AIED 2013),* (Vol. 7, pp. 71-79).

Fancsali, S. E., Yudelson, M. V., Berman, S. R., & Ritter, S. (2018). Intelligent Instructional Hand Offs. In *Proceedings of the 2018 International Conference on Educational Data Mining (EDM 2018)*. IEDMS.

Feng, M., & Heffernan, N. T. (2007). Towards live informing and automatic analyzing of student learning: Reporting in the ASSISTment system. *Journal of Interactive Learning Research (JILR)*, 18(2), 207-230.

Forlizzi, J., & Zimmerman, J. (2013). Promoting service design as a core practice in interaction design. In *Proceedings of the 2013 IASDR World Conference on Design Research.*

Frey, C. B., & Osborne, M. A. (2017). The future of employment: how susceptible are jobs to computerisation?. *Technological Forecasting and Social Change*, 114, 254-280.

Friedman, B., Kahn, P. H., & Borning, A. (2008). Value sensitive design and information systems. *The Handbook of Information and Computer Ethics*, 69-101.

Gerritsen, D., Zimmerman, J., & Ogan, A. (2018). Towards a Framework for Smart Classrooms that Teach Instructors to Teach. In *Proceedings of the 13th International Conference of the Learning Sciences (ICLS 2018)*. ISLS.

Grawemeyer, B., Holmes, W., Gutiérrez-Santos, S., Hansen, A., Loibl, K., & Mavrikis, M. (2015). Light-bulb moment?: Towards adaptive presentation of feedback based on students' affective state. In *Proceedings of the 2015 International Conference on Intelligent User Interfaces (IUI 2015)*, (pp. 400-404). ACM.

Gugenheimer, J., Mai, C., McGill, M., Williamson, J., Steinicke, F., & Perlin, K. (2019). Challenges using head-mounted displays in shared and social spaces. Workshop at *the 2019 CHI Conference on Human Factors in Computing Systems (CHI 2019)*. ACM.

Hagerty, G., & Smith, S. (2005). Using the web-based interactive software ALEKS to enhance college algebra. *Mathematics & Computer Education*, 39(3).

Hanington, B., & Martin, B. (2012). Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions. Rockport Publishers.

Harpstead, E. (2017). Projective Replay Analysis: A Reflective Approach for Aligning Educational Games to Their Goals. *Unpublished doctoral dissertation.* Human-Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA.

Harrison, C. (2018). The HCI innovator's dilemma. *Interactions*, 25(6), 26-33.

Heer, J. (2019). Agency plus automation: Designing artificial intelligence into interactive systems. *Proceedings of the National Academy of Sciences*, 116(6), 1844-1850.

Heffernan, N. T., Ostrow, K. S., Kelly, K., Selent, D., Van Inwegen, E. G., Xiong, X., & Williams, J. J. (2016). The Future of Adaptive Learning: Does the Crowd Hold the Key? *International Journal of Artificial Intelligence in Education (IJAIED)*, 26(2), 615-644.

Helms, K., Brown, B., Sahlgren, M., & Lampinen, A. (2018). Design Methods to Investigate User Experiences of Artificial Intelligence. In *2018 AAAI Spring Symposium Series.*

Hiniker, A., Sobel, K., & Lee, B. (2017). Co-designing with preschoolers using fictional inquiry and comicboarding. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI 2017), (pp. 5767-5772)*. ACM.

Holstein, K. (2018). Towards teacher-AI hybrid systems for K-12 education. In *Companion Proceedings of the Eighth International Learning Analytics & Knowledge Conference (LAK 2018)*. ACM.

Holstein, K. & Doroudi, S. (2019). Fairness and equity in learning analytics systems (FairLAK). In *Companion Proceedings of the Ninth International Learning Analytics & Knowledge Conference (LAK 2019)*. ACM.

Holstein, K., Hong, G., Tegene, M., McLaren, B. M., & Aleven, V. (2018). The classroom as a dashboard: Co-designing wearable cognitive augmentation for K-12 teachers. In *Proceedings of the Eighth International Learning Analytics & Knowledge Conference (LAK 2018)*. (pp. 79-88). ACM.

Holstein, K., McLaren, B. M., & Aleven, V. (2017a). SPACLE: Investigating learning across virtual and physical spaces using spatial replays. In *Proceedings of the Seventh International Learning Analytics & Knowledge Conference (LAK 2017)*. (pp. 358-367). ACM.

Holstein, K., McLaren, B. M., & Aleven, V. (2017b). Intelligent tutors as teachers' aides: Exploring teacher needs for real-time analytics in blended classrooms. In *Proceedings of the Seventh International Learning Analytics & Knowledge Conference (LAK 2017)*. (pp. 257-266). ACM.

Holstein, K., McLaren, B. M., & Aleven, V. (2018a). Informing the design of teacher awareness tools through Causal Alignment Analysis. In *Proceedings of the 13th International Conference of the Learning Sciences (ICLS 2018)*. ISLS.

Holstein, K., McLaren, B. M., & Aleven, V. (2018b). Student learning benefits of a mixed-reality teacher awareness tool in AI-enhanced classrooms. In *Proceedings of the 19th International Conference on Artificial Intelligence in Education (AIED 2018)*. (pp. 154-168). Springer, Cham.

Holstein, K., McLaren, B.M., & Aleven, V. (2019a). Co-designing a real-time classroom orchestration tool to support teacher-AI complementarity. In *Journal of Learning Analytics (JLA)*. Society for Learning Analytics Research (SoLAR).

Holstein, K., McLaren, B. M., & Aleven, V. (2019b). Designing for Complementarity: Teacher and Student Needs for Orchestration Support in AI-Enhanced Classrooms. In *Proceedings of the International Conference on Artificial Intelligence in Education (AIED 2019)* (pp. 157-171). Springer, Cham.

