Developing Affect-Aware Robot Tutors

by

Samuel Lee Spaulding

B.S., Yale University (2013)

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
in partial fulfillment of the requirements for the degree of

Master of Science in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2015

Author_____

Samuel Lee Spaulding
Program in Media Arts and Sciences
May 12, 2015

Certified by_____

Cynthia Breazeal
Associate Professor of Media Arts and Sciences
Program in Media Arts and Sciences
Thesis Supervisor

Accepted by_____

Pattie Maes
Professor of Media Arts and Sciences
Academic Head, Program in Media Arts and Sciences

Developing Affect-Aware Robot Tutors

by

Samuel Lee Spaulding

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
on May 12, 2015, in partial fulfillment of the
requirements for the degree of
Master of Science in Media Arts and Sciences

Abstract

In recent years there has been a renewed enthusiasm for the power of computer systems and digital technology to reinvent education. One-on-one tutoring is a highly effective method for increasing student learning, but the supply of students vastly outpaces the number of available teachers. Computational tutoring systems, such as educational software or interactive robots, could help bridge this gap. One problem faced by all tutors, human or computer, is assessing a student's knowledge: how do you determine what another person knows or doesn't know? Previous algorithmic solutions to this problem include the popular Bayesian Knowledge Tracing algorithm and other inferential methods. However, these methods do not draw on the affective signals that good human teachers use to assess knowledge, such as indications of discomfort, engagement, or frustration. This thesis aims to make understanding affect a central component of a knowledge assessment system, validated on a dataset collected from interactions between children and a robot learning companion. In this thesis I show that (1) children emote more when engaging in an educational task with an embodied social robot, compared to a tablet and (2) these emotional signals improve the quality of knowledge inference made by the system. Together this work establishes both human-centered and algorithmic motivations for further development of robotic systems that tightly integrate affect understanding and complex models of inference with interactive, educational robots.

Thesis Supervisor: Cynthia Breazeal
Title: Associate Professor of Media Arts and Sciences, Program in Media Arts and Sciences

Developing Affect-Aware Robot Tutors

by

Samuel Lee Spaulding

The following people served as readers for this thesis:

Thesis Reader_____

Rosalind Picard
Professor of Media Arts and Sciences
Program in Media Arts and Sciences

Thesis Reader_____

Sepandar Kamvar
Associate Professor of Media Arts and Sciences
Program in Media Arts and Sciences

Table Of Contents

Acknowledgements

There are many, many people to whom I owe thanks for their support. I would like to thank my advisor, Cynthia Breazeal, for her incredible vision of a world of social robots and for giving me the opportunities, resources, and guidance to build towards that vision. My readers, Roz Picard and Sep Kamvar, also deserve thanks and recognition for their help. Roz, your work on emotional machines underlies and makes possible the ideas in this thesis. Practically, your many helpful comments regarding data interpretation and model validation also made this thesis much stronger. Sep, your deep understanding of how the design of technology reveals its values has been a guiding principle for my work. Your advice and encouragement were very useful in shaping these ideas, especially in the crucial early stages.

It has been a tremendous privilege to work with the Personal Robots Group at MIT. Robotics is truly a team sport, and it has been incredible to be a part of such a talented, hard-working, fun, and friendly lab. Jin Joo, you've been a great friend and resource since Day 1 and have helped me out of way more problems than I can count. Nick, thanks for always encouraging me to dream big, question ideology, and pursue real problems. Goren, you deserve an extra shoutout because this thesis work all came out of your dataset! But you've also been a huge help on the computational side and your infectious enthusiasm for the scientific future is an inspiring and positive force. Michal, it has been great learning how to work Android and dig through the old AIDA code videos with you. Jackie, thanks for being there with a relevant reference or an interesting question to spark conversation about brains and minds. Palash, you've made me a way better software engineer and your knowledge of how everything fits, or should fit, together is astounding. David, thanks for exposing me the world of performance and robots! Sooyeon, your unflappable demeanor, even when things are going wrong, is a great example for how I wish I could approach every challenge. Luke, your engineering skills and attitude are incredibly inspiring - seeing you at work makes me believe we really can make (almost) anything here at the lab. Brad, you pushed me to be a more rigorous researcher and I'll always be grateful for your mentorship during my first year. And of course nothing would ever get done without Polly, the beating

heart of the Personal Robots Group, who somehow manages to keep the lab from collapsing under its own ambitious weight through the power of hugs, punches, and the occasional lunch! Big thanks are also owed to all PRG alums, especially Matt, Jesse, Siggi, and Kris: your work forms the foundations of practically everything we do here.

To all my colleagues at the Yale Social Robotics Laboratory, especially Scaz and Dan, I will forever be grateful that you let a freshman with only a semester of Java programming experience come into your lab and work with your robots. Getting to stay in the SAR Expedition family has been a great opportunity, and I can't imagine a better team to work with on these exciting challenges.

Finally, I need to give special thanks to my girlfriend Caroline, you have made me a better person in nearly every way and meeting you five years ago was one of the best things to ever happen to me. And of course, the most important people of all: my family. To my parents and siblings, Mom, Dad, Matt, Becca, Sarah, I hope I've made you proud. I owe you everything.

This research was supported by the National Science Foundation (NSF) under Grants CCF-1138986 and an NSF GRF under Grant No. 1122374. Any opinions, findings and conclusions, or recommendations expressed in this paper are those of the authors and do not represent the views of the NSF.

1 Research Problem & Motivation

In 1984, educational researcher Benjamin Bloom presented his now-famous "two-sigma" result: students that received one-on-one tutoring from a teacher performed nearly two standard deviations higher than the mean student who received only group instruction [Bloom (1984)]. Given the vast gap between the supply of available teachers and needy students, Bloom challenged educational researchers to "find methods of group instruction as effective as one-to-one tutoring."

The search for such methods may turn out to be unnecessary. The field of Intelligent Tutoring Systems (ITS) seeks to develop artificially intelligent software that can provide the benefits of personal instruction at scale. Already, thousands of students nationwide benefit from the fruits of ITS research, through commercial software such as the Carnegie Math Tutor.

Yet many of these systems fall short of Bloom's initial estimate of the benefit of one-on-one tutoring. Though the initial "two-sigma" benefit of one-on-one tutoring may have been overstated, a recent meta-analysis of ITS research suggests that ITSs provide approximately a ".76 sigma" increase [VanLehn (2011)]. In addition to learning gains, the subjective experience of interacting with an ITS still leaves much to be desired. ITSs are often designed as digital workbooks, with the central activity focused on the tutoring system providing practice problem after practice problem, and offering the student a limited action space (often just the choice of providing an answer or asking for a hint).

Recent research on school-age children suggests that interactive learning styles such as "learning through play" and "peer learning" are highly effective, for example [Slavin (2011)], [Crouch and Mazur (2001)], and [Keppell et al. (2006)]. Therefore, we believe these principles should be incorporated into the design of our computational tutors. The ICAP framework differentiates between four types of educational activity, emphasizing computational systems that support *Interactive* and *Constructive* activities over merely *Active*[1] or *Passive* learning opportunities [Chi (2009)]. This framework increasingly forms the basis of new research in the ITS community.

Even those ITSs that *do* attempt to engage students in interactive dialogue face limitations compared to human tutors. Typically, they are limited to sensing the student only by her[2] actions in the program, and their actions are likewise limited to on-screen events. In contrast, the interaction between a human tutor and his student draws on rich, multimodal data and permits a wide range of curricular and meta-curricular actions that take place in the real world. Program performance is merely one feature among many that a human tutor uses to assess a student's knowledge state, and students have a vastly expanded space of actions beyond what is typically afforded by ITSs.

By centering the interaction around collaborative, interactive problem-solving and using affective information to construct models of students' learning, the next generation of computational tutoring systems may provide benefits much closer to those provided by human tutors. In this thesis, I argue that *physically embodied social robots, capable of perceiving and understanding affective signals* are a more appropriate platform for accomplishing this task than software-only tutors. I support this argument with results

[1] An example of an educational activity that is *active* but not *constructive* or *interactive* is answering a question in a workbook-style ITS

[2] In order to avoid cumbersome constructions such as 'he or she' I adopt the following convention: in odd numbered chapters, I use the masculine 'he' for the tutor and 'she' for the student. In even numbered chapters the roles are reversed. Thus in Chapter 2, 'she' refers to the tutor and 'he' to the student

addressing two main research questions:

1. Are children more emotionally expressive when engaged in an educational interaction with a social robot, compared to a tablet?

2. Can these emotional expression signals improve the performance of state-of-the-art student models?

To address these questions, I conducted a post-hoc analysis of emotional expression on video recordings of the experiment described in [Gordon and Breazeal (2015)] and [Gordon et al. (2015)]. In this experiment, children played a short, interactive story-telling game with either a robot or a tablet. Periodically, the robot/tablet would assess the child's reading ability by verbally asking her to point to a particular word written on the screen. Chapter 3 describes the results of the analysis of children's facial expressions during this educational interaction and shows that children who interacted with a robot produced more emotional expressions. Chapter 4 describes work showing that by using this emotional expression data as feature input to train a knowledge inference model, the model is better able to generalize to test data from the same population.

Summary: Preliminary research suggests that physically embodied robot tutors can deliver more engaging and empathic educational experiences. In this thesis, I seek to add to this increasing body of evidence by establishing social robots' unique capability to engage children in *affect-aware tutoring experiences*. I show that children are more emotionally expressive when interacting with a social robot and that an autonomous tutoring agent can leverage these emotional signals to construct richer models of child learning that can better infer children's hidden mental states.

2 Related Work

In this section I give an overview of recent Intelligent Tutoring Systems research, with particular emphasis on the current state-of-the-art in affect-aware tutors. I then discuss recent results from the field of Human-Robot Interaction that illustrate how the physical embodiment of a robot can facilitate deeper engagement and foster better learning. Finally, I examine previous attempts to unify affect-aware algorithms with physical robots and discuss how this thesis work represents a distinct contribution to the previous body of research.

2.1 Affect-aware Tutoring Software

Intelligent Tutoring Systems (ITSs) refer to a wide variety of computer-based educational tools. Common features of an ITS include the ability to change its behavior in response to student input, provide help in the form of a hint or additional instruction, and conduct some form of evaluation of the user. VanLehn[VanLehn (2011)] distinguishes between two broad classes of computer-based tutors: workbook style systems that provide

hints or feedback on students' answers, which he refers to as 'Computer-Aided Instruction' (CAI) systems and more freeform software, characterized by interactivity, open-response answers, and feedback on students' *process* towards a solution, rather than just the solution itself. VanLehn refers to these types of software tutors as 'Intelligent Tutoring Systems'.

As previously discussed, ITSs are already in use outside of the lab, in schools or daycares. But, as is the case in many of the applied sciences, the deployed systems typically lag behind the cutting edge of research. Thus, while commercial tutoring systems rarely (if at all) consider students' affective or emotional states, the research community has begun to address these problems. The subfield of "affect-aware tutors" [Woolf et al. (2009)] seeks to design more effective artificially intelligent tutoring systems that explicitly sense, model, and reason about students' affective states. Inspired by psychological theories of emotion and learning, affect-aware tutors seek to foster engagement and learning from data-driven estimates of students' affective states. For example, the Wayang geometry tutor[Arroyo et al. (2004)] is a system that features a virtual agent which helps students solve geometry problems. In order to foster engagement, the tutor uses an empathy-based affective behavior system: the emotional actions of the tutor are intended to mirror the (estimated) emotional state of the user. For example, if a child appears bored, the tutor might also display signs of boredom before suggesting a new topic or problem to keep the student engaged.

Recently, several major shifts in the research landscape have enabled more capable and effective affect-aware tutors. First, affective sensing technology has improved greatly over the past decade. The modern concept of Affective Computing [Picard (2000)] was born less than 20 years ago and has matured into a highly active research topic with impact in many other fields. Major advances in unintrusive affective sensing technology, such as algorithms for facial expression analysis [El Kaliouby and Robinson (2005)], wristbands that sense electrodermal activity, and pressure sensitive chairs and computer peripherals [Reynolds and Picard (2004)] have enabled the collection of large quantities of real-time affective data without significantly disrupting the interaction.

Second, the widespread use of machine learning methods in Artificial Intelligence and increase in "Big Data" technologies have given researchers the tools to manage and analyze this affective data. As a result, researchers working in these areas are better equipped to understand how to interpret and act on the collected data.

Finally, enthusiasm for computer education technology has skyrocketed in the last decade. This enthusiasm, largely precipitated by the rise of Massive Open Online Courses (MOOCs), has led to significant development of online educational technology outside of research labs, for instance, by venture-backed start-ups (e.g., Coursera, Udacity) or dedicated branches of academic institutions (e.g., EdX - a collaboration between Harvard and MIT). These trends combine to make the current research environment well-suited for a renewed effort towards affect-aware tutoring technology.

Recent efforts to develop affect-aware tutoring systems have culminated in a number of major systems (e.g., the Wayang Tutor[Arroyo et al. (2004)] and Affective MetaTutor[VanLehn et al. (2011)] projects) which have been extensively studied. Yet much of the work on affect and modeling in the ITS literature focuses on models to *infer* affect. Typically, once affective states are detected or identified, they trigger simple behavioral rules - a tutor might change its facial expression or offer a supportive comment. However,

these rules are hardcoded by the developers and remain fixed throughout the deployment.

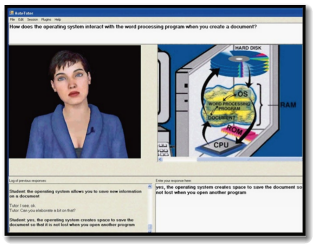
One of the first projects to combine affect and tutoring was the Affective Learning Companion[Burleson (2006)], developed by Winslow Burleson and Roz Picard at the MIT Media Lab. The Affective Learning Companion used sensors including a pressure sensitive mouse, cameras to analyze facial expression, and a posture-sensitive chair to infer a student's affective state. Two of the most important states considered were 'Flow', a theorized state of enjoyable focus on an optimally challenging task (described by Mihalyi Csikszentmihalyi[Csikszentmihalyi and Csikszentmihalyi (1992)]), and 'Stuck', an opposing affective state characterized by frustration towards an overwhelming challenge, while completing a complex puzzle task. This project has since spawned several successors, and many of the features of the Affective Learning Companion (e.g., the use of virtual agents as supportive tutors, the use of physiological sensors to infer affect, and the use of affective signals to influence the tutor's behavior in real time) are now core aspects of nearly all other affect-aware tutoring systems.

In the Affective Meta-Tutor project, affect is primarily used as a tool to help students maintain motivation. Tutoring occurs at several levels: at a basic level, students use a graphical interface to define causal networks in order to develop 'systems thinking'. The 'Meta-Tutor' gives students advice on how to best plan their solution strategy (e.g., by focusing on solving one part of the problem before considering the bigger picture). The affective component of the Meta-Tutor uses information from students' facial expressions and posture to infer affect, then periodically provides supportive statements (e.g., "It seems like you are using the strategy and that all your efforts are helping you to make strong connections in your brain. Nice work!") to either congratulate or motivate students to follow the Meta-Tutor's advice. Evaluation of the affective component was mixed: students who had interacted with the affective Meta-Tutor showing learning gains (over students who interacted with a non-affective Meta-Tutor) during a training interaction, but these gains did not persist into the later 'transfer' phase of the experiment.

The Affective Autotutor[D'Mello et al. (2011)] is an extension of the more widely studied Autotutor ITS[Graesser et al. (2005)]. The Affective Autotutor also estimates students' affective states from posture and facial expression data which triggers behavioral rules

(a) An affect-sensitive learning companion [Burleson (2006)]

(b) The Wayang Mathematics Tutor uses diverse agent representations, capable of showing positive, neutral, and negative affect[Woolf et al. (2009)]



(c) AutoTutor provides motivational support based on real-time affect detection[D'Mello et al. (2011)]

Figure 1: Previous attempts to develop affect-aware tutoring systems

designed to engage and motivate students. Autotutor is designed around natural language dialogue and has more sophisticated models of conversation and dialogue than a typical ITS. In an experiment comparing the effectiveness of the Affective Autotutor to the ordinary Autotutor, researchers found that for low domain knowledge students, the Affective Autotutor system increased learning gains, but for students who began the interaction with a high level of domain knowledge, the Affective Autotutor *reduced* their learning. One possible explanation is that the additional affective dialogue was distracting or seemed unnecessary to these students, which may have caused them to ignore the tutor. This highlights the need for more sophisticated models of dialogue that *explicitly* consider affect. Understanding how frequently an ITS should provide feedback has long been an active research topic[Corbett and Anderson (2001)], and affective information may be one key to the solution.

The ability of ITSs to recognize and understand affect is growing. Several ITSs have been developed and studied that respond to affect, and results show that affect can indeed improve students learning, *under certain conditions*. These results also emphasize the importance of developing new techniques for interaction that explicitly consider affect. Currently, most affect-aware tutors use rule-based systems, informed by modern theories of emotion and learning, to act on affective information. While these systems currently represent the state-of-the-art, their limitations suggest that more flexible and general models of decision making may ultimately be necessary.

2.2 Physically Embodied Robots

Personal robots are rapidly moving into our daily lives. Over the last 15 years, researchers have dedicated serious effort towards the development of *social* robots: robots that can interact naturally with humans, using the same sorts of multi-modal social cues used in typical human-human interaction. Researchers have long recognized the potential of technology to inspire and motivate childhood education [Papert (1980)] and social robots are the latest heirs to that tradition[Mead et al. (2012)]. Social robots have been successfully deployed in schools to teach topics as varied as English vocabulary [Tanaka and Matsuzoe (2012)], chess [Leite et al. (2014)], and nutrition [Short et al. (2014)] to children. Other research has explored the use of social robots as tutors for mathematics [Brown and Howard (2014)], anatomy [Howley et al. (2014)], and general analytic skills [Leyzberg et al. (2014)]. Long-term studies of robots, while less common, have demonstrated that social robots can be an effective tool for improving young children's literacy skills[Kory and Breazeal (2014)].

Much work in the field of Human-Robot Interaction (HRI) has been dedicated to investigating the effects of a physical robot's presence, compared to screen-based representations, in assistive and educational contexts. I summarize a few key works and their results here:

(1) *Physical robots increase task compliance.* In a 2011 study, students were more likely to comply with a physical robot's request than a screen-based representation's request, even if the task was odd or uncomfortable [Bainbridge et al. (2011)].

(2) *Physical robots are more able to maintain effective long-term relationships.* In a six week study of robots in the home, participants with physical robot partners recorded caloric intake and exercise habits for twice as long as those who used the same software or a paper equivalent [Kidd and Breazeal (2008)].

(3) *Physical robots produce greater learning gains.* In a 2012 study, students played several rounds of a puzzle game and received lessons on techniques for solving the puzzle. Those who got lessons from a physical robot completed the final puzzles significantly faster and improved their solving time significantly more than those who received identical lessons from an on-screen video of the same robot [Leyzberg et al. (2012)].

These results provide the initial suggestion that physical robot tutors may be particularly well-suited to tutoring and educational applications. However, due to the difficulty of long-term deployment of robots, many findings in the HRI literature are results gathered from one-time interactions, in a laboratory setting. Typically the robots used in these studies are controlled either by a hidden human operator (in what is called "Wizard-of-Oz" or "WOz" control) or by simple scripted behavioral rules. Thus, there remain significant challenges to overcome before social robots are as widely used as software-only tutoring systems. Chief among these are developing techniques to enable robots to make intelligent decisions about *social contexts* and giving robots the ability to construct complex action models so that they remain engaging and helpful over repeated interactions.

2.3 Robots and Affect

Within HRI, some work has attempted to enable physical robots to respond to human affect in educational scenarios. As with ITSs, however, these responses are typically the result of scripted rules. For instance, in a recent study by Szafir and Mutlu [Szafir and Mutlu (2012)], students wore an EEG sensor while a robot told a story. In one condition, the robot responded to perceived student decreases in attention by producing emphatic, exaggerated gestures. Following the robot's story, students were quizzed on details from the story. Participants that heard the story from the affect-adaptive robot had better recall, compared to students who heard the story from a robot that did not respond to the EEG signals.

As part of the LIREC project, researchers from the EU have developed a robotic chess tutoring system that uses sophisticated affective models to give empathy-based support[Leite et al. (2014)]. The robot models a child's affective state by tracking in-game events associated with known emotional states (e.g. losing a game, or making a bad move is associated with low valence, while a very evenly-matched situation is associated with high-engagement) in combination with external physical sensors. The robot offers support by choosing a supportive strategy that mirrors the child's (estimated) affective state.

The work for this thesis is distinct from previous efforts to unite affect, tutoring, and robots, in that we are incorporating affect into a *learning* model. While other work has used affect as an input to behavioral rules, this work is the first to have a robotic tutor system use affective data to *learn* models of how students' affect and learning intersect.

Summary: Over the last several years, great progress has been made in the area of affect sensing and our understanding how humans learn and interact with expressive, social robots has become more sophisticated. These foundational steps have paved the way for the development of more sophisticated modeling and control algorithms, capable of replicating the abilities of good human teachers. Thus far, research efforts to develop affect-aware robot tutors within the HRI community have taken for granted the idea that

physical robots are a better platform for artificially intelligent tutors. In this thesis, I adapt the modeling techniques of Intelligent Tutoring Systems to take advantage of the unique interaction dynamics and sensing modalities that a physical robot provides. I show that emotional signals can be a strong signal of a student's level of understanding, and that students display these signals more prominently in an interaction with a physically embodied robot than in the same interaction with software alone. Together with other research on the impact of emotion and embodiment in tutoring interactions, these results constitute a strong endorsement for the use of physically embodied, emotionally responsive social robots as computational tutors.

3 Are Children More Expressive When Interacting With a Robot?

This chapter addresses the first of the two major research questions posed previously: "Are children more emotionally expressive during educational interactions with a social robot, compared to a tablet?" By answering this question we seek to isolate and identify aspects of how the interaction between children and educational technology changes, depending on that technology's form factor. In the subsequent chapter, I will show how these emotional signals can be used by an affect-aware tutor to build more accurate models of what parts of a curriculum students have or have not mastered, and discuss how sensing affective expression can help computational tutors learn *personalized* models of how emotion and learning intersect for students.

Here, I wish to determine whether children generate more emotional signals when interacting with a social robot, compared to a tablet. Anecdotally, some parents report allowing their children to play with tablets in order to keep them from being *too* expressive. Anyone with young children may find the anecdote, reported in [Bilton (2013)], familiar:

"I recently watched my sister perform an act of magic. We were sitting in a restaurant, trying to have a conversation, but her children, 4-year-old Willow and 7-year-old Luca, would not stop fighting. The arguments – over a fork, or who had more water in a glass – were unrelenting. Like a magician quieting a group of children by pulling a rabbit out of a hat, my sister reached into her purse and produced two shiny Apple iPads, handing one to each child. Suddenly, the two were quiet. Eerily so. They sat playing games and watching videos, and we continued with our conversation."

In contrast to children's affect when using iPads or other screen-based technologies, children are *very* socially expressive around robots, according to the latest research on Child-Robot Interaction (e.g., [Belpaeme et al. (2012)]). Perhaps most strikingly, social robots have been used as therapeutic tools for autistic children [Scassellati et al. (2012)]. In one experiment, autistic children that were typically withdrawn or socially unresponsive when interacting with a therapist became much more socially engaged and active in a triadic interaction with a therapist *and* a social robot [Kim et al. (2013)]. This phenomenon is currently the subject of much research; currently, no comprehensive theories of how or why have been accepted by the autism research community.

Even so, it seems plausible that, due to a social robot's resemblance to recognizable social entities (e.g., other humans, pets, television characters), children are predisposed to interact with a robot in a social way. This initial predisposition, combined with reinforcement from the technological capabilities of the robot itself – social robots are typically designed to respond to social interaction cues – may lead children to interact more expressively and socially with robots than they would with traditional on-screen technologies. In this thesis, I wish to empirically test this hypothesis in the context of an educational interaction with a robotic tutor, without making claims about the cause of this phenomenon. In the following sections, I will define what I mean by "emotional expression" and tools and techniques I used to measure it.

Most research studying human affect uses human coders to manually review video footage of interactions and label the video with a discrete set of pre-determined affective states. Sometimes the coders will have experience or training in emotion recognition, but

most commonly, the video coders are either the experimenters themselves, undergraduate research assistants, or online workers (e.g., workers on Amazon Mechanical Turk). Inter-coder reliability ratings can help verify the accuracy of labeling, but in this thesis I take an entirely different approach. I envision a world in which affect-aware robots will be widely deployed, robust enough to support long-term interaction, and intelligent enough to act completely autonomously. Human annotation poses an impediment to the first criterion, and is directly opposed to the last. As described in Section 2.1, methods for unobtrusively and *autonomously* sensing affect are rapidly improving. In line with this vision, I rely solely on autonomous sensing throughout this thesis. Given these constraints, the research question can now be better specified as "Are children more expressive, *as measured by an autonomous emotion detection system*, when interacting with a robot?"

3.1 Affdex: a tool for autonomous affect detection

In order to analyze childrens' emotional expression, I used the Affdex mobile SDK, a commercial tool marketed by Affectiva, Inc. to enable developers to develop affect-aware mobile applications. Affdex uses state-of-the-art face detection and analysis algorithms to extract estimates of four physical facial expression features **(Smile, BrowFurrow, BrowRaise, and LipDepress)** and two hidden affective features **(Valence and Engagement)** from video or images of faces. For each of these six metrics, the SDK produces a number in the range of [0, 100], with the exception of Valence, which ranges from [-100, 100].

The Affdex SDK can operate on four different types of data: real-time input from phone camera, single images, video files (played back as if they were real-time streaming data), and sets of timestamped images. The first method was rejected because this thesis describes a *post-hoc* analysis of emotion, and therefore real-time data was not available. The second method was rejected because temporal features (such as the dynamics of facial landmarks over time) are highly informative for classification of emotion. The famous "tennis face" example (in which human participants were found to be unable to reliably distinguish between the facial expressions of winners and losers in tennis matches) [Aviezer et al. (2012)] is a simple and forceful illustration that expression analysis of single images in isolation is a difficult problem, even for other humans – the most advanced processors of (human) facial expressions on the planet.

The third option (video file) has the disadvantage of producing results at a slower rate, determined by the SDK itself, whenever the algorithm has accumulated enough information to generate a result. Because the results cannot be definitively linked back to a particular moment of video, this method makes reintegration with the rest of the data difficult. The final option, which I chose for this analysis, is to analyze sequences of time-stamped still images, all extracted from the same video. Because the images are all derived from the same source, Affdex can apply the more advanced affect detection algorithms described above by using each image's associated timestamp to extract temporal features. In addition, results are received for every frame, and each frame result can be precisely linked back to the rest of the data through its timestamp.

Figure 2: Faces of professional tennis players at the moment of winning (top row) or losing (bottom row) a match. Human participants could not reliably distinguish the two cases by facial expression alone.

Figure 3: Picture of the interaction setup from which the dataset was collected. Children played a storytelling game with a robot, during which they were periodically prompted to read a word

3.2 Task and Dataset

The data used in this analysis come from video records of a previous experiment conducted in the Personal Robots Group [Gordon and Breazeal (2015), Gordon et al. (2015)]. In this experiment, children (age 4-8) played a story game with a robot. The interaction was designed around a child-robot-tablet interface, in which the child and robot sit across from each other, with a tablet placed in between them to act as a shared, social context that can be sensed by the robot (Figure 3).

The child and the robot played an app-based game called "Storymaker" which was developed in-house. In the game, graphics of characters (such as animals or objects) float on a background of different scenes (such as a beach or jungle) and can be moved by the child via physical touches and swipes. As the child moves characters to different regions of the scene, the game procedurally generates a natural language sentence that characterizes this action[3]. For example, if the child moves the Dragon character graphic towards the rightmost side of the forest scene (depicting a tree), the game might generate the sentence "Dragon goes to the tree". This sentence is spoken aloud by the robot, and the words of the spoken sentence appear at the top of the screen.

[3]this procedure is determined by an .xml file specified by the experimenters for each story. For more detail, see [Gordon et al. (2015)]

The robot itself was framed as a younger peer. It greeted the child by saying "Let's play word games together!" and was introduced by the experimenter as "a young robot who has just learned to speak, and wants to learn to read." This framing encourages the child to treat the robot as a peer, rather than as an authority figure, and casts the evaluative portions of the interaction in a more playful, inquisitive light rather than as a formal 'test'. Prior to the interaction with the robot, the child *did* complete a formal test. The experimenter administered the TOWRE [Torgesen et al. (1999)] test, to assess the child's initial level of reading ability.

During the storytelling phase of the interaction, the child would move the different characters across the scene. After each 'action' taken by the child, the robot would speak a line of the story, procedurally generated by the game, based on both a pre-determined script and which characters the child had recently moved. The spoken sentence (e.g. "The dragon goes to the bird") would also appear in written form at the top of the tablet.

50% of the time, after the robot had spoken a sentence, the robot would prompt the child to read one of the words in the sentence by saying: "I don't know how to read the word [X], can you show it to me?" The game would then pause until the child tapped on one of the words in the sentence (shown at the top of the tablet). If the child tapped the correct word, the game would continue. Otherwise, the *tablet* would read the tapped word[4], thereby letting the child know whether she was right or wrong. If the child tapped the wrong word, she would be prompted to try again by the robot ("I don't think that's right. Can you try again?"). After two incorrect tries, the tablet would highlight and read the correct word, and the game would continue. On average, a child experienced 11 of these demonstration opportunities during the interaction. After the storytelling game, the robot asked the child to teach it a few more words. During this post-test, the robot again asked the child to identify a series of requested words, each presented as part of a full sentence, by tapping on the requested word in the sentence.