Holstein, K., Yu, Z., Popescu, O., Sewall, J., McLaren, B. M., & Aleven, V. (2018). Opening up an intelligent tutoring system development environment for extensible student modeling. In *Proceedings of the 19th International Conference on Artificial Intelligence in Education (AIED 2018)*. (pp. 169-183). Springer, Cham.

Holstein, K., Wortman Vaughan, J., Daumé III, H., Dudík, M., Wallach, H. (2018). Opportunities for machine learning research to support fairness in industry practice. In the *2018 Workshop on Critiquing and Correcting Trends in Machine Learning at the Conference on Neural Information Processing Systems (NeurIPS 2018)*.

Holstein, K., Wortman Vaughan, J., Daumé III, H., Dudík, M., Wallach, H. (2019). Improving fairness in machine learning systems: What do industry practitioners need? In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2019)*. ACM.

Holstein, K., Xhakaj, F., Aleven, V., McLaren, B. (2016). Luna: A Dashboard for Teachers Using Intelligent Tutoring Systems. In *Proceedings of the 4th International Workshop on Teaching Analytics at the European Conference on Technology-Enhanced Learning (EC-TEL 2016)*.

Horvitz, E. (1999). Principles of mixed-initiative user interfaces. In *Proceedings of the 1999 CHI Conference on Human Factors in Computing Systems* (pp. 159-166). ACM.

Huber, A., Lammer, L., Weiss, A., & Vincze, M. (2014). Designing adaptive roles for socially assistive robots: a new method to reduce technological determinism and role stereotypes. *Journal of Human-Robot Interaction*, 3(2), 100-115.

Hutchinson, H., Mackay, W., Westerlund, B., Bederson, B. B., Druin, A., Plaisant, C., ... & Roussel, N. (2003. Technology probes: inspiring design for and with families. In *Proceedings of the 2003 CHI Conference on Human Factors in Computing Systems* (pp. 17-24). ACM.

Inkpen, K., De Choudhury, M., Chancellor, S., Veale, M., & Baumer, E. P. S. (2019). Where is the human? Bridging the gap between AI and HCI. Workshop at the *2019 CHI Conference on Human Factors in Computing Systems (CHI 2019)*.

Jivet, I., Scheffel, M., Drachsler, H., & Specht, M. (2017). Awareness is not enough: pitfalls of learning analytics dashboards in the educational practice. In *European Conference on Technology Enhanced Learning (EC-TEL 2017)* (pp. 82-96). Springer, Cham.

Kai, S., Ma. Victoria Almeda, Baker, R. S., Shechtman, N., Heffernan, C., & Heffernan, N. T. (2017). Modeling Wheel-spinning and Productive Persistence in Skill Builders. *Pre-print.*

Kamar, E. (2016). Directions in Hybrid Intelligence: Complementing AI Systems with Human Intelligence. In *Proceedings of the 2016 International Joint Conference on Artificial Intelligence* (*IJCAI 2016)* (pp. 4070-4073).

Karumbaiah, S., Ocumpaugh, J., & Baker R.S. (2019). The Influence of School Demographics on the Relationship Between Students' Help-Seeking Behavior and Performance and Motivational Measures. In *Proceedings of the 2019 International Conference on Educational Data Mining (EDM 2019)*. 16. IEDM.

Käser, T., Klingler, S., & Gross, M. (2016). When to stop?: towards universal instructional policies. In *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge (LAK 2016),* (pp. 289-298). ACM.

Kay, J. (2000). Stereotypes, student models and scrutability. In *International Conference on Intelligent Tutoring Systems (ITS 2000),* (pp. 19-30). Springer, Berlin, Heidelberg.

Kelly, K., Heffernan, N., Heffernan, C., Goldman, S., Pellegrino, J., & Goldstein, D. S. (2013). Estimating the effect of web-based homework. In *International Conference on Artificial Intelligence in Education (AIED 2013)* (pp. 824-827). Springer, Berlin, Heidelberg.

Kery, M. B., & Myers, B. A. (2017). Exploring exploratory programming. In *Proceedings of the 2017 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC 2017)* (pp. 25-29). IEEE.

Khachatryan, G. A., Romashov, A. V., Khachatryan, A. R., Gaudino, S. J., Khachatryan, J. M., Guarian, K. R., & Yufa, N. V. (2014). Reasoning Mind Genie 2: An intelligent tutoring system as a vehicle for international transfer of instructional methods in mathematics. *International Journal of Artificial Intelligence in Education (IJAIED),* 24(3), 333-382.

Khajah, M., Lindsey, R. V., and Mozer, M. C. (2015). How Deep is Knowledge Tracing? In *Proceedings of the 2015 Conference on Educational Data Mining (EDM 2015),* (pp. 94–101).

Kitto, K., Shum, S. B., & Gibson, A. (2018). Embracing imperfection in learning analytics. In *Proceedings of the 8th International Conference on Learning Analytics and Knowledge (LAK 2018),* (pp. 451-460). ACM.

Koedinger, K. R., Baker, R. S., Cunningham, K., Skogsholm, A., Leber, B., & Stamper, J. (2010). A data repository for the EDM community: The PSLC DataShop. *Handbook of Educational Data Mining*, 43, 43-56.

Koedinger, K. R., Stamper, J. C., McLaughlin, E. A., & Nixon, T. (2013). Using data-driven discovery of better student models to improve student learning. In *International Conference on Artificial Intelligence in Education (AIED 2013),* (pp. 421-430). Springer, Berlin, Heidelberg.

Kulik, J. A., & Fletcher, J. D. (2016). Effectiveness of intelligent tutoring systems: a meta-analytic review. *Review of Educational Research (RER)*, 86(1), 42-78.

Kulkarni, C. (2019). Design Perspectives of Learning at Scale: Scaling Efficiency and Empowerment. In *Proceedings of the Sixth (2019) ACM Conference on Learning@ Scale (L@S 2019)* (p. 18). ACM.