3.2.1 Tablet-only Interaction Participants

The preceding description describes the interaction for the majority of participants. However, there was a smaller control group that did *not* interact with the robot. In this condition, a white cardboard box was placed over the robot for the duration of the experiment, The children were not told there was a robot underneath the box, only that they would be playing word games on a tablet. Other than the presence of the covering box and the removal of experimenter references to the robot, however, the experiment proceeded exactly as described above. The robot's story 'speech' still came from the same speakers and the children were still prompted to read words, the underlying system did not change. The presence of this condition provides an ideal scenario to examine how children's emotional expressiveness varies with the presence or absence of a social robot during otherwise *identical* interactions.

3.3 Capturing Affdex Data

In this section I discuss the process for extracting and classifying affective data from video of the recorded interactions.

[4]the tablet was framed as a partial agent in the interaction as well, fulfilling the role of teacher by giving feedback to the child and robot

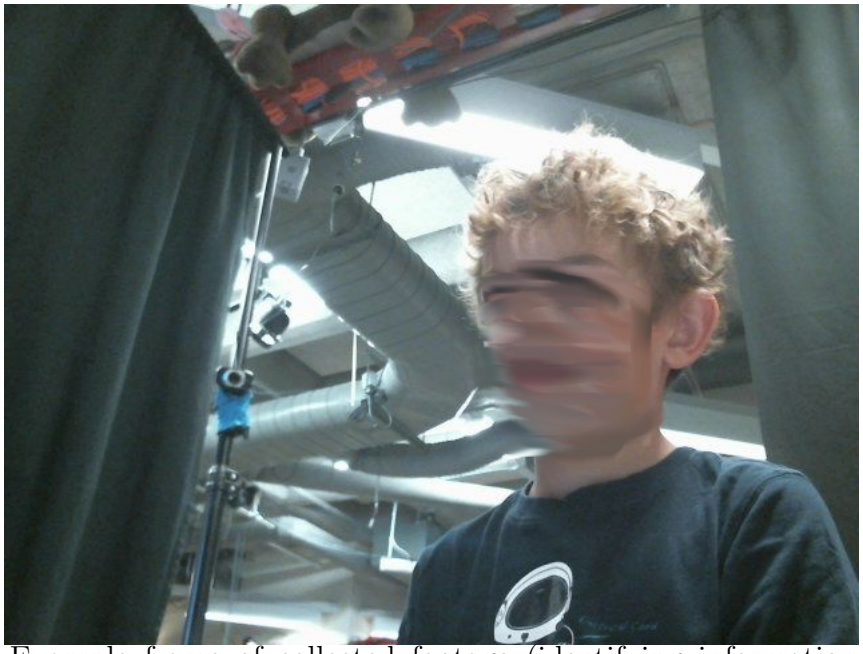


Figure 4: Example frame of collected footage (identifying information removed for publication). All video data was recorded at approximately this angle

Video of the entire interaction was recorded from multiple angles. In this thesis, I deal exclusively with video footage recorded from a single camera, located behind and to the side of the robot, aimed at the child's face (Fig. 4). In addition to video data, the full state of the game, all child actions on the tablet, all robot speech, and all tablet actions (e.g., highlighting the correct word if the child answered incorrectly twice) were recorded and synchronized via ROS, an open source robotics framework that features a unique data structure called ROS 'bags', synchronized records of multimodal data that support search, playback, re-recording, and many other functions[Quigley et al. (2009)].

Though complete footage of the interaction was recorded, in this thesis we are primarily concerned with analyzing children's emotions while they are engaged in educational activity. Therefore, we only analyzed video footage from specific times during each interaction. As discussed in the previous section, after a child moved the story characters, the robot would sometimes ask the child to read a word, denoted [X], by asking the child "I don't know the word [X]. Can you show it to me?" The robot was asking the child to identify word [X] in text on the screen, thus providing both an opportunity for the child to demonstrate her reading ability and an opportunity to collect spontaneous facial expression data in real educational tutoring scenarios. Because all video was collected several months before conducting the affective analysis, the first step was to identify the relevant portions of the interaction that contained the data of interest. As noted above, all data was logged via ROS.

By searching through the ROSbag logs, I was able to identify the precise times at which the robot *asked* a question (an "asking event") and the times at which the child *gave an answer to that question* (a "response" event). I extracted video footage from a window of 5 seconds before each asking event and 5 seconds after each response event, which constitutes a **"session"**. Each session corresponds to the time between the robot asking a single question and the child providing a response, plus 5 seconds of buffer before and after these points. In this chapter, which addresses Research Question 1 - whether children are more expressive when interacting with a robot - I primarily analyze the data at the *participant* level, that is, by aggregating the full set of a participant's sessions into a

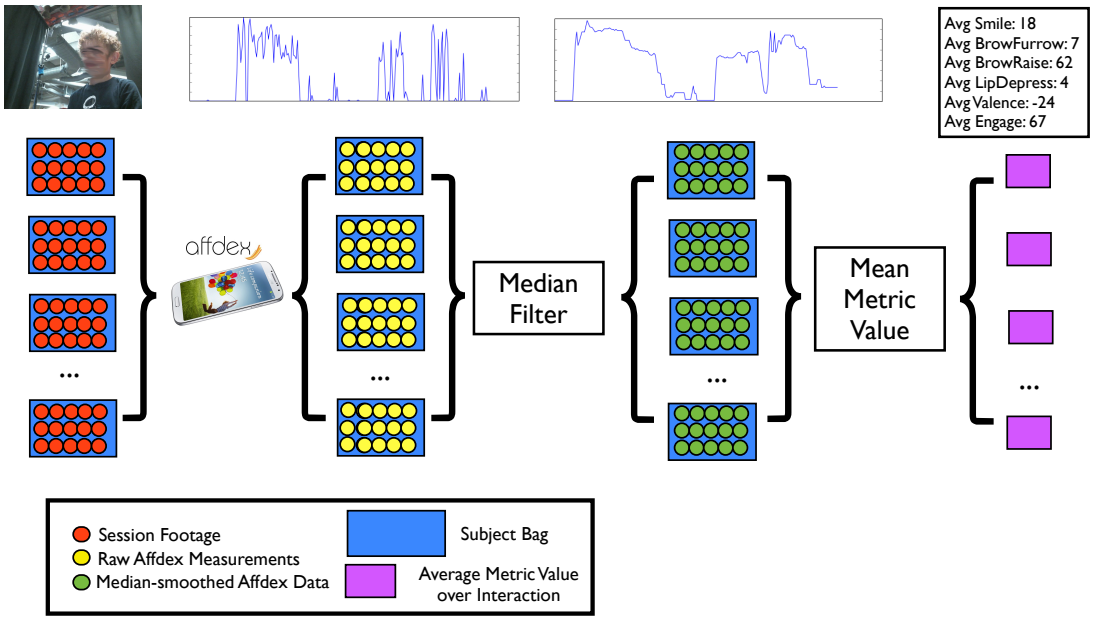


Figure 5: Complete data pipeline, from video footage of each session to average emotional expression over the full interaction. Still frames are sampled from session video footage at 20fps (depicted as red circles), then passed to the Affdex SDK application to produce unfiltered measurements for each frame, depicted as yellow circles. These measurements are then passed through a median filter for smoothing, producing *data points* for each metric (depicted as green circles). Finally, for each participant, the data points from each metric are averaged to produce a mean metric value across the entire interaction, depicted as purple blocks.

single point of participant data. In the subsequent chapter, which deals with the question of whether emotional signals can improve computational tutor inference, a session forms the basic time step unit of analysis. Section 4.1.2 details the construction and training of different models of student knowledge from session data.

3.4 Measuring Emotional Expressivity

After identifying the video footage that made up each session, I needed to extract an understanding of the emotional expression in each session. I sampled still frames from video footage of each session (each child experienced 29 sessions, on average) at approximately 20fps. These frames were then loaded onto an Android phone (as of March 2014, Affdex only supported iOS and Android SDKs for public use) for analysis. Using the Affdex Android SDK, I wrote a simple app that, for each session, created a Detector object and analyzed all frames in that session, using the temporal algorithm described in Sec 3.1. The output of the algorithm was, for each frame, either a single measured value for each of the 6 metrics, or a null measurement (indicating that no face was identified in the frame). Each frame result was then written to file, and manually transferred to a secure data storage location for analysis.

Autonomously sensed affective data is difficult to acquire and even more difficult to interpret. It was no surprise, then, that the initial measurements from Affdex were noisy, sporadic, and highly variable, even on a frame-by-frame basis. Often, frame results would

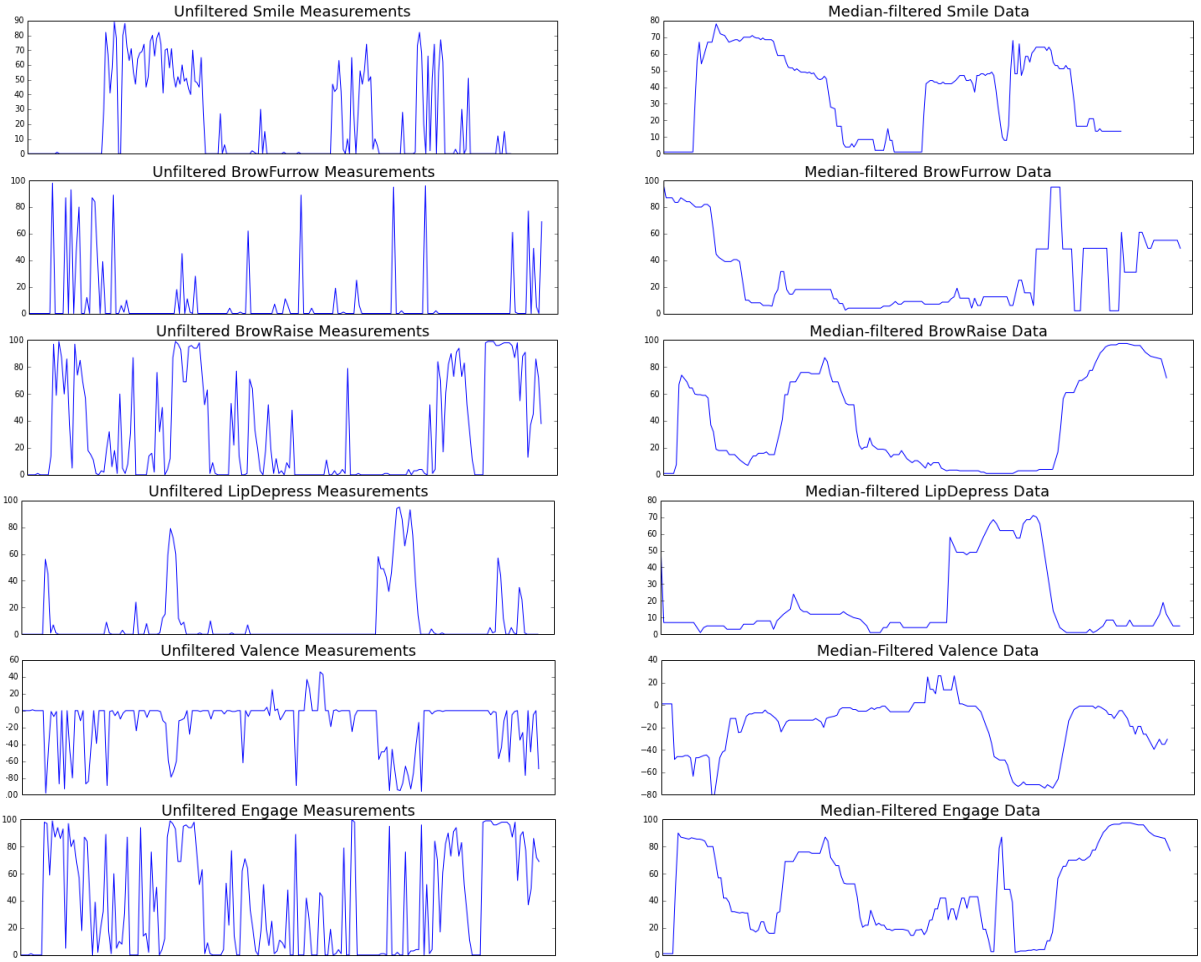


Figure 6: Line graph of measurements from Affdex, before and after median filtering

alternate rapidly between consistent, modest values (indicative of correct processing and analysis) and null results (indicated by Affdex reporting "No Face Found") or extreme results (e.g., minimum or maximum values for all metrics). In order to smooth out these discontinuities, I applied a median filter to the data, operating over a sliding window of approximately 1s, or 20 frames. More concretely, for each frame in the measured data, I computed a median of the 10 measurements before that frame and the 10 measurements after it. For the purposes of this computation, null measurements or measurements of 0 were excluded from the median[5]. If, excluding null or 0 results, at least 25% of the measured values in the sliding window were valid, then I computed a new *median-smoothed* **data point** for that frame; otherwise, that frame was marked as "no data". This process is visualized in Figure 6, which depicts results of the unfiltered measurements received from Affdex and the same results after smoothing via the median filter. For clarity, I will henceforth refer to the raw results computed by Affdex as *measurements*, and the corresponding median-smoothed values as *data points*. Note that a frame with a null measurement could generate a valid data point if, e.g., it were surrounded by valid (i.e. non-null or zero) measurements. Similarly, a frame with a valid measurement could fail to generate a valid data point, e.g., if it were surrounded by null or zero measurements. The end result of this median smoothing process is a set of data points, somewhat smaller

[5]Measurements of 0 were so frequent, compared to nearby measurements of (e.g., 1 or 2) that I treated them equivalently to null measurements

than the size of the original measurement data, with significantly fewer spikes or rapid increases and decreases. However, when such increases or decreases are present, they are more likely due to a *bona fide* affective signal, rather than mere noise.

3.5 Analyzing Affdex Data

Following the median-smoothing process, I examined individual videos in an attempt to evaluate Affdex as a tool for developing affect-aware robots. For each of the 6 Affdex metrics discussed in Section 3.1, I calculated an "efficiency" score: the number of frames that generated a valid (median-smoothed) data point, divided by the total number of frames analyzed. The efficiency of a metric, m, across the footage of a participant p is given by:

$$E_{m,p} = \frac{\text{\# of valid data points of } m \text{ in Participant } p\text{'s sessions}}{\text{total frames in Participant } p\text{'s sessions}} \tag{1}$$

Thus, for each metric we can identify how well Affdex was able to extract valid data, in essence identifying which metrics are most reliably sensed by Affdex. In addition, I combined all six metrics at the participant level, in order to identify and eliminate data from low-efficiency participant footage, that is, footage from interactions in which Affdex did not produce much data. All of a participant's sessions occurred during a single interaction, and were recorded in the same ROSbag. Therefore, I refer to the set of footage from which a participant's sessions were derived as simply "Participant p's bag".

Figure 7 depicts a histogram of participant bag efficiency, E_p, over all metrics.

$$E_p = \frac{\sum_{m \in M} E_{m,p}}{|M|} \tag{2}$$

Bags which had an efficiency score below 20% were excluded from subsequent analysis. 8 bags (out of 50 total) fell into this category. Of these, 2 bags had an efficiency score of 0, indicating that *no* valid data was extracted from these bags, and 2 more were below 5%. Subsequent review of the footage from these bags indicated that this was likely to happen if, for instance, the video had been recorded at an unusual viewing angle or if the child's face was simply not processed well by the Affdex classifier. Figure 8 shows the average efficiency of each metric over *all* bags (excluding those that met the low-efficiency exclusion criteria discussed above).

When interpreting these results, it is important to remember that low efficiency does not necessarily imply that Affdex is less accurate at detecting these metrics. The efficiency score is based on the overall amount of non-null, non-zero affect measurement, thus it depends on both how much genuine affective signal there is to be measured, in addition to how well it is processed. In this sense, the efficiency score is a measure of *the total amount of useful information each metric provides.* The efficiency of each metric will provide some guidance when considering how to incorporate affective information into a predictive model (Section 4.4).

What, then, can be inferred from the efficiency scores of each metric? Engagement and Valence have high efficiencies - possibly because they are aggregate metrics that are not *directly* dependent on only a few physical facial action units. Of the physical metrics, BrowRaise is a curious outlier, it has much higher efficiency than any other physical metric. This, combined with the high efficiency of engagement, could imply that children

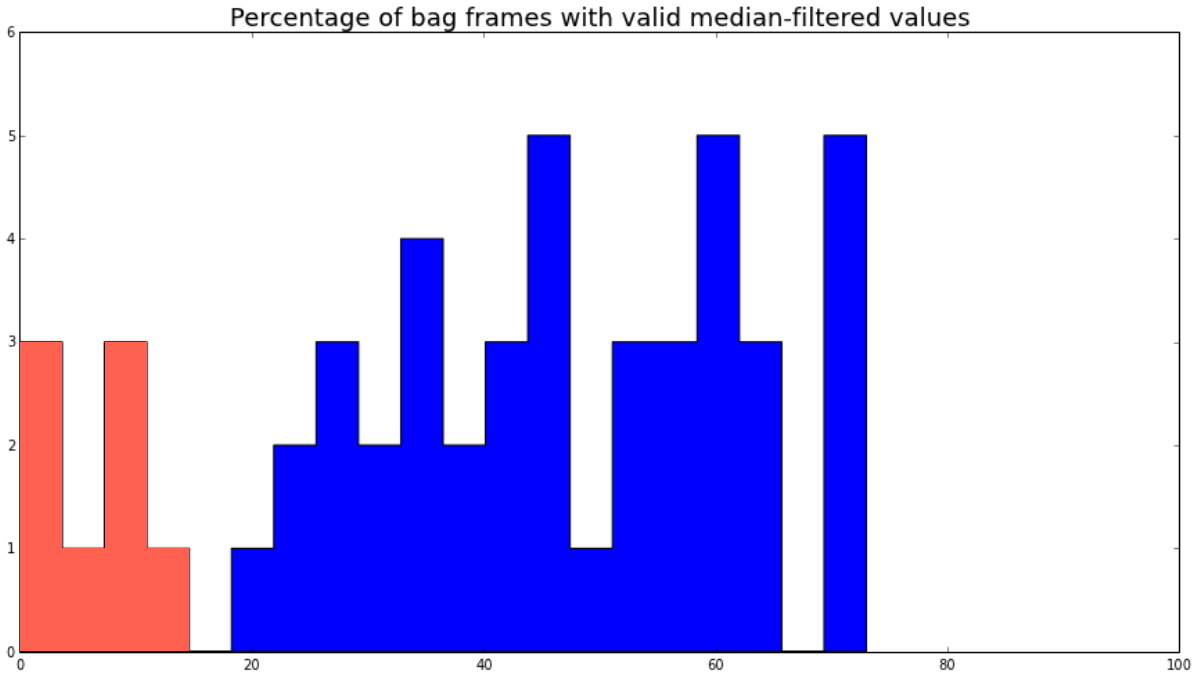


Figure 7: Histogram of participant data by percent of frames that generated valid data points. The eight participants (shown in red) whose footage generated valid data from $< 20\%$ of frames were excluded from all further analysis.

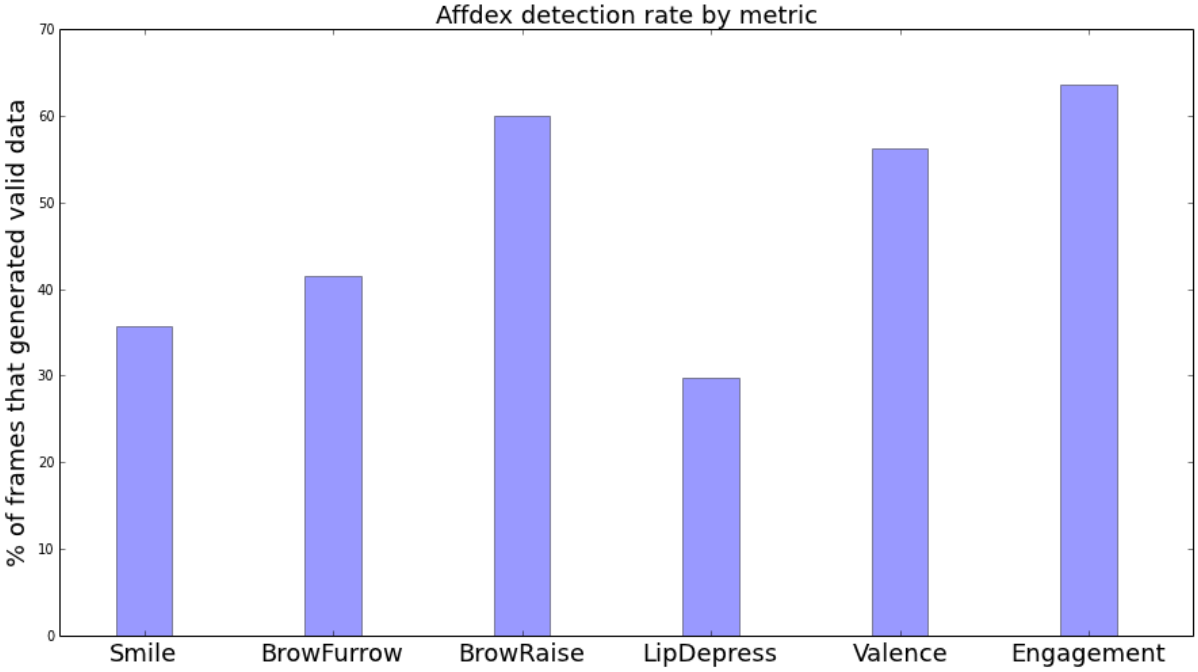


Figure 8: % of frames that generated a valid data point, for each metric.

were, in general, highly engaged during the interaction. Ultimately, however, efficiency is a somewhat limited statistic from which to draw conclusions *in isolation*. In the next section, we will revisit the efficiency results in the context of additional data on the overall level of emotional expressiveness in both robot and tablet conditions (i.e., what was the average non-zero, non-null value recorded by Affdex?)

Table 1: T-test results by metric, comparing differences in mean interaction value across Robot and Tablet conditions

Metric	T-test statistic value	p-value
Smile	2.23	***0.037***
BrowFurrow	0.898	0.378
BrowRaise	2.68	***0.011***
LipDepress	0.328	0.746
Valence	0.844	0.405
Engagement	2.59	***0.014***

3.6 Assessing Overall Emotional Expression

In order to determine a participant's overall level of emotional expressiveness, I combined the data points from all of a participant's question-response sessions into a single 'bag' of analysis. Within these bags, for each metric, I calculated the *mean data point value*: in other words, the average value, over the entire participant interaction, of the median-filtered Affdex measurements. Thus, for each metric, I obtained 27 mean interaction values from participants that interacted with a robot, and 13 mean interaction values from participants that interacted with a tablet. Because valence, unlike other metrics, ranges from [-100, 100] rather than [0,100]. In order to analyze all metrics within the same scale, I computed the mean of the *absolute value* of the valence data, assuming that scores of -50 and 50 represent equal amounts of 'expressivity' (in opposite directions). For each metric, the means and standard error of the mean for each population (Robot vs. Tablet) are shown in Figure 9. Figure 5 depicts the complete data analysis workflow, from video footage to average emotional expression data.

3.7 Results

For each metric, I first conducted a Shapiro-Wilk test on both populations to establish whether the distribution of mean interaction values was non-normal. The results in all cases were negative - there was not enough evidence to conclude that the data was non-normal. Then, for each metric, I conducted a Student's T-test on the robot and tablet sets of interaction means to determine whether there was a statistically significant difference between the emotional expression data from the robot condition and that from the tablet condition.

The test results showed that children in the robot condition did generate higher average emotional metric values, compared to the children in the tablet-only condition. These differences were statistically significant for 3 of the 6 metrics: **Smile, BrowRaise, and Engagement**. While none of the other metric differences reached significance, the average interaction value was higher in the robot condition for those metrics.

The overall expressivity results in Figure 9 can shed some additional light on the efficiency results of Figure 8. For instance, BrowRaise is an outlier among physical metrics (Smile, BrowRaise, BrowFurrow, LipDepress) in efficiency, yet the average recorded value is second lowest. Similarly, the BrowFurrow metric has only average efficiency, but has the highest average recorded value. One possible explanation is that throughout the interaction, children's affective mood tended towards modest surprise - leading to more consistent non-zero values for BrowRaise, but fewer large values. By this explanation,

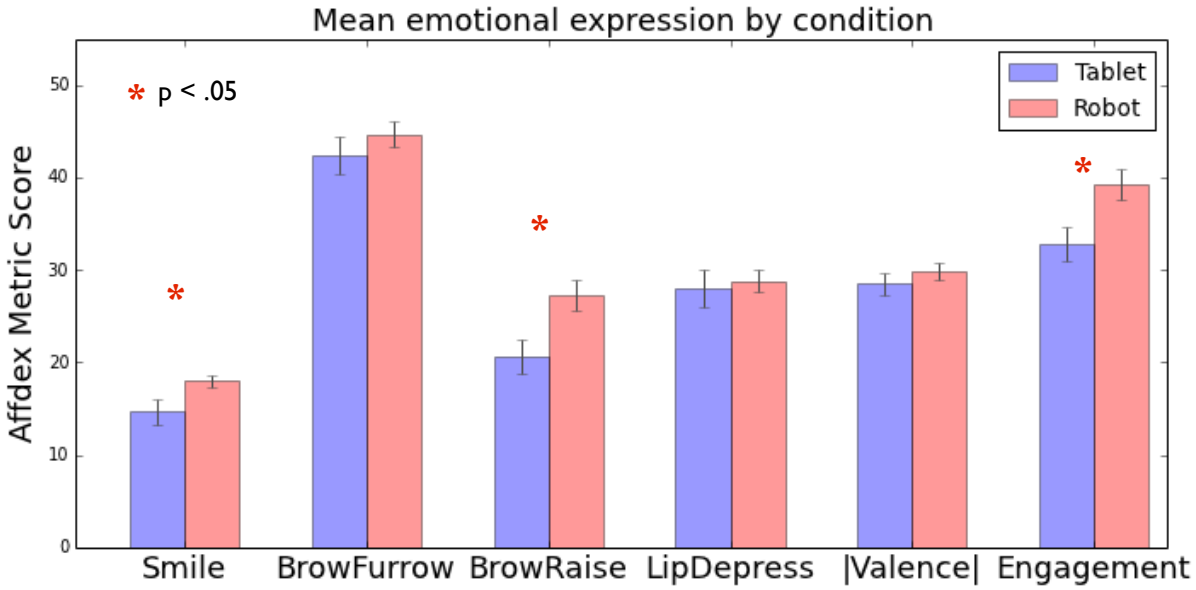


Figure 9: Average emotional expression value, by condition. Error bars represent standard error of the mean.

the relative inefficiency of the BrowFurrow metric could be attributed to the idea that children did not generate BrowFurrow values as often, but when they did, they were intense. The overall picture - spending most of the interaction in a state of modest, pleasant engagement, punctuated by some severe moments of confusion or frustration, matches the experimenter's subjective assessment of the interaction.

3.8 Conclusion

Based on our analysis of the data, we can see that children produce higher emotional expression values overall when interacting with a robot, with significant differences in three metrics - Smile, BrowRaise, and Engagement. It is not entirely surprising that children interacting with a robot would exhibit the largest increases in these three metrics, previous research has suggested that children tend to smile and be highly engaged around social robots. What is exciting about this research, however, is that we were able to demonstrate this *quantitatively* and *autonomously* (i.e. without human judgement). Now we turn our attention to the second research question: can we use these emotional expression measurements to improve the performance of ITS assessment algorithms?

Summary: In this thesis, I describe work that seeks to answer two questions. The first question is "Are children more emotionally expressive in an educational interaction with an embodied social robot, compared to just a tablet?" In this chapter, I have introduced our working definition of "emotional expression" (defined by the Affdex sensor), detailed the process for extracting meaningful information from large quantities of affective data, and confirmed that children who interacted with a robot during an educational task generated higher average emotional expression readings than children who did not see the robot. These results reached statistical significance in three of six metrics – Smile, BrowRaise, and Engagement – confirming previous research that the mere *physical presence* of robots can significantly and positively alter the dynamics of an educational interaction.

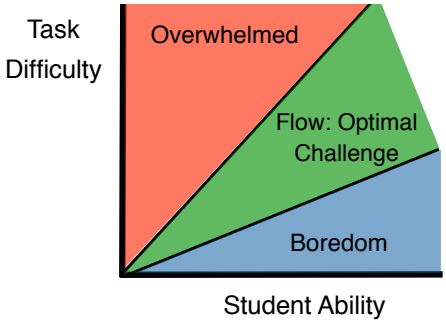


Figure 10: A visualization of "Flow"

4 Can Emotional Signals Improve the Performance of Computational Tutors?

In this chapter, I address the second research question: "Can we design algorithms and models that incorporate students' emotional expression data in order to more accurately model student learners?" I describe work to construct models, trained from students' affective data, that are capable of inferring an individual student's specific skill level while simultaneously modeling a profile of affective responses.

Students' emotional reactions are highly diverse. Some students may display signs of boredom because they find the material too easy, while others may show signs of boredom out of frustration at a task beyond their ability. Some students may answer questions quickly if they know all the answers, while others may diligently spend their time double-checking. Some students who encounter a difficult problem may choose to skip past it, without wasting time. Others may dedicate a substantial amount of time to a challenging problem, only to realize they lack the necessary skills to solve it. The work described in this thesis represents a crucial step towards the development of models that can adapt to these diverse learning styles in order to better provide personalized tutoring.

In this chapter, I first introduce some additional background material, followed by a detailed description of what models were created and how they incorporate affective data. Then, I describe the evaluation procedure, present the results, and describe the implications and conclusions. Last, I include a discussion of how this work fits into the larger context of affect-aware tutoring research.