Kusunoki, D., Sarcevic, A., Zhang, Z., & Yala, M. (2015). Sketching awareness: A participatory study to elicit designs for supporting ad hoc emergency medical teamwork. *Computer Supported Cooperative Work (CSCW 2015),* 24(1), 1-38.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences (BBS)*, 40.

Landén, D., Heintz, F., & Doherty, P. (2010). Complex task allocation in mixed-initiative delegation: A UAV case study. In *International Conference on Principles and Practice of Multi-Agent Systems* (pp. 288-303). Springer, Berlin, Heidelberg.

Lee, M. K., & Baykal, S. (2017). Algorithmic mediation in group decisions: Fairness perceptions of algorithmically mediated vs. discussion-based social division. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW 2017)*, (pp. 1035-1048). ACM.

Lee, M. K., Kusbit, D., Kahng, A., Kim, J. T., Yuan, X., Chan, A., Noothigattu, R., See, D., Psomas, C. A. & Procaccia, A. D. (2018). WeBuildAI: Participatory Framework for Fair and Efficient Algorithmic Governance. *Preprint.*

Lesta, L., & Yacef, K. (2002). An intelligent teaching assistant system for logic. In *International Conference on Intelligent Tutoring Systems (ITS 2002)* (pp. 421-431). Springer, Berlin, Heidelberg.

Lipton, Z. C. (2016). The mythos of model interpretability. arXiv preprint arXiv:1606.03490.

Liu, R., Davenport, J., & Stamper, J. (2016). Beyond log files: Using multi-modal data streams towards data-driven KC model improvement. In *Proceedings of the International Conference on Educational Data Mining (EDM 2016)*, 436-441.

Liu, R., & Koedinger, K. R. (2017). Closing the loop: Automated data-driven cognitive model discoveries lead to improved instruction and learning gains. *Journal of Educational Data Mining (JEDM)*, 9(1), 25-41.

Liu, R., Stamper, J. C., & Davenport, J. (2018). A Novel method for the in-depth multimodal analysis of student learning trajectories in intelligent tutoring systems. *Journal of Learning Analytics (JLA)*, 5(1), 41-54.

Liu, Z., Pataranutaporn, V., Ocumpaugh, J., & Baker, R. (2013). Sequences of frustration and confusion, and learning. In *Proceedings of the 2013 Conference on Educational Data Mining (EDM 2013),* (pp. 114-120).

Long, Y. (2015). Supporting learner-controlled problem selection in intelligent tutoring systems. *Unpublished doctoral dissertation.* Human-Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA.

Long, Y., & Aleven, V. (2013). Supporting students' self-regulated learning with an open learner model in a linear equation tutor. In *International Conference on Artificial Intelligence in Education (AIED 2013),* (pp. 219-228). Springer, Berlin, Heidelberg.

Long, Y., & Aleven, V. (2016). Supporting shared student/system control over problem selection with an open learner model in a linear equation tutor. In *Proceedings of the 2016 International Conference on Intelligent Tutoring Systems (ITS 2016)*, 90-100. Springer International Publishing.

Long, Y., & Aleven, V. (2017). Educational game and intelligent tutoring system: A classroom study and comparative design analysis. In *ACM Transactions on Computer-Human Interaction (TOCHI)*, 24(3), 20.

Long, Y., Aman, Z., & Aleven, V. (2015). Motivational design in an intelligent tutoring system that helps students make good task selection decisions. In *Proceedings of the 2015 International Conference on Artificial Intelligence in Education*, (pp. 226-236). Springer, Cham.

Long, Y., Holstein, K., & Aleven, V. (2018). What exactly do students learn when they practice equation solving? Refining knowledge components with the Additive Factors Model. In *Proceedings of the Eighth International Learning Analytics & Knowledge Conference (LAK 2018),* (pp. 399-408). ACM.

Lovejoy, J., (2018). The UX of AI. https://design.google/library/ux-ai/. Google.

Lubars, B., & Tan, C. (2019). Ask not what AI can do, but what AI should do: Towards a framework of task delegability. arXiv preprint arXiv:1902.03245.

Lucas, C. G., Holstein, K., & Kemp, C. (2014). Discovering Hidden Causes using Statistical Evidence. In *Proceedings of the 36th Annual Conference of the Cognitive Science Society (CogSci 2014).*

Lucas, G. M., Gratch, J., King, A., & Morency, L. P. (2014). It's only a computer: Virtual humans increase willingness to disclose. *Computers in Human Behavior*, 37, 94-100.

Luckin, R., & Clark, W. (2011). More than a game: The participatory design of contextualised technology-rich learning experiences with the ecology of resources. *Journal of e-learning and Knowledge Society*, 7(3), 33-50.

van Leeuwen, A. (2015). Learning analytics to support teachers during synchronous CSCL: Balancing between overview and overload. *Journal of Learning Analytics (JLA)*, 2(2), 138-162.

van Leeuwen, A., Rummel, N., Aleven, V., Gal, K., Holstein, K., Knoop-van Campen, C., McLaren, B.M., Molenaar, I., Prusak, N., Schwarz, B., Segal, A., Swidan, O., & Wise, A. (2018). Orchestration tools for teachers in the context of mathematics: What information do teachers need and what do they do with it? In *Proceedings of the 13th International Conference of the Learning Sciences (ICLS 2018)*. ISLS.

Ma, W., Adesope, O. O., Nesbit, J. C., & Liu, Q. (2014). Intelligent tutoring systems and learning outcomes: A meta-analysis. *Journal of Educational Psychology*, 106(4), 901.

MacLellan, C. J., Harpstead, E., Patel, R., & Koedinger, K. R. (2016). The Apprentice Learner Architecture: Closing the Loop between Learning Theory and Educational Data. In *Proceedings of the 2016 International Conference on Educational Data Mining (EDM 2016)*. IEDMS.

MacLellan, C. J., Koedinger, K. R., & Matsuda, N. (2014). Authoring tutors with SimStudent: An evaluation of efficiency and model quality. In *Proceedings of the 2014 International Conference on Intelligent Tutoring Systems (ITS 2014)*, (pp. 551-560). Springer, Cham.

Mahani, M. A. R. Y. A. M., & Eklundh, K. S. (2009). A survey of the relation of the task assistance of a robot to its social role. *Communication KCSa Royal Institute of Technology: Stockholm, Sweden.*

Manolev, J., Sullivan, A., & Slee, R. (2018). The datafication of discipline: ClassDojo, surveillance and a performative classroom culture. *Learning, Media and Technology*, 1-16.

Martinez-Maldonado, R. (2019). I Spent More Time with that Team: Making Spatial Pedagogy Visible Using Positioning Sensors. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge (LAK 2019),* (pp. 21-25). ACM.

Martinez-Maldonado, R. (2016). Seeing learning analytics tools as orchestration technologies: Towards supporting learning activities across physical and digital spaces. In *First International Workshop on Learning Analytics Across Physical and Digital Spaces co-located with the 2016 International Conference on Learning Analytics & Knowledge (LAK 2016)*. CEUR.

Martinez-Maldonado, R., Clayphan, A., Yacef, K., & Kay, J. (2015). MTFeedback: providing notifications to enhance teacher awareness of small group work in the classroom. *IEEE Transactions on Learning Technologies (TLT)*, 8(2), 187-200.

Martinez-Maldonado, R., Echeverria, V., Santos, O. C., Santos, A. D. P. D., & Yacef, K. (2018). Physical learning analytics: A multimodal perspective. In *Proceedings of the 8th International Conference on Learning Analytics and Knowledge (LAK 2019)* (pp. 375-379). ACM.

Martinez-Maldonado, R., Kay, J., Yacef, K., & Schwendimann, B. (2012). An interactive teacher's dashboard for monitoring groups in a multi-tabletop learning environment. In *International Conference on Intelligent Tutoring Systems (ITS 2012)* (pp. 482-492). Springer, Berlin, Heidelberg.

Martinez-Maldonado, R., Pardo, A., Mirriahi, N., Yacef, K., Kay, J., & Clayphan, A. (2016). LATUX: An Iterative Workflow for Designing, Validating, and Deploying Learning Analytics Visualizations. *Journal of Learning Analytics (JLA)*, 2(3), 9-39.

Matuk, C., Gerard, L., Lim-Breitbart, J., & Linn, M. C. (2016). Teachers' reflections on the uses of real-time data in their instruction. Poster presented at the *Annual Meeting of the American Educational Research Association (AERA),* Washington, DC, USA.

Mavrikis, M., Gutierrez-Santos, S., & Poulovassilis, A. (2016). Design and evaluation of teacher assistance tools for exploratory learning environments. In *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge (LAK 2016)* (pp. 168-172). ACM.

McGraw Hill (2019). ALEKS reports. https://www.aleks.com/k12/guide_reports.

McLaren, B.M., Scheuer, O., & Mikšátko, J. (2010). Supporting collaborative learning and e-Discussions using artificial intelligence techniques. *International Journal of Artificial Intelligence in Education (IJAIED),* 20(1), 1-46.

Metcalfe, J. (2017) Learning from errors." In *Annual Review of Psychology (ARP)* 68(2017), 465-489.

Microsoft (2017). HoloLens - Microsoft. https://www.microsoft.com/en-us/hololens.

Miller, J. K., Friedman, B., Jancke, G., & Gill, B. (2007). Value tensions in design: the value sensitive design, development, and appropriation of a corporation's groupware system. In *Proceedings of the 2007 International ACM Conference on Supporting Group Work (GROUP 2007)*, (pp. 281-290). ACM.

Miller, W. L., Baker, R. S., Labrum, M. J., Petsche, K., Liu, Y. H., & Wagner, A. Z. (2015). Automated detection of proactive remediation by teachers in Reasoning Mind classrooms. In P*roceedings of the Fifth International Conference on Learning Analytics and Knowledge (LAK 2015),* (pp. 290-294). ACM.

Mitchell, V., Ross, T., May, A., Sims, R., & Parker, C. (2016). Empirical investigation of the impact of using co-design methods when generating proposals for sustainable travel solutions. *CoDesign*, 12(4), 205-220.

Mitrovic, A., Martin, B., Suraweera, P., Zakharov, K., Milik, N., Holland, J., & McGuigan, N. (2009). ASPIRE: an authoring system and deployment environment for constraint-based tutors. *International Journal of Artificial Intelligence in Education (IJAIED)*, 19(2), 155-188.

Molenaar, I., Horvers, A., & Baker, R. S. (2019). Towards hybrid human-system regulation: Understanding children'SRL support needs in blended classrooms. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge (LAK 2019),* (pp. 471-480). ACM.

Molenaar, I., & Knoop-van Campen, C. (2017). Teacher dashboards in practice: Usage and impact. In *European Conference on Technology Enhanced Learning (EC-TEL 2017),* (pp. 125-138). Springer, Cham.

Moraveji, N., Li, J., Ding, J., O'Kelley, P., & Woolf, S. (2007). Comicboarding: Using comics as proxies for participatory design with children. In *2007 CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2007),* (pp. 1371-1374). ACM.

Morelli, N. (2003). Product-service systems, a perspective shift for designers: A case study: the design of a telecentre. *Design Studies*, 24(1), 73-99.

Mulligan, D. K., & King, J. (2011). Bridging the gap between privacy and design. *U. Pa. J. Const. L.*, 14, 989.