4.1 Research Background and Modeling Approach

In this section, I introduce (1) the Knowledge Assessment problem for Intelligent Tutoring Systems, (2) The Bayesian Knowledge Tracing algorithm (a widely-used approach to solving knowledge assessment), and (3) the *Affective* BKT model (Aff-BKT), one of the main contributions of this thesis, an augmentation of the BKT algorithm that incorporates information about the student's facial expressions during problem-solving as additional observation features to drive inference.

4.1.1 Knowledge Assessment

In Section 2, I discussed the concept of Flow - a state in which a student is highly engaged in a task at the optimal challenge level. Flow is considered to be a key component of education[Csikszentmihalyi (1997)]. One approach used by affect-aware tutors is to monitor whether a student is in Flow and, if not, attempt to guide him back to a Flow state.

With this in mind, I now introduce the central computational problem of this chapter: the Knowledge Assessment problem. The Knowledge Assessment problem is, in plain language, "How can a teacher determine what a student does or does not know?" The Knowledge Assessment problem is important for any tutor that wishes to keep her students in a state of Flow. Consider Figure 10: if a student's skill level is far above the level required by the presented content, he may grow bored or feel unmotivated to continue. If his skill level is far below the level required by the presented content, he may feel overwhelmed or discouraged and will be less likely to try to master the new content. Flow occupies the space in between the two extremes – where the student's level of skill is acceptably close to the level required by the content. One of the aims of a skilled tutor is to keep her student in that space. Of course, a tutor does not have (direct) control over her student's skill level, but she *can* control the content. If the tutor knows a student's skill level within a certain degree of accuracy, she can select the appropriate content to keep her student in a state of Flow. Hence, arriving at an accurate estimate of a student's skill level is a crucial step for all tutors that wish to keep their students engaged and challenged. But how does one assess another's knowledge?

Testing is one option, but there are many other approaches used by expert human teachers - particularly in one-on-one tutoring scenarios. Some examples include direct dialogue, variations on the Socratic method (e.g., asking the student to explain a concept to the teacher), or observing the student's emotions and workflow as he tries to solve a problem. For computational tutors, Bayesian inference on graphical models is the most popular approach. If the subject domain can be suitably modeled, algorithms for inference can allow computational systems to estimate student skill levels. The most popular and widely used of these methods is known as Bayesian Knowledge Tracing (BKT), a domain-general algorithm for inferring skill mastery from student data. BKT models are widely used in ITS research [Baker et al. (2008)]. For readers interested in a deeper understanding of the history and complexities of Knowledge Assessment, VanLehn[VanLehn (2008)] provides an excellent summary of both the problems and promise of "continuous, embedded assessment" in ITS research.

4.1.2 Bayesian Knowledge Tracing

Overview: Bayesian Knowledge Tracing (BKT) is perhaps the most widely used computational model for assessing student knowledge. BKT is a general modeling approach to solving the Knowledge Assessment problem that can be applied to any educational domain which can be decomposed into different component "skills"[Corbett and Anderson (1994)]. Under the BKT model, these different educational skills are encoded as nodes in a Bayesian network. Each "skill node" in the network represents a student's understanding of a specific skill. The models used in BKT are a special case of Hidden Markov Models[6], in which each skill node is assumed to be in one of two hidden states, "skill known" or "skill not-known", and the observables are binary evaluations of answers to questions requiring knowledge of a particular skill (i.e., was the student's response to this question Correct or Not Correct?).

Skill nodes are not directly observable - a tutor cannot see the *actual* state (whether a student knows or does not know some skill). Instead, a BKT model maintains an

[6]Hidden Markov Models (HMMs) are a general class of Dynamic Bayesian Network models that have been applied, with great success, to problems in speech technology, bioinformatics, and many other fields of science and engineering

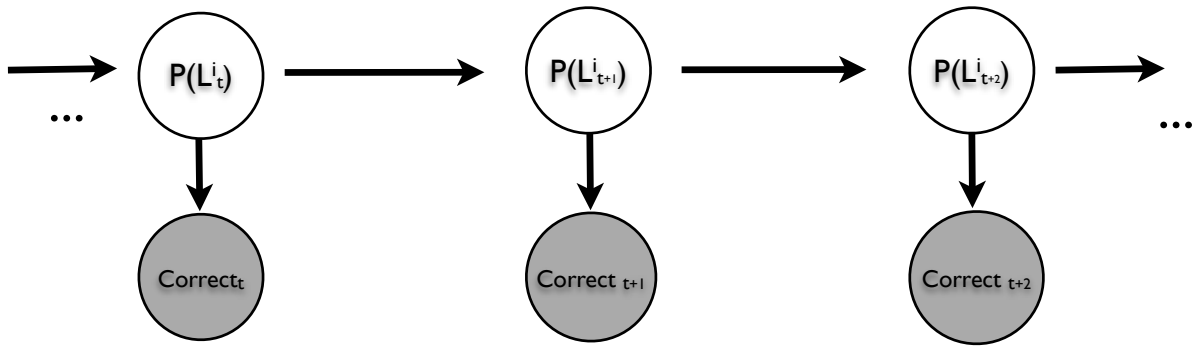


Figure 11: Standard BKT Hidden Markov Model. From patterns of correct and incorrect responses, the model can infer the student's hidden knowledge state

estimate, a probability distribution, that represents the probability that a student is in each state (skill known or skill not-known). As an example, let us consider modeling the mastery of the skill "solving quadratic equations." Initially, we have no knowledge about whether the student does or does not understand how to solve quadratic equations, so one reasonable prior distribution might be $\{0.5, 0.5\}$, we believe there is a 50% chance the student knows how to do so and a 50% chance he does not.

The power and flexibility of the BKT model is the notion of the Bayesian update. In the face of new data (e.g., observing that a student answers three questions correctly about solving quadratics), we would want to update our estimate of how likely it is that the student has mastered the skill in question. Bayes' Rule concretely formalizes what the posterior distribution should be and how to calculate it. Each question/response pair is presented/observed at a unique, discrete time point, and the hidden state can change from 'not-learned' to 'learned' in between any two consecutive time points (corresponding to the idea that a student can learn a skill at any time).

As noted above, a BKT model is a special case of a full Hidden Markov Model (HMM). A two-state HMM with a single, binary observation is fully specified by 5 parameters, described in Table 2. These parameters determine how the knowledge state estimate changes in response to new data, and govern the relationship between hidden states and between a hidden state and an observable. A single parameter (the prior) also specifies the initial conditions of the model. From these 5 parameters, we can calculate precise, numerical answers to a wide range of relevant questions, such as: "If a student solves 3 questions incorrectly, then solves 2 questions correctly, what is the probability he has mastered the skill?"; "If a student has solved 3 questions correctly, what is the probability he will answer the next question incorrectly?"; "If a student solves a problem incorrectly, what is the probability it was due to a slip-up on his part, rather than a genuine lack of mastery?"

Michael Jordan describes graphical models as "a marriage between probability theory and graph theory." The graphical representation allows for human-readable semantics, while its probabilistic nature admits a wide ontology of phenomena that can be appropriately modeled. In the BKT model, a parameter can be interpreted as a number *per se*, the fuel that drives the engine of inference in an HMM. But parameter values can also

Table 2: BKT model parameters and interpretations

Notation	Interpretation as HMM	BKT Semantic Interpretation
$\Pr(L_0^i)$	For model i, the prior probability of being in state L at time 0	The probability a student already knows skill i prior to the start of tutoring.
$\Pr(\neg L_{t+1}^i \mid L_t^i)$	For model i, the probability of transitioning to state $\neg L$ at time $t+1$ given the model was in state L at time t	The probability a student *forgets* skill i in between time steps. Sometimes called the **'forget'** parameter.
$\Pr(L_{t+1}^i \mid \neg L_t^i)$	For model i, the probability of transitioning to state L at time $t+1$ given the model was in state $\neg L$ at time t	The probability a student learns skill i in between time steps. Sometimes called the **'learning rate'** parameter.
$\Pr(Correct_t^i \mid \neg L_t^i)$	For model i, the probability of seeing observation $Correct$ at time t given the model was in state $\neg L$ at time t	The probability a student answers question t (which requires skill i) correctly, given that he does not know skill i. Sometimes called the **'guess'** parameter.
$\Pr(\neg Correct_t^i \mid L_t^i)$	For model i, the probability of seeing observation $Correct$ at time t given the model was in state L at time t	The probability a student answers question t (which requires skill i) incorrectly, given that he knows skill i. Sometimes called the **'slip'** parameter.

be interpreted as degrees of belief[Jaynes (2003)]. And if the HMM is a BKT model, these degrees of belief correspond to specific factors relevant to the context of inferring whether a student has mastered a skill, based on his pattern of correct or incorrect answers. The notation of these 5 parameters, alongside their functional interpretation in the HMM and semantic interpretation in the context of Knowledge Assessment, are given in Table 2.

Because BKT models are intended to provide semantically useful information, they are typically subject to constraints on parameter values. Models with values that violate these constraints are known as *degenerate* models, models in which the semantic interpretations of parameter values no longer accurately reflect the world[van De Sande (2013)]. For example, one assumption typically enforced on the BKT model is that a correct answer is more likely due to a proper application of the skill than a guess. Similarly, a mistake should be more likely to be caused by not knowing a skill than a slip. Mathematically, a model in which $\Pr(Correct_t \mid \neg L_t) > \Pr(Correct_t \mid L_t)$ is degenerate because it implies that a correct answer is more likely to come from not knowing the related skill. Similarly, a model with $\Pr(Correct_t \mid L_t) < .5$ and/or $\Pr(Correct_t \mid \neg L_t) > .5$ is degenerate because it implies that if a student knows a skill he is more likely to answer questions related to that skill *incorrectly* than correctly. These assumptions may seem obvious, but previous work has found that when all parameters are allowed to range freely during model training, BKT models tend towards degeneracy[Baker et al. (2008)].

Limitations: The BKT model, while popular for its relatively straightforward analysis and ease of implementation, does suffer from some limitations. Most significantly for this thesis, the BKT model relies solely on a student's pattern of correct/incorrect

answers to drive inference, while ignoring the vast ocean of relevant contextual information that good human tutors draw on to assess students, such as a student's affective expressions. The Affective BKT model described in the following section (Aff-BKT) is an attempt to improve the BKT model, by augmenting it to draw inference from *affective* as well as knowledge-based features.

BKT's emphasis on information from correct/incorrect answers has broader implications for the design of ITSs as well. Because correct/incorrect answers are a relatively weak channel of inference (compared to a system that uses information from many multimodal sources), in order to arrive at an accurate estimate of a student's knowledge, such models typically require large datasets. In turn, this requirement steers the interaction design of BKT-based ITSs towards frequent prompting of the student (culminating, in extreme case, in 'digital workbook' style tutors), rather than the interactive and immersive styles of tutoring that characterize the best human teachers. Additionally, BKT skills are most commonly treated independently, which limits the kind of curricula that can be successfully modeled by BKT. There are some examples of using implementing BKT with hierarchical or interdependent skill models[Ferguson et al. (2006)] in which demonstrating knowledge of one skill influences the probability of mastery of other skills that may not be *directly* relevant. However, this is still an area under active research, as more advanced model structures greatly increases the complexity of training and can require much more data. In this thesis, both BKT and Aff-BKT models treat each skill independently.

In spite of BKT's limitations, it is still one of the most popular methods for knowledge assessment and forms the basis of a substantial body of research. While the Aff-BKT model does not address all of the limitations of BKT, it representsan initial advance. The work presented here serves as proof-of-concept that affective information can be reliably detected and made useful, and that models that take advantage of such information can outperform standard techniques, all while keeping the tutoring interactive and engaging.

4.1.3 Affective-BKT

The Affective-BKT (Aff-BKT) model forms one of the central contributions of this thesis, presented for the first time. It is a BKT model with additional observation nodes representing features of the student's facial expression during an educational interaction with a social robot. That is, the Aff-BKT model is a Hidden Markov Model with multiple observations per time step, whereas the standard BKT model has only one. In addition to the observable node corresponding to a correct/incorrect response, the Aff-BKT model includes observable nodes that correspond to affective features, determined by analyzing a participant's facial expression. The additional observable nodes are structured identically to the standard BKT observation of correct/incorrect answers (see Fig. 11), therefore each additional node requires just two additional parameters per skill model. These additional parameters and their semantic interpretations are given in Table 3.

As noted in Section 2, much of the work on affect and machine learning models in the ITS literature focuses on trying to *infer* affect. Typically, once affective states are detected or identified, simple behavioral rules are triggered - the affective state is not used to improve or train any internal model. There is, however, one notable exception: a recent project by Xu et al. uses EEG input as an input to a Knowledge Tracing model [Xu et al. (2014)]. However, EEG signals are extremely noisy, subject to a large degree of individual variance, and lack a clear semantics. Thus, while the models incorporating EEG data did exhibit slightly improved performance, the utility of such models is

Table 3: Additional parameters used in the Aff-BKT model and their interpretations

Additional Aff-BKT Parameters	**Interpretation as HMM**	**Semantic Interpretation**
$\Pr(Smile_t^i \mid L_t^i)$	For model i, the probability of seeing observation *Smile* at time t given the model was in state L at time t	The probability a student smiles while answering question t (which requires skill i), given that he knows skill i.
$\Pr(Smile_t^i \mid \neg L_t^i)$	For model i, the probability of seeing observation *Smile* at time t given the model was in state $\neg L$ at time t	The probability a student smiles while answering question t (which requires skill i), given that he does **not** knows skill i.
$\Pr(Engage_t^i \mid L_t^i)$	For model i, the probability of seeing observation *Engage* at time t given the model was in state L at time t	The probability a student appears engaged while answering question t (which requires skill i), given that he knows skill i.
$\Pr(Engage_t^i \mid \neg L_t^i)$	For model i, the probability of seeing observation *Engage* at time t given the model was in state $\neg L$ at time t	The probability a student appears engaged while answering question t (which requires skill i), given that he does **not** know skill i.

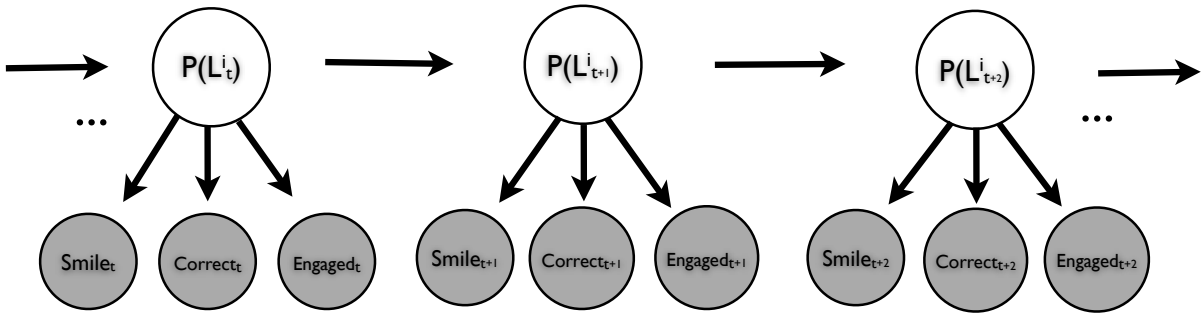


Figure 12: The Affective BKT model, which incorporates affective signals into knowledge state inference.

limited, as they do not readily lend themselves to autonomous model construction nor interpretation by researchers or educational experts. In contrast, the Affective-BKT models described in this thesis have a clear semantics and the resultant, trained parameters can yield intriguing, data-driven insights into the intersection between emotion and learning.

Summary: In this section, I have introduced the Knowledge Assessment problem – the central computational challenge of this thesis – and its motivations. I have also described the Bayesian Knowledge Tracing model, a special case of a Hidden Markov Model that supports inference to solve the Knowledge Assessment problem. Finally, I introduced a novel extension of the BKT model: Aff-BKT, which incorporates estimates of affective states, derived from facial expressions, into its inference. Next, I describe the fundamental processes by which I implemented and evaluated these models, in the

context of assessing students' *reading skills*. First, I cover how to computationally model a complex task like reading, followed by detailing the structural implementation of each model, how the training data was obtained, how the parameters for each model were trained from the data, and how the models were evaluated.

4.2 Modeling the reading process as discrete alphabetic principle skills

In this section I discuss how I chose to represent and assess students' mastery of a complex educational task, such as reading, within BKT and Aff-BKT models. Reading is an extraordinarily complicated process, yet despite its incredible complexity, almost all adults living in developed countries are literate. Learning to read is an acquired skill that requires years of practice and mastery of many foundational sub-skills before fluent "reading" occurs. One particular class of skills is known as the "alphabetic principle" skills. The alphabetic principle, in its simplest formulation, is the recognition that *written* letters and their combinations correspond to particular *spoken* sounds in specific and predictable ways. Alphabetic principle skills are the various rules and mappings that connect those written combinations with the more familiar spoken sounds. The alphabetic principle and its associated skills are fundamental precursors to developing full literacy, and understanding how to best teach those skills is an area of significant research (see [Byrne and Fielding-Barnsley (1989), Liberman et al. (1989), Byrne (1998)], *inter alia* for an overview).

In this thesis, I model the complex act of reading a word by constructing and tracking three different BKT models, corresponding to three alphabetic principle skills: correctly recognizing the first grapheme[7] of a word, correctly identifying a word of (roughly) the same written length as a spoken word, and correctly recognizing the final grapheme of a word.

The first skill, denoted First-Letter, requires the child to hear a requested word, decompose its sounds into phonemes, map the phonemes into graphemes, and then select a word that features those graphemes at the beginning of the word. Roughly, it corresponds to the first step of the oft-repeated advice to a young reader - "Sound it out!" Homophonous graphemes (e.g., 'H' and 'Wh') were considered equivalent, as resolving this type of collision in a phoneme-grapheme mapping is considered a more advanced skill.

The second skill, denoted Length, requires the child to understand that the length of a written word corresponds to the number of syllables in its spoken instantiation. For this skill, I considered a student to have correctly demonstrated this skill if they selected a word with a length within 1 letter of the requested word's length.

The third skill, denoted Last-Letter, requires the child to go through the same process as First-Letter, only applied to the end of the word. This poses slightly more of a challenge, as it is natural to attempt to read a word from its start, thus I expect Last-Letter may depend somewhat on First-Letter as a precursor.

Lastly, I did model the complete, correct reading of a word. This skill, denoted Exact-Correct, requires the child to fully and correctly identify the requested word, representing the complete ability to read the requested word. Because the requested

[7] a grapheme fulfills roughly the same function in writing as a phoneme does in speech: it is the smallest unit used to describe writing from a *linguistic* perspective. Examples include *'a', 'ch', 'f', 'sh', 'd', etc.*

word was spoken aloud, homophones were considered correct (e.g. if the robot asked for the word 'to', and both 'to' and 'too' were among the possible answers, either would be considered 'exactly correct').

There are some limitations and tradeoffs to these modeling choices. As noted above, there are many fundamental sub-skills required for a complex act such as reading. Even within the subset of alphabetic principle skills, there are far more than I could cover in the scope of this thesis. I chose to analyze the four skills discussed above because they are well-suited to the task from which the dataset was derived (see Sec 3.2), relatively easy to detect computationally (compared to, e.g., analysis of a child's pronunciation), and because they are developmentally appropriate for the age group of the population [of Oregon Center for Teaching and Learning (2015)]. Lastly, though the alphabetic principle skills are clearly interrelated, BKT skills are most commonly treated independently. In line with the stated research focus of this chapter (examining whether *affective features* improve the model), in this thesis both BKT and Aff-BKT models assume that each alphabetic principle skill is learned independently of the others.

4.3 Building the BKT and Aff-BKT Skill Models

I constructed basic BKT models for each skill using Kevin Murphy's Bayes Net Toolkit [Murphy et al. (2001)], a freely available, open-source library that provides support for construction of and inference on a wide variety of graphical models. Each BKT model has one hidden node – with two possible states, (Skill Learned or Skill Not-Learned) – and one observation node – with two possible values (Question Correct or Question Incorrect) – per time step. Each BKT model has the structure depicted in Figure 11 and is completed by specifying the parameters listed in Table 2. I will describe how these parameters are ultimately learned from data in Section 4.4.

As with the BKT models, I constructed an Aff-BKT model for each skill using Kevin Murphy's Bayes Net Toolkit (BNT), implemented in Matlab. Each Aff-BKT model includes two additional observable nodes per time step – $Smile_t$ and $Engaged_t$. In the next section, I discuss how we derived the correctness and emotional training data and how that data was used to learn parameters for each skill model.

4.4 Training the BKT and Aff-BKT Skill Models

4.4.1 Deriving Session Skill-Correctness Data

In Section 3.3, I described the concept of a *session* - a window of time ranging from just before the robot asks for a word to be read to just after the child gives an answer. As described in Section 3.3, the robot's requested word and the child's response were recorded in ROSbags during the interaction.

To determine whether each session represented a correct or incorrect application of a skill, I wrote 'correctness' string matching functions for each of the four alphabetic principle skills. Each function compared the requested word to the answered word and computed whether the chosen word represented a correct application of the relevant skill to the requested word. For example, answering **'prince'** when the requested word was **'princess'** is a correct application of First-Letter, but not of Exact-Correct, Length, or Last-Letter. I then applied these functions to each of the requested and answered words in each session. The end result was, for each session, four boolean results (Correct or Incorrect) representing whether or not the child's answer represented

a correct demonstration of each skill (First-Letter, Length, Last-Letter, Exact-Correct).

4.4.2 Deriving Session Affect Data

In Section 3.4, I described the process for obtaining affective data from each session. For this analysis, I started with the set of median-smoothed data points (green dots in Figure 5) for each session. For each *session*, I calculated the mean value[8] of the Engage and Smile metrics, both of which have been used as features in prior affective-aware tutoring work. Each session was then labeled as Smile/No Smile and Enaged/Not Engaged via a mean value threshold of 30. If the average value of Smile was over 30 in a session, then it seems reasonable to assume that the child did, in fact, smile during that session and it is labeled as such. Similarly, if the average Engagement value was above 30 for the session, the child was probably fairly engaged during the session. The end result of this process was, for each session, two sets of boolean results (Smile/No Smile and Engaged/Not Engaged). Note that, unlike the correctness data, the affective data does not change with the skill being modeled. Next, I discuss how the affective and correctness data were used to train and evaluate parameter sets for the Aff-BKT and BKT models.

4.4.3 Expectation Maximization: learning model parameters from observed data

The parameter values for each skill model – 4 BKT and 4 Aff-BKT, one each per skill – were trained via Expectation Maximization (EM) from the training data. Expectation Maximization (EM) is a general technique for estimating a set of parameters from data. Under the assumption that some set of parameters, θ, generated a set of data, D, Expectation Maximization tries to find the Maximum Likelihood Estimate (MLE) of the parameters, given the data: $max_\theta Pr(\theta|D)$. For the BKT models, the data used to train each skill-model is the set of skill-correctness data (described in Sec. 4.4.1) for the corresponding skill. For the AKT skill models, the training data is the skill-correctness data *plus* the affective data (described in Sec. 4.4.2).

EM works by alternating steps. In the 'Expectation' step, the algorithm computes a function, $Q(\theta)$, that gives the expected value of the Log Likelihood function, with respect to the current parameter estimate, θ_t. In the 'Maximization' step, the algorithm then finds a new set of parameters, θ_{t+1} that *maximizes* the expected Log Likelihood. This new set of parameters then becomes the parameter estimate for the 'Expectation' step of the following iteration. Mathematically, this can be expressed as: $\theta_{t+1} = max_\theta Q(\theta)$. The process repeats until the parameter values converge (i.e., the difference in expected log-likelihood values between EM iterations falls below a certain threshold. That is, EM converges at time step t when $|Q_{t-1}(\theta_{t-1}) - Q_t(\theta_t)| < \epsilon$). In this thesis, we used $\epsilon = 10^{-5}$. All models converged in <15 iterations. Each BKT model took approximately 20 minutes to train; each Aff-BKT model took approximately 40 minutes to train.

[8]Whereas in Section 3, I computed the average metric value over a participant's complete interaction, for this analysis, I computed the average metric value for each *session*. Each participant experienced 29 sessions, on average.

Parameter	Initial Value	Justification for Choice
$\Pr(L_0^i)$	0.5	Initially, we have no knowledge of the child's reading ability, hence any child is equally likely to know or not know a given skill.
$\Pr(\neg L_{t+1}^i \mid L_t^i)$	0.0	BKT is most commonly used to model short interactions and assumes that a child will not 'forget' a skill, once it is mastered (e.g., [Ferguson et al. (2006)])
$\Pr(L_{t+1}^i \mid \neg L_t^i)$	0.2	There is a low probability that a child will learn a skill at any given time, but we expect some children to learn over the course of the interaction.
$\Pr(Correct_t^i \mid \neg L_t^i)$	.25	We chose .25 as the initial guess rate. Though the actual chance of answering correctly varies by both skill and session, it ranged from 11% to 33%.
$\Pr(\neg Correct_t^i \mid L_t^i)$	.25	We chose an initial slip rate equivalent to the initial guess rate.

Table 4: Initial parameter values used in training via Expectation Maximization

4.4.4 Initial Conditions

EM is a deterministic algorithm, and therefore performance is somewhat sensitive to initial conditions. Table 4 shows the parameters of the model, their initial values, and the justification for setting those values.

4.5 Bayesian Model Selection

The main research question this chapter addresses is whether we can improve BKT models by augmenting them to incorporate affective information. Thus, we must identify some way of comparing models. As discussed in Section 4.1.1, Knowledge State Assessment is difficult precisely because a student's *actual* knowledge state is not directly observable. Because we have no "ground-truth" data, it is infeasible to use traditional supervised learning benchmarks (e.g., precision + recall metrics or F-score), to evaluate the Aff-BKT and BKT models. Rather, I evaluated the two models from the perspective of Bayesian Model Selection.

Bayesian Model Selection is the general problem of determining which of several possible models should be preferred, given some data. Approaches to model selection vary based on the space of possible models and datasets under consideration, but generally revolve around calculating the probabilities of different models, given the observed data.

Equation 3 shows this equivalence mathematically. On the left hand side, we have the quantity we wish to calculate: the posterior probability of a model, θ, given some observed data, D. This quantity is proportional to the **prior probability** of θ (i.e, *a priori*, what do we think is the probability that theta is the correct model?) multiplied by the probability of the data occurring if θ determined the data generation process. This second quantity, $\Pr(D|\theta)$, is also called the **likelihood** of the model, given the data.

In Section 4.4.3, I described how I used the Expectation Maximization algorithm to learn model parameters for the BKT and Aff-BKT models. These parameters approxi-

$$\Pr(\theta|D) \propto \Pr(\theta) \times \Pr(D|\theta) \tag{3}$$

Equation 3: Bayes' Rule. The posterior probability of a model, after seeing some data, is proportional to the prior probability of the model multiplied by the probability of the observed data occurring under the model.

$$\mathcal{L}(\theta|D) = \Pr(D|\theta) \tag{4}$$

Equation 4: The **Likelihood** of a model, given some data, is equal to the **Probability** of the observed data occurring under the model.

mate the maximum likelihood model for their respective training data. Put another way, the EM algorithm tries to find a set of parameters that *maximizes* the probability of the training data occurring. These terms – likelihood of a model, given data and probability of data, under a model – are equivalent (see Equation 4), and are used interchangeably, whenever it makes more sense to refer to one or the other.

Based on equations 3 and 4, assuming equal priors for both models, we should prefer the model with the highest likelihood - i.e., the model under which the observed data has the highest probability of occurring. Simple likelihood-based metrics for evaluating a model, such as the Bayesian Information Criterion (BIC) or Akaike Information Criterion (AIC) metrics include a penalty term to account for additional parameters, hence they are appropriate for models with different structures. The BIC and AIC have been used to compare Knowledge Assessment models in which the number of parameters differ [Ferguson et al. (2006)].

Unfortunately, the models we wish to compare differ not only in their structure but also in the data they model and are trained from, and therefore straightforward likelihood metric comparison is not appropriate for this case. Informally, this is because the BKT training data is composed of a single boolean observation per time step, representing whether the student gave a correct or incorrect answer during that session. In contrast, the Aff-BKT training data is composed of *three* boolean observations per time step. Because there are more possible datasets in a 3-observation model, there is, on average, less probability mass to be assigned to each possible dataset. The probability of the larger dataset, under one model, is not fairly comparable to the probability of a smaller dataset, under another model, hence comparing the likelihoods of those two models (given different size datasets) is not a fair comparison either.