Muñoz-Cristóbal, J. A., Prieto, L. P., Asensio-Pérez, J. I., Jorrín-Abellán, I. M., Martínez-Monés, A., & Dimitriadis, Y. (2013). Sharing the burden: Introducing student-centered orchestration in across-spaces learning situations. In *European Conference on Technology Enhanced Learning (EC-TEL 2013),* (pp. 621-622). Springer, Berlin, Heidelberg.

Nelson-Le Gall, S. (1981). Help-seeking: An understudied problem-solving skill in children. *Developmental Review*, 1(3), 224-246.

Nye, B. D. (2014). Barriers to ITS adoption: A systematic mapping study. In *Proceedings of the International Conference on Intelligent Tutoring Systems (ITS 2014)*, 583-590. Springer International Publishing.

Ocumpaugh, J. (2015). Baker Rodrigo Ocumpaugh Monitoring Protocol (BROMP) 2.0 Technical and Training Manual. Technical Report. New York, NY: Teachers College, Columbia University. Manila, Philippines: Ateneo Laboratory for the Learning Sciences.

Ocumpaugh, J., Baker, R., Gowda, S., Heffernan, N., & Heffernan, C. (2014). Population validity for Educational Data Mining models: A case study in affect detection. *British Journal of Educational Technology (BJET)*, 45(3), 487-501.

Odom, W., Zimmerman, J., Davidoff, S., Forlizzi, J., Dey, A. K., & Lee, M. K. (2012). A fieldwork of the future with user enactments. In *Proceedings of the Designing Interactive Systems Conference (DIS 2012)* (pp. 338-347). ACM.

Ogan, A., Walker, E., Baker, R. S., Rebolledo Mendez, G., Jimenez Castro, M., Laurentino, T., & De Carvalho, A. (2012). Collaboration in cognitive tutor use in Latin America: Field study and design recommendations. In *2012 CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2012)* (pp. 1381-1390). ACM.

Ogan, A., Walker, E., Baker, R., Rodrigo, M. M. T., Soriano, J. C., & Castro, M. J. (2015). Towards understanding how to assess help-seeking behavior across cultures. *International Journal of Artificial Intelligence in Education (IJAIED)*, 25(2), 229-248.

Olsen, J. (2017). Orchestrating Combined Collaborative and Individual Learning in the Classroom. *Unpublished doctoral dissertation.* Human-Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA.

Olsen, J. K., Belenky, D. M., Aleven, V., Rummel, N., Sewall, J., & Ringenberg, M. (2014). Authoring tools for collaborative intelligent tutoring system environments. In *Proceedings of the 2014 International Conference on Intelligent Tutoring Systems (ITS 2014)* (pp. 523-528). Springer, Cham.

Olsen, J., Rummel, N., & Aleven, V. (2018). Co-Designing Orchestration Support for Social Plane Transitions with Teachers: Balancing Automation and Teacher Autonomy. *International Society of the Learning Sciences (ISLS)*.

O'Shea, T. (1982) A self-improving quadratic tutor. In *Intelligent Tutoring Systems* (eds. D. H. Sleeman & J. S. Brown). Academic Press, London, pp. 283–307.

Oulasvirta, A., Kurvinen, E., & Kankainen, T. (2003). Understanding contexts by being there: Case studies in bodystorming. *Personal and Ubiquitous Computing*, 7(2), 125-134.

Pane, J. F., Griffin, B. A., McCaffrey, D. F., & Karam, R. (2013). Effectiveness of Cognitive Tutor Algebra I at scale. *Educational Evaluation and Policy Analysis*, 2013.

Paquette, L., Baker, R. S., de Carvalho, A., & Ocumpaugh, J. (2015). Cross-system transfer of machine learned and knowledge engineered models of gaming the system. In *Proceedings of the 2015 International Conference on User Modeling, Adaptation, and Personalization (UMAP 2015),* (pp. 183-194). Springer, Cham.

Paquette, L., Baker, R. S., & Moskal, M. (2018). A system-general model for the detection of gaming the system behavior in CTAT and LearnSphere. In *International Conference on Artificial Intelligence in Education (AIED 2018),* (pp. 257-260). Springer, Cham.

Payne, A. F., Storbacka, K., & Frow, P. (2008). Managing the co-creation of value. *Journal of the Academy of Marketing Science*, 36(1), 83-96.

Pavlik, P. I., Cen, H., & Koedinger, K. R. (2009). Performance Factors Analysis – A New Alternative to Knowledge Tracing. In *Proceedings of the 2009 Conference on Artificial Intelligence in Education (AIED 2009)*, (pp. 531-538). IOS Press.

Pelánek, R., & Řihák, J. (2017). Experimental analysis of mastery learning criteria. In *Proceedings of the 2017 Conference on User Modeling, Adaptation and Personalization (UMAP),* (pp. 156-163). ACM.

Prahalad, C. K., & Ramaswamy, V. (2004). Co-creation experiences: The next practice in value creation. *Journal of Interactive Marketing*, 18(3), 5-14.

Prieto-Alvarez, C. G., Martinez-Maldonado, R., & Anderson, T. D. (2018). Co-designing learning analytics tools with learners. *Learning Analytics in the Classroom: Translating Learning Analytics for Teachers.*

Prieto, L. P. (2012). Supporting orchestration of blended CSCL scenarios in Distributed Learning Environments. Unpublished doctoral thesis.

Prieto, L. P., Dlab, M. H., Gutiérrez, I., Abdulwahed, M., & Balid, W. (2011). Orchestrating technology enhanced learning: a literature review and a conceptual framework. *International Journal of Technology Enhanced Learning (IJTEL)*, 3(6), 583.

Prieto, L. P., Magnuson, P., Dillenbourg, P., & Saar, M. (2017). Reflection for action: Designing tools to support teacher reflection on everyday evidence. Preprint available at https://osf. io/bj2rp.