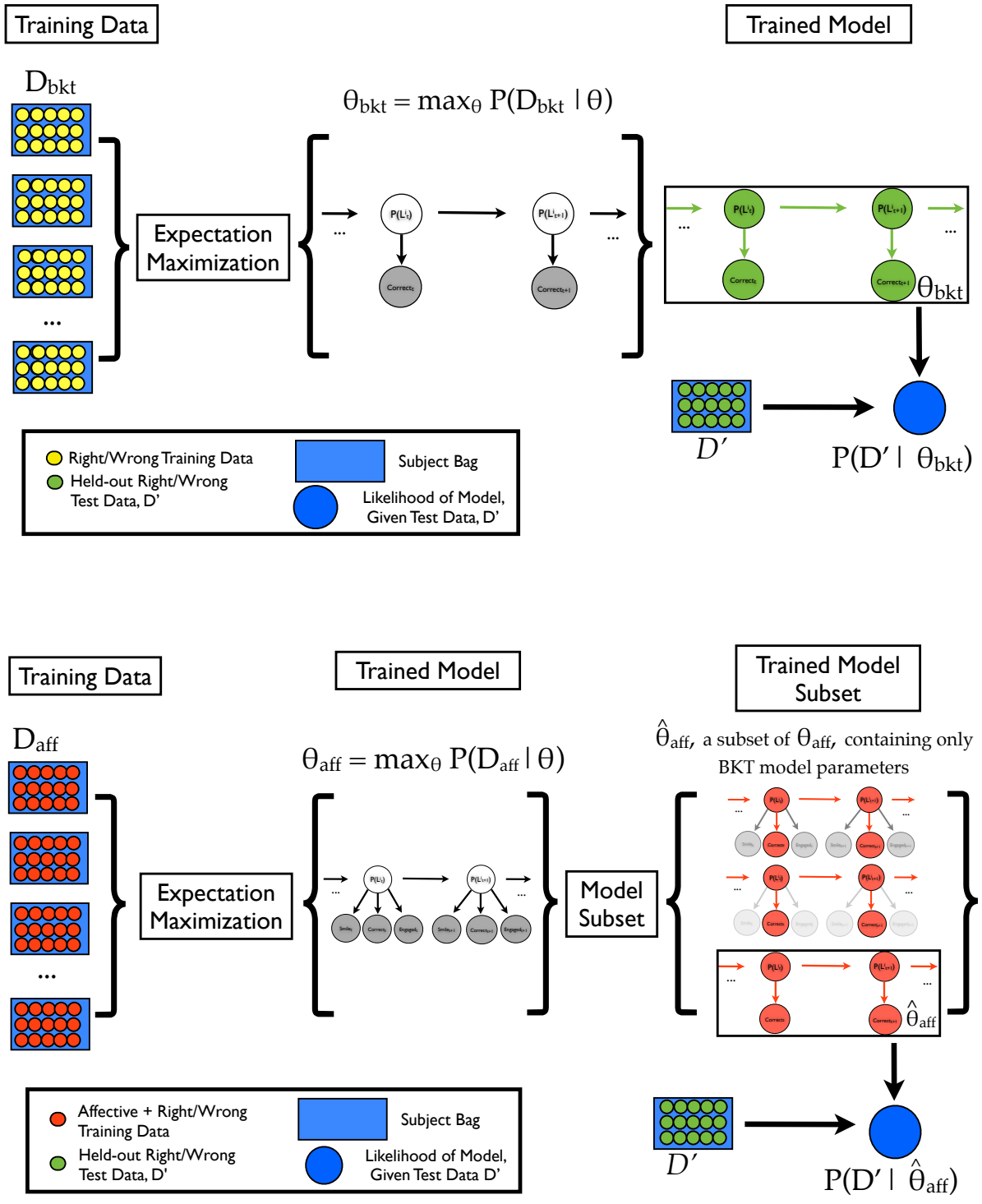


Figure 13: A single fold of the full training and evaluation pipeline for BKT skill models (top) and Aff-BKT skills models (bottom). The BKT models are trained from the skill-correctness session data described in Section 4.4.1, depicted as yellow circles. The Aff-BKT models are trained from *both* the skill-correctness session data described in Section 4.4.1 *and* the affective session data, depicted as red circles. Both models were trained via the EM algorithm, with a single participant's data "held-out" for testing, resulting in a set of learned model parameters. The BKT model is evaluated on the 'held-out' participant's skill-correctness data (depicted as green circles), producing a single 'likelihood estimate' data point (shown here as a large blue circle). The Aff-BKT model undergoes one additional step, in which a subset of the trained Aff-BKT model parameters are used to construct a new model, $\hat{\theta}_{aff}$, that is structurally identical to the BKT model. Then, $\hat{\theta}_{aff}$ is evaluated in the same way as the BKT model: using the *same* held-out participant's skill-correctness data to produce a 'likelihood estimate' data point. This process is repeated once for each participant, thus the final result is a set of 38 'likelihood estimate' data points for each (Aff-BKT and BKT) skill model.

4.6 Evaluation Methodology

Having discussed the central issues and relevant mathematics of model evaluation in the previous section, I now describe the *actual* procedures used to evaluate the models in this thesis. At this point, I strongly encourage the reader to examine Figure 13, which depicts the model training and evaluation process graphically (with some mathematical notation as well), and which I will frequently reference for clarity. First, however, I introduce the some mathematical notation for reference in the following discussion.

4.6.1 Model evaluation notation

In this section I introduce the relevant mathematical notation for understanding the model training and evaluation procedure.

Let D^i_{bkt} denote the set of training data for a traditional BKT model of skill i (only the relevant skill-correctness data from all sessions, depicted as yellow circles in Fig. 13) and let D^i_{aff} denote the set of training data for an Aff-BKT model of skill i (the affective data plus the relevant skill-correctness data from all sessions, depicted as red circles in Fig. 13). These two sets compose the full training sets for the BKT and Aff-BKT models throughout this thesis.

Then, let θ_{bkt} (informally referred to as the 'BKT model' and depicted graphically at the top-center of Fig. 13) denote the set of BKT parameters trained from D^i_{bkt}, and let θ_{aff} denote the set of Aff-BKT parameters (informally referred to as the 'Aff-BKT model' and depicted graphically at the bottom-center of Fig. 13) trained from D^i_{aff}.

In order to compare the likelihoods of the two models, I introduce a new model, called Aff-BKT-Subset and denoted $\hat{\theta}_{aff}$. This model is formed from the subset of the θ_{aff} parameters listed in Table 2 (i.e., the BKT parameter subset of the Aff-BKT model.) Notably, these parameters are trained from D^i_{aff}, that is, they reflect the additional information of the affective training data. But Aff-BKT-Subset also has the same *structure* as θ_{bkt}, the BKT skill model. Thus, Aff-BKT-Subset provides a compromise for evaluation: the model's parameters are *trained* from both affective and skill-correctness data, but can be fairly *evaluated* on the same data as θ_{bkt}, the skill-correctness data. $\hat{\theta}_{aff}$ is depicted graphically as the red nodes of the model in the bottom right section of Fig. 13.

4.6.2 Model evaluation procedure

Most of the model selection techniques discussed above are used to compare models with different *structures*, trained on the same *data*. We rejected those techniques because θ_{aff} and θ_{bkt} differ both in structure *and* training data. By introducing $\hat{\theta}_{aff}$ (the Aff-BKT-Subset model), which has the same *structure* as θ_{bkt}, though trained from different *data*, we have reduced the difficulty of model comparison to a more tractable situation. Comparing $\hat{\theta}_{aff}$ and θ_{bkt} enables us to conduct a more straightforward analysis of the models' respective likelihoods: Leave-one-out cross-validation (LOOCV), a process in which all but one of the participants' data is used to train the model, then the likelihood of each trained model is evaluated, with respect to the held-out participant's data.

LOOCV occurs in two phases: during the "training phase", a single participant's data is held-out and the two models are trained according to the procedure described in Sec. 4.4 (Expectation Maximization). Then, during the "testing phase", the probability of

the held-out data, under the trained model is calculated (separately, for each model) as an estimate of the model's overall performance. The probability of held-out data is often used a measure of model 'fit' – that is, how well the model explains the observed data. Models with higher fit are, naturally, preferred.

The process described above (hold out a participant, train model, test on held-out participant's data) is known as a *fold* and results in a single data point per model, representing the model's 'fit' to the test data (depicted as a large blue circle in the center-right and bottom-right of Fig. 13). During a complete LOOCV evaluation, this process is then repeated for each of the participants, so that at the end of a complete LOOCV analysis with n participants, n different training sessions have occurred (each differing slightly, due to slight variation in training data), and n different test data points (the probabilities of all n different held-out participant's data or, equivalently, n different model likelihoods) have been collected per model.

Mathematically speaking, during each fold, we hold out a single participant's *skill-correctness* data (denoted $D_{test}^{i,k}$, the test data representing skill i, during fold k), shown as green circles in Fig. 13. Then each model is trained on the data from the remaining participants: θ_{aff} is trained on the affective *and* skill-correctness data from the training participants, D_{aff}^{i} – $\mathrm{D}_{test}^{i,k}$ (shown as small red circles in Fig. 13), while θ_{bkt} is trained on *just* the skill-correctness data from the training participants D_{bkt}^{i} – $\mathrm{D}_{test}^{i,k}$ (shown as yellow circles in Fig. 13). During the test phase, we construct the Aff-BKT-Subset model, $\hat{\theta}_{aff}$, by removing the extra nodes and parameters from θ_{aff}. Finally, we compute the probabilities of the test data under each model (or, equivalently, the likelihood of each model given the data): $\Pr(D_{test}^{i,k}|\hat{\theta}_{aff})$ and $\Pr(D_{test}^{i,k}|\theta_{bkt})$ – shown as a large blue circle in Fig 13.

The key here is that both models are being evaluated on *identical* test data! Because these model likelihood data points[9] are derived from structurally identical models ($\hat{\theta}_{aff}$ and θ_{bkt}) under the *same* data, after each fold, the two model likelihoods can be directly compared. The model with the higher likelihood better 'fits' the data, and is more likely to be able to generalize to new data.

Data from 38 participants was used. Thus, we repeated the 'fold' process 38 times, holding-out one participant each fold, and computed 38 model likelihoods per skill model. Figure 13 shows the complete evaluation pipeline - from training data, to model construction, to model evaluation. Full results by skill and model type are presented in Figures 14 and 15. Aggregate statistics for each skill and model type are presented in Table 5.

4.7 Results

Figure 14 show the complete evaluation data: the log-likelihood of each skill model, evaluated on every participant. For three of the four skills (Exact-Correct, First-Letter, and Last-Letter) the Aff-BKT-Subset model error is lower for nearly all participant test data. For Length, the Aff-BKT-Subset model error is generally lower, but there were some cases in which the BKT model had better fit. Overall, however, it is clear that the Aff-BKT-Subset model fits the data much better than the BKT model. Table 5 reinforces these results: every Aff-BKT-Subset skill model has higher mean likelihood than the traditional BKT model for the same skill. Furthermore, every

[9]for practical reasons, computed as and often presented as, log-likelihoods

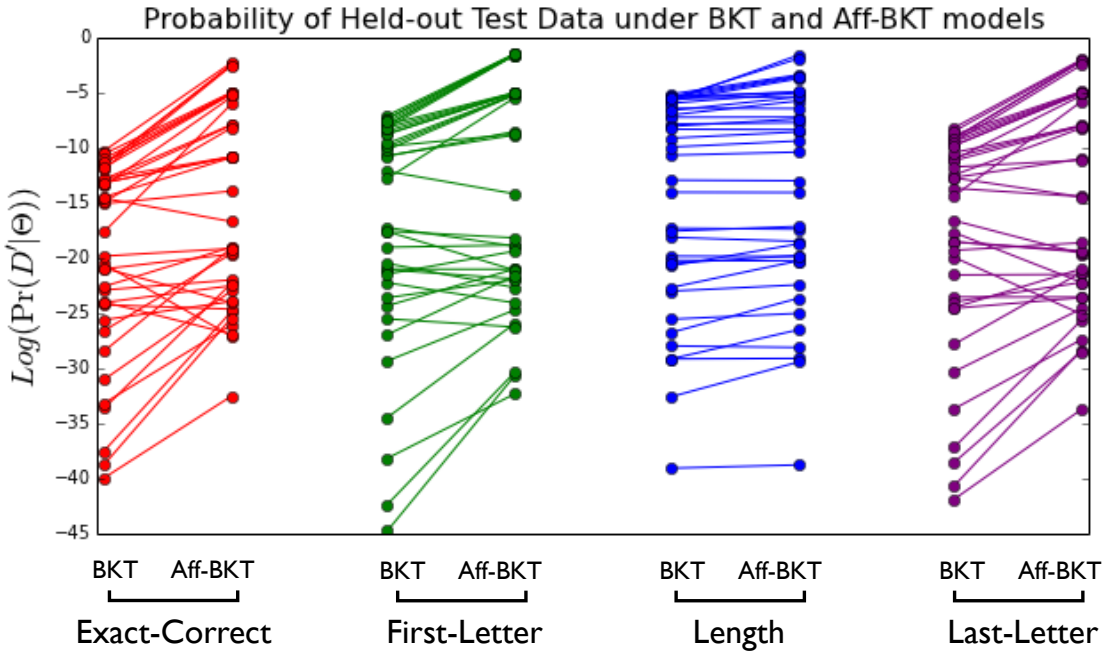


Figure 14: Probability of Test Data under BKT and Aff-BKT models, color-coded by skill. Within each skill, data points to the left represent the log-likelihood (model fit) of the BKT model evaluated on one test participant's data. Each point is connected to the log-likelihood of the Aff-BKT-Subset model evaluated on the same participant's data.

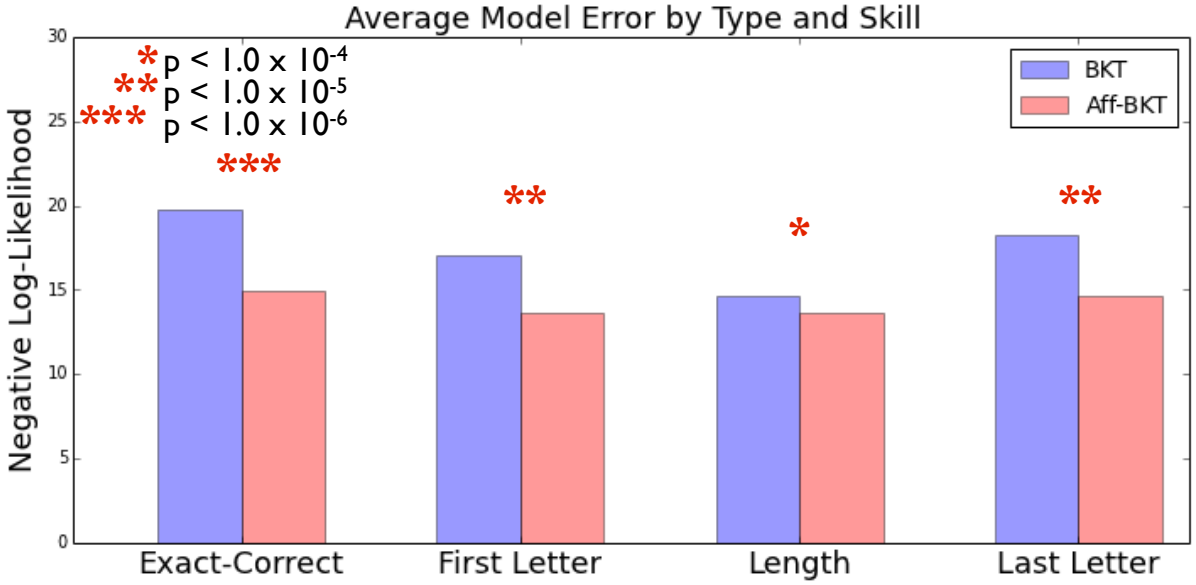


Figure 15: Mean error estimates of BKT and Aff-BKT models by skill. Significance was calculated via a one-sample T-test on the *difference* between model error estimates, evaluated on the same test participant data: $|\Pr(D^{i,k}_{test}|\ \hat{\theta}_{aff}) - \Pr(D^{i,k}_{test}|\theta_{bkt})|$

Aff-BKT-Subset model has a higher maximum and minimum likelihood than the BKT model for the same skill.