Prieto, L. P., Sharma, K., Dillenbourg, P., & Rodriguez-Triana, M. J. (2016). Teaching analytics: Towards automatic extraction of orchestration graphs using wearable sensors. In *Proceedings of the 2016 International Conference on Learning Analytics and Knowledge (LAK 2016)*, (pp. 148-157). ACM.

Quintana, R., Quintana, C., Madeira, C., & Slotta, J. D. (2016). Keeping Watch: Exploring Wearable Technology Designs for K-12 Teachers. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI 2016)* (pp. 2272-2278). ACM.

Raca, M., & Dillenbourg, P. (2013). System for assessing classroom attention. In *Proceedings of the 2013 International Conference on Learning Analytics and Knowledge (LAK 2013)*, (pp. 265-269(. ACM.

Razzaq, L., Patvarczki, J., Almeida, S. F., Vartak, M., Feng, M., Heffernan, N. T., & Koedinger, K. R. (2009). The Assistment Builder: Supporting the life cycle of tutoring system content creation. *IEEE Transactions on Learning Technologies (TLT),* 2(2), 157-166.

Ritter, S., Anderson, J. R., Koedinger, K. R., & Corbett, A. (2007). Cognitive Tutor: Applied research in mathematics education. *Psychonomic Bulletin & Review*, 14(2), 249-255.

Ritter, S., Carlson, R., Sandbothe, M., & Fancsali, S. E. (2015). Carnegie learning's adaptive learning products. In *Proceedings of the 2015 International Conference on Educational Data Mining (EDM 2015)*. IEDMS.

Ritter, S., & Koedinger, K. R. (1995). Towards lightweight tutoring agents. In *Proceedings of the 1995 International Conference on Artificial Intelligence in Education (AIED 1995)*, (pp. 16-19).

Ritter, S., Yudelson, M., Fancsali, S. E., & Berman, S. R. (2016a). How mastery learning works at scale. In *Proceedings of the Third (2016) ACM Conference on Learning@ Scale* (pp. 71-79). ACM.

Ritter, S., Yudelson, M., Fancsali, S., & Berman, S. R. (2016b). Towards Integrating Human and Automated Tutoring Systems. In *Proceedings of the 2016 Conference on Educational Data Mining (EDM 2016),* (pp. 626-627).

Robertson, A. (2019). Nreal's AR sunglasses cost $499 and should ship in 'limited quantities' this year. *Verge*. Retrieved on July 11 from https://www.theverge.com/2019/5/30/18646160/nreal-light-ar-xr-qualcomm-snapdragon-sunglasses-consumer-price-shipping-release-5g.

Robins, J. M., Scheines, R., Spirtes, P., & Wasserman, L. (2003). Uniform consistency in causal inference. *Biometrika*, 90(3), 491-515.

Rodríguez-Triana, M. J., Prieto, L. P., Martínez-Monés, A., Asensio-Pérez, J. I., & Dimitriadis, Y. (2018). The teacher in the loop: Customizing multimodal learning analytics for blended learning. In *Proceedings of*

the 2018 International Conference on Learning Analytics and Knowledge (LAK 2018) (pp. 417-426). ACM.

Rodríguez-Triana, M. J., Prieto, L. P., Vozniuk, A., Boroujeni, M. S., Schwendimann, B. A., Holzer, A., & Gillet, D. (2017). Monitoring, awareness and reflection in blended technology enhanced learning: a systematic review. *International Journal of Technology Enhanced Learning (IJTEL)*, 9(2-3), 126-150.

Roll, I., Aleven, V., McLaren, B. M., & Koedinger, K. R. (2011). Improving students' help-seeking skills using metacognitive feedback in an intelligent tutoring system. *Learning and instruction*, 21(2), 267-280.

Roll, I., Baker, R. S. D., Aleven, V., & Koedinger, K. R. (2014). On the benefits of seeking (and avoiding) help in online problem-solving environments. *Journal of the Learning Sciences (JLS)*, 23(4), 537-560.

Rosé, C. P., McLaughlin, E. A., Liu, R. and Koedinger, K. R. (2019). Explanatory learner models: Why machine learning (alone) is not the answer. *British Journal of Educational Technology (BJET)*. BERA. doi:10.1111/bjet.12858

Rummel, N. (2018). One framework to rule them all? Carrying forward the conversation started by Wise and Schwarz. *International Journal of Computer-Supported Collaborative Learning (CSCL)*, 13(1), 123-129.

Rummel, N., Walker, E., & Aleven, V. (2016). Different futures of adaptive collaborative learning support. *International Journal of Artificial Intelligence in Education (IJAIED)*, 26(2), 784-795.

San Pedro, M. O. C. Z., Baker, R. S., & Rodrigo, M. M. T. (2011). Detecting carelessness through contextual estimation of slip probabilities among students using an intelligent tutor for mathematics. In *Proceedings of the 2011 International Conference on Artificial Intelligence in Education (AIED 2011)*, 304-311. Springer Berlin Heidelberg

Segedy, J., Sulcer, B., & Biswas, G. (2010). Are ILEs ready for the classroom? Bringing teachers into the feedback loop. In *Proceedings of the 2010 International Conference on Intelligent Tutoring Systems (ITS 2010),* (pp. 405-407). Springer, Berlin, Heidelberg.

Schoenfeld, A. H. (2008). On modeling teachers' in-the-moment decision-making. *A study of teaching: Multiple lenses, multiple views*, 45-96.

Schoenfeld, A. H. (2010). How we think: A theory of goal-oriented decision making and its educational applications. Routledge.

Schofield, J. W., Eurich-Fulcer, R., & Britt, C. L. (1994). Teachers, computer tutors, and teaching: The artificially intelligent tutor as an agent for classroom change. *American Educational Research Journal (AERJ)*, 31(3), 579-607.

Schofield, J. W. (1997). Psychology: Computers and classroom social processes—a review of the literature. *Social Science Computer Review*, 15(1), 27-39.

Sears, A., Lin, M., Jacko, J., & Xiao, Y. (2003). When computers fade: Pervasive computing and situationally-induced impairments and disabilities. In *HCI international* (Vol. 2, No. 03, pp. 1298-1302).

Sharples, M. (2013). Shared orchestration within and beyond the classroom. *Computers & Education*, 69, 504-506.

Sherin, M., Jacobs, V., & Philipp, R. (Eds.). (2011). Mathematics teacher noticing: Seeing through teachers' eyes. Routledge.

Sottilare, R. A., Baker, R. S., Graesser, A. C., & Lester, J. C. (2018). Special issue on the Generalized Intelligent Framework for Tutoring (GIFT): creating a stable and flexible platform for innovations in AIED research. *International Journal of Artificial Intelligence in Education (IJAIED)*, 28(2), 139-151.

Spirtes, P., Glymour, C. N., & Scheines, R. (2000). Causation, prediction, and search. MIT press.

Stamper, J., Koedinger, K., Pavlik Jr, P. I., Rose, C., Liu, R., Eagle, M., & Veeramachaneni, K. (2016). Educational data analysis using LearnSphere. In *Workshop and Tutorial Proceedings of the 2016 International Conference on Educational Data Mining (EDM 2016)*. IEDMS.

Stang, J. B., & Roll, I. (2014). Interactions between teaching assistants and students boost engagement in physics labs. *Physical Review Special Topics–Physics Education Research,* 10(2), 020117.

Steenbergen-Hu, S., & Cooper, H. (2013). A meta-analysis of the effectiveness of intelligent tutoring systems on K–12 students' mathematical learning. *Journal of Educational Psychology*, 105(4), 970.

Steenbergen-Hu, S., & Cooper, H. (2014). A meta-analysis of the effectiveness of intelligent tutoring systems on college students' academic learning. *Journal of Educational Psychology*, 106(2), 331.

Tissenbaum, M., Matuk, C., Berland, M., Lyons, L., Cocco, F., Linn, M., Plass, J. L., Hajny, N., Olsen, A., Schwendimann, B., & Boroujeni, M. S. (2016). Real-time visualization of student activities to support classroom orchestration. Symposium at the *2016 International Conference of the Learning Sciences (ICLS 2016)*. Singapore: International Society of the Learning Sciences.

Tohidi, M., Buxton, W., Baecker, R., & Sellen, A. (2006). User sketches: A quick, inexpensive, and effective way to elicit more reflective user feedback. In *Proceedings of the 4th Nordic conference on Human-computer interaction (NordiCHI 2006)* (pp. 105-114). ACM.

Toombs, A., Devendorf, L., Shih, P., Kaziunas, E., Nemer, D., Mentis, H., & Forlano, L. (2018). Sociotechnical Systems of Care. In *Companion of the 2018 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW 2018)* (pp. 479-485). ACM.

Toombs, A. L., Dow, A., Vines, J., Gray, C. M., Dennis, B., Clarke, R., & Light, A. (2018). Designing for Everyday Care in Communities. In *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2018)* (pp. 391-394). ACM.

Toyama, K. (2018). From needs to aspirations in information technology for development. *Information Technology for Development*, 24(1), 15-36.

Trischler, J., Pervan, S. J., Kelly, S. J., & Scott, D. R. (2018). The value of codesign: The effect of customer involvement in service design teams. *Journal of Service Research*, 21(1), 75-100.

Tukey, J.W. (1977). Exploratory Data Analysis. Addison-Wesley.

Vanlehn, K. (2006). The behavior of tutoring systems. *International Journal of Artificial Intelligence in Educatio*n (IJAIED), 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.

VanLehn, K. (2016). Regulative loops, step loops and task loops. *International Journal of Artificial Intelligence in Education (IJAIED)*, 26(1), 107-112.

VanLehn, K., Burkhardt, H., Cheema, S., Kang, S., Pead, D., Schoenfeld, A., & Wetzel, J. (2019). Can an orchestration system increase collaborative, productive struggle in teaching-by-eliciting classrooms?. *Interactive Learning Environments*, 1-19.

Vatrapu, R. K., Kocherla, K., & Pantazos, K. (2013, April). iKlassroom: Real-Time, Real-Place Teaching Analytics. In *Proceedings of the International Workshop on Teaching Analytics (IWTA) at the 2013 International Conference on Learning Analytics and Knolwedge (LAK 2013).*

Veale, M., Van Kleek, M., & Binns, R. (2018). Fairness and accountability design needs for algorithmic support in high-stakes public sector decision-making. In *2018 CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2018),* (p. 440). ACM.

Veitch, J., Salmon, J., & Ball, K. (2007). Children's active free play in local neighborhoods: A behavioral mapping study. *Health Education Research*, 23(5), 870-879.

Verbert, K., Duval, E., Klerkx, J., Govaerts, S., & Santos, J. L. (2013). Learning analytics dashboard applications. *American Behavioral Scientist*, 57(10), 1500-1509.

Waalkens, M., Aleven, V., & Taatgen, N. (2013). Does supporting multiple student strategies lead to greater learning and motivation? Investigating a source of complexity in the architecture of intelligent tutoring systems. *Computers & Education*, 60(1), 159-171.

Walker, E., Rummel, N., & Koedinger, K. R. (2014). Adaptive intelligent support to improve peer tutoring in algebra. *International Journal of Artificial Intelligence in Education (IJAIED)*, 24(1), 33-61.

Walsh, G., Foss, E., Yip, J., & Druin, A. (2013). FACIT PD: a framework for analysis and creation of intergenerational techniques for participatory design. In *Proceedings of the 2013 CHI Conference on Human Factors in Computing Systems (CHI 2013)*, (pp. 2893-2902). ACM.

Wang, X., Talluri, S. T., Rosé, C., & Koedinger, K. (2019). UpGrade: Sourcing student open-ended solutions to create scalable learning opportunities.

Wang, D., Yang, Q., Abdul, A., & Lim, B. Y. (2019). Designing Theory-Driven User-Centric Explainable AI. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI 2019)* (p. 601). ACM.

Watters, A. (2014). The monsters of education technology. CreateSpace.

Wetzel, J., Burkhardt, H., Cheema, S., Kang, S., Pead, D., Schoenfeld, A., & VanLehn, K. (2018). A preliminary evaluation of the usability of an AI-infused orchestration system. In *International Conference on Artificial Intelligence in Education (AIED 2018),* (pp. 379-383). Springer, Cham.

Williams, J. J., Kim, J., Rafferty, A., Maldonado, S., Gajos, K. Z., Lasecki, W. S., & Heffernan, N. (2016). AXIS: Generating explanations at scale with learnersourcing and machine learning. In *Proceedings of the 2016 ACM Conference on Learning@ Scale (L@S 2016)* (pp. 379-388). ACM.

Williamson, B. (2016). Calculating children in the dataveillance school: Personal and learning analytics. In *Surveillance Futures* (pp. 62-90). Routledge.

Williamson, B. (2017). Decoding ClassDojo: psycho-policy, social-emotional learning and persuasive educational technologies. *Learning, Media and Technology*, 42(4), 440-453.

Wobbrock, J. O., Kane, S. K., Gajos, K. Z., Harada, S., & Froehlich, J. (2011). Ability-based design: Concept, principles and examples. *ACM Transactions on Accessible Computing (TACCESS)*, 3(3), 9.

Wobbrock, J. O., & Kientz, J. A. (2016). Research contributions in human-computer interaction. *Interactions*, 23(3), 38-44.

Wong, R. Y., & Mulligan, D. K. (2019). Bringing Design to the Privacy Table: Broadening. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI 2019)*, (p. 262). ACM.

Woolf, B., Burleson, W., Arroyo, I., Dragon, T., Cooper, D., & Picard, R. (2009). Affect-aware tutors: recognising and responding to student affect. *International Journal of Learning Technology*, 4(3-4), 129-164.

Xhakaj, F., Aleven, V., & McLaren, B. M. (2016). How teachers use data to help students learn: Contextual inquiry for the design of a dashboard. In *European Conference on Technology Enhanced Learning (EC-TEL 2016)* (pp. 340-354). Springer, Cham.

Xhakaj, F., Aleven, V., & McLaren, B. M. (2017). Effects of a teacher dashboard for an intelligent tutoring system on teacher knowledge, lesson planning, lessons and student learning. In *European Conference on Technology Enhanced Learning (EC-TEL 2017)* (pp. 315-329). Springer, Cham.

Xu, Z., Wijekumar, K., Ramirez, G., Hu, X., & Irey, R. (2019). The effectiveness of intelligent tutoring systems on K−12 students' reading comprehension: A meta−analysis. *British Journal of Educational Technology (BJET).*

Yacef, K. (2002). Intelligent teaching assistant systems. In *Proceedings of the 2002 International Conference on Computers in Education (ICCE 2002),* (pp. 136-140). IEEE.

Yang, Q., Scuito, A., Zimmerman, J., Forlizzi, J., & Steinfeld, A. (2018). Investigating how experienced UX designers effectively work with machine learning. In *Proceedings of the 2018 Designing Interactive Systems Conference (DIS 2018),* (pp. 585-596). ACM.

Yang, Q., Steinfeld, A., & Zimmerman, J. (2019). Unremarkable AI: Fitting Intelligent Decision Support into Critical, Clinical Decision-Making Processes. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (p. 238). ACM.

Yang, Q., Zimmerman, J., Steinfeld, A., Carey, L., & Antaki, J. F. (2016). Investigating the heart pump implant decision process: Opportunities for decision support tools to help. In *2016 CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2016)* (pp. 4477-4488). ACM.

Yudelson, M. V., Koedinger, K. R., & Gordon, G. J. (2013). Individualized bayesian knowledge tracing models. In *International Conference on Artificial Intelligence in Education (AIED 2013)* (pp. 171-180). Springer, Berlin, Heidelberg.

Zarraonandia, T., Aedo, I., Díaz, P., & Montero, A. (2013). An augmented lecture feedback system to support learner and teacher communication. *British Journal of Educational Technology (BJET)*, 44(4), 616-628.

Zhang, C., Huang, Y., Wang, J., Fang, W., Lu, D., Stamper, J., Fancsali, S., Holstein, K., & Aleven, V. (2019). Early detection of wheel spinning: Comparison across tutors models, features, and operationalizations. In *Proceedings of the Twelfth International Conference on Educational Data Mining. (EDM'19)*. IEDMS.

Zhu, H., Yu, B., Halfaker, A., & Terveen, L. (2018). Value-Sensitive Algorithm Design: Method, Case Study, and Lessons. *Proceedings of the ACM on Human-Computer Interaction, 2(CSCW)*, 194.

Zimmerman, B. J. (2008). Investigating self-regulation and motivation: Historical background, methodological developments, and future prospects. *American Educational Research Journal (AERJ),* 45(1), 166-183.

Zimmerman, J., & Forlizzi, J. (2014). Research through design in HCI. In *Ways of Knowing in HCI* (pp. 167-189). Springer, New York, NY.

Zimmerman, J., & Forlizzi, J. (2019). Service design. In *The Encyclopedia of Human-Computer Interaction 2nd Edition* (53). Interaction Design Foundation.

Zimmerman, J., & Forlizzi, J. (2017). Speed Dating: Providing a Menu of Possible Futures. *She Ji: The Journal of Design, Economics, and Innovation*, 3(1), 30-50.