Figure 15 shows the statistical significance of these results, presented as a decrease in model error, rather than an increase in 'fit'. Following [Yudelson et al. (2013)], we use the

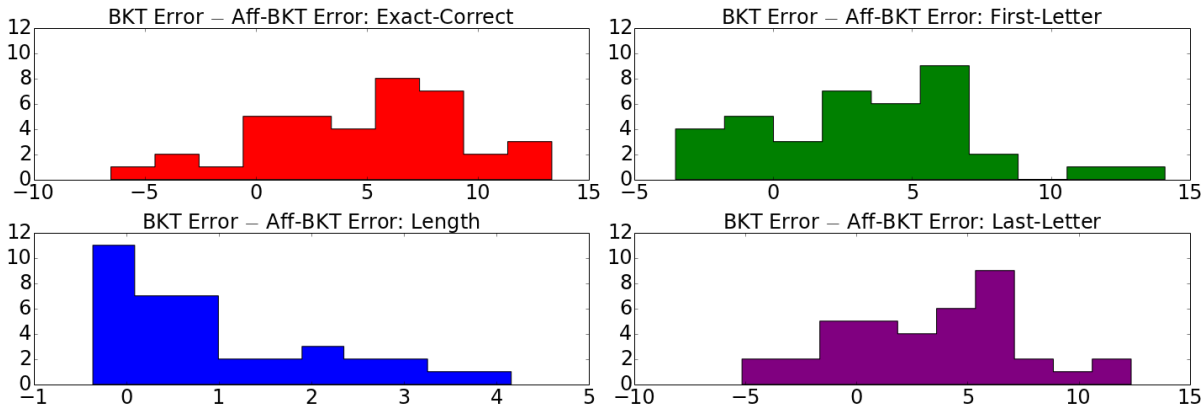


Figure 16: Histograms showing the distribution of error *difference* between BKT and Aff-BKT models on the same test data. Error distributions for 3 of 4 skills were normally distributed.

Skill	Model	Mean Log-Likelihood	Std Deviation	Max	Min
Exact-Correct	A-BKT	-14.923	9.01	-2.2964	-32.545
Exact-Correct	BKT	-19.79	10.386	-10.304	-39.979
First-Letter	A-BKT	-13.652	10.091	-1.4883	-32.343
First-Letter	BKT	-17.090	10.386	-7.3040	-44.777
Length	A-BKT	-13.6160	9.596	-1.62	-38.7923
Length	BKT	-14.5992	9.513	-5.160	-39.094
Last-Letter	A-BKT	-14.644	9.493	-2.022	-33.753
Last-Letter	BKT	-18.222	9.98	-8.202	-41.915

Table 5: T-test results by metric, comparing differences in mean interaction value across Robot and Tablet conditions

negative log-likelihood as an error function for model evaluation (for ease of presentation). This definition of error follows naturally from the interpretation of model likelihood as a measure of 'fit' (see Sec. 4.6.2). Figure 15 shows the mean error (that is, *average negative log-likelihood*) of each skill model.

Because the BKT and Aff-BKT-Subset model errors are derived from the same set of test data, I compared the *difference* between the model errors, for each piece of test data, in order to determine if the BKT model error was significantly higher than the Aff-BKT-Subset model error. Figure 16 shows histograms of the error difference by skill. The majority of each distribution lies above 0 (indicating that the BKT model has higher error overall), though parts of the each distribution lie below 0, just as the BKT model may outperform the Aff-BKT model on some individual pieces of test data.

The distribution of error difference for Exact-Correct, First-Letter, and Last-Letter appear normal, while the the distribution of error differences for Length does not. This was confirmed by a Shapiro-Wilk test on each distribution. To calculate the statistical significance of error differences (shown in Fig. 14), I conducted a one-sample T-test on the Exact-Correct, First-Letter, and Last-Lettererror differences,

and a one-sample Wilcoxon signed-rank test on the LENGTH error difference. For every skill, the BKT model had significantly higher mean error ($p < .0001$).

These results answer our second research question – "Can we build models that use affective data to improve the performance of traditional Knowledge Assessment models?" – affirmatively. Interestingly, we were able to reach this conclusion through comparisons between the BKT model and the Aff-BKT-Subset model, a model that is structurally identical to the traditional BKT model and hence *does not explicitly model affect*. Rather, the influence of the affective data manifests itself in the trained values of the traditional BKT parameters. In other words, by including the additional parameters, structure, and data to support affect-awareness, we found that the Aff-BKT model learns better parameters *even for parts of the model that have nothing to do with affect*! Without further study, the underlying computational reasons for this cannot be conclusively determined. However, I hypothesize that the affective data helps "explain away" some of the observed variance in Correct/Incorrect answers. Essentially, the BKT model tries to fit all five of its parameters to explain the *full* variance in the skill-correctness data. However, modern theories of affect and learning suggest that the two are deeply related, hence some variance in the skill-correctness data can likely be explained by affective factors. The additional affective parameters of the Aff-BKT model can adjust to the variance due to affective variables (e.g., distraction), leaving the Aff-BKT-Subset parameters free to better fit to the variance due to *knowledge-based* factors (e.g., genuine skill mastery).

5 Contributions and Conclusions

In this thesis we have answered both research questions from Chapter 1 affirmatively. We have shown that children are **more emotionally expressive when engaged in an interactive educational task with a social robot than when engaged in an *identical* task with a tablet alone**. Furthermore, we have demonstrated that these kinds of emotional expressions can be successfully integrated into an inferential model for assessing children's knowledge states. **These affective models outperform traditional approaches to the Knowledge Assessment problem, demonstrating the utility of sensing affective data and constructing tutoring models to make use that data.** Together, these results suggest that physical, social robots may be a more appropriate medium for developing affect-aware computational tutors. Previous research has shown that the mere physical presence of social robots can alter important interaction dynamics: this thesis adds to that body of work by identifying a particularly important aspect of that phenomenon (children's increased emotional expressivity) and showing how this shift in interaction style can lead to improved model performance.

Research agencies are beginning to recognize the potential of affect-aware robotic tutors: the research described in this thesis is supported by a National Science Foundation "Expedition in Computing" grant, to develop Socially Assistive Robots, robots capable of improving children's health, education, and well-being through social interaction. The European Union recently launched the EMOTE project: a multi-year, multi-institution effort under the FP-7 program to develop "embodied perceptive tutors for empathy-based learning." Clearly, researchers on a global scale feel the time is ripe to investigate the potential of affect-aware robot tutors to significantly improve the scope and quality of computational tutoring systems. This thesis elucidates some of the potential improve-

ments physical robots have over software-only systems, by demonstrating social robots' ability to induce higher degrees of emotional expressivity and adapting and improving traditional ITS inference techniques to the novel paradigm of human-robot tutoring interactions.

6 Future Work

This thesis describes work that supports the idea that physically embodied, affect-aware, social robot tutors have the potential to provide more effective, empathic, and scalable educational experiences. We have shown that even simple model augmentations that integrate affect into inference can improve the performance of modern methods. Naturally, however, this is only the beginning, and there remain many exciting research challenges we hope to address, based on the results of this thesis. In the future, we could determine which affective features are most helpful in improving model fit, by repeating the analysis of Section 4.3 with different combinations of affective features. We could also explore more complex affective models. In this thesis, affective variables were treated simply as new, independent observations, but future work could explore hierarchical models of affect. In addition, the BKT and Aff-BKT models were trained from data gathered from the entire population of participants. Future work could explore developing personalized models of student affect, and perhaps could cluster students as having similar or different emotional learning profiles. Finally, the work described in this thesis is purely inferential. That is, it demonstrates that robot tutors can use affective information to more accurately model the state of the world. It does not, however, address the problem of action: given the state of the world, what actions should the tutor take to best help the child? Developing action policies for educational interaction is a challenging multi-objective problem. For instance, an educational tutor may have to balance short-term goals (such as keeping the student engaged or interested) and longer-term goals (such as mastery of educational material) as well as trade-offs between computational goals (such as gathering information that could improve its internal model of the student) and learning-based objectives (introducing content that the model is relatively sure will help the student). Action models in education are an active topic of research, though little work has examined how affective information could be used to shape these policies.

Researchers in psychology and cognitive science are coming to understand that affect, far from being an 'irrational' influence, is actually crucial to everyday decision making and judgements. This understand is beginning to influence the design of agents and algorithms that model or simulate intelligent behavior and decision making. While a relatively recent effort, many researchers are now pursuing efforts to integrate affect into intelligent systems, and, encouraged by the results of this thesis, we believe that this research will ultimately lead to more effective, intelligent, and enjoyable interactions with technology.

References

I. Arroyo, C. Beal, T. Murray, R. Walles, and B. P. Woolf. Web-based intelligent multimedia tutoring for high stakes achievement tests. In *Intelligent Tutoring Systems*, pages 468–477. Springer, 2004.

H. Aviezer, Y. Trope, and A. Todorov. Body cues, not facial expressions, discriminate between intense positive and negative emotions. *Science*, 338(6111):1225–1229, 2012.

W. A. Bainbridge, J. W. Hart, E. S. Kim, and B. Scassellati. The benefits of interactions with physically present robots over video-displayed agents. *International Journal of Social Robotics*, 3(1):41–52, 2011.

R. S. Baker, A. T. Corbett, and V. Aleven. More accurate student modeling through contextual estimation of slip and guess probabilities in bayesian knowledge tracing. In *Intelligent Tutoring Systems*, pages 406–415. Springer, 2008.

T. Belpaeme, P. E. Baxter, R. Read, R. Wood, H. Cuayáhuitl, B. Kiefer, S. Racioppa, I. Kruijff-Korbayová, G. Athanasopoulos, V. Enescu, et al. Multimodal child-robot interaction: Building social bonds. *Journal of Human-Robot Interaction*, 1(2):33–53, 2012.

N. Bilton. The child, the tablet, and the developing mind, March 2013. [Online; Accessed March 25, 2015].

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

L. N. Brown and A. M. Howard. The positive effects of verbal encouragement in mathematics education using a social robot. In *Integrated STEM Education Conference (ISEC), 2014 IEEE*, pages 1–5. IEEE, 2014.

W. Burleson. *Affective learning companions: strategies for empathetic agents with real-time multimodal affective sensing to foster meta-cognitive and meta-affective approaches to learning, motivation, and perseverance*. PhD thesis, Massachusetts Institute of Technology, 2006.

B. Byrne and R. Fielding-Barnsley. Phonemic awareness and letter knowledge in the child's acquisition of the alphabetic principle. *Journal of Educational Psychology*, 81 (3):313, 1989.

B. J. Byrne. *The foundation of literacy: The child's acquisition of the alphabetic principle*. Psychology Press, 1998.

M. T. Chi. Active-constructive-interactive: A conceptual framework for differentiating learning activities. *Topics in Cognitive Science*, 1(1):73–105, 2009.

A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278, 1994.

A. T. Corbett and J. R. Anderson. Locus of feedback control in computer-based tutoring: Impact on learning rate, achievement and attitudes. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 245–252. ACM, 2001.

C. H. Crouch and E. Mazur. Peer instruction: Ten years of experience and results. *American Journal of Physics*, 69(9):970–977, 2001.

M. Csikszentmihalyi. Flow and education. *NAMTA journal*, 22(2):2–35, 1997.

M. Csikszentmihalyi and I. S. Csikszentmihalyi. *Optimal experience: Psychological studies of flow in consciousness*. Cambridge University Press, 1992.

S. K. D'Mello, B. Lehman, and A. Graesser. A motivationally supportive affect-sensitive autotutor. In *New perspectives on affect and learning technologies*, pages 113–126. Springer, 2011.

R. El Kaliouby and P. Robinson. Real-time inference of complex mental states from facial expressions and head gestures. In *Real-time vision for human-computer interaction*, pages 181–200. Springer, 2005.

K. Ferguson, I. Arroyo, S. Mahadevan, B. Woolf, and A. Barto. Improving intelligent tutoring systems: Using expectation maximization to learn student skill levels. In *Intelligent Tutoring Systems*, pages 453–462. Springer, 2006.

G. Gordon and C. Breazeal. Bayesian active learning-based robot tutor for children?s word-reading skills. In *Proceedings of the Twenty-Ninth Conference on Artificial Intelligence (AAAI-15)*, 2015.

G. Gordon, C. Breazeal, and S. Engel. Can children catch curiosity from a social robot? In *Human-Robot Interaction (HRI), 2015 10th ACM/IEEE International Conference on*, 2015.

A. C. Graesser, P. Chipman, B. C. Haynes, and A. Olney. Autotutor: An intelligent tutoring system with mixed-initiative dialogue. *Education, IEEE Transactions on*, 48 (4):612–618, 2005.

I. Howley, T. Kanda, K. Hayashi, and C. Rosé. Effects of social presence and social role on help-seeking and learning. In *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, pages 415–422. ACM, 2014.

E. T. Jaynes. *Probability theory: the logic of science*. Cambridge university press, 2003.

M. Keppell, E. Au, A. Ma, and C. Chan. Peer learning and learning-oriented assessment in technology-enhanced environments. *Assessment & Evaluation in Higher Education*, 31(4):453–464, 2006.

C. D. Kidd and C. Breazeal. Robots at home: Understanding long-term human-robot interaction. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 3230–3235. IEEE, 2008.

E. S. Kim, L. D. Berkovits, E. P. Bernier, D. Leyzberg, F. Shic, R. Paul, and B. Scassellati. Social robots as embedded reinforcers of social behavior in children with autism. *Journal of autism and developmental disorders*, 43(5):1038–1049, 2013.

J. Kory and C. Breazeal. Storytelling with robots: Learning companions for preschool children's language development. In *Robot and Human Interactive Communication, 2014 RO-MAN: The 23rd IEEE International Symposium on*, pages 643–648. IEEE, 2014.

I. Leite, G. Castellano, A. Pereira, C. Martinho, and A. Paiva. Empathic robots for long-term interaction. *International Journal of Social Robotics*, 6(3):329–341, 2014.

D. Leyzberg, S. Spaulding, M. Toneva, and B. Scassellati. The physical presence of a robot tutor increases cognitive learning gains. In *Proceedings of the 34th Annual Conference of the Cognitive Science Society. Austin, TX: Cognitive Science Society*, 2012.

D. Leyzberg, S. Spaulding, and B. Scassellati. Personalizing robot tutors to individuals' learning differences. In *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, pages 423–430. ACM, 2014.

I. Y. Liberman, D. Shankweiler, and A. M. Liberman. The alphabetic principle and learning to read. 1989.

R. A. Mead, S. L. Thomas, and J. B. Weinberg. *From grade school to grad school: an integrated STEM pipeline model through robotics*. Information Science Reference, 2012.

K. Murphy et al. The bayes net toolbox for matlab. *Computing science and statistics*, 33(2):1024–1034, 2001.

U. of Oregon Center for Teaching and Learning. Big ideas in beginning reading: Alphabetic principle, April 2015. URL `http://reading.uoregon.edu/`. [Online; Accessed April 22, 2015].

S. Papert. *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc., 1980.

R. W. Picard. *Affective computing*. MIT press, 2000.

M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5, 2009.

C. Reynolds and R. Picard. Affective sensors, privacy, and ethical contracts. In *CHI'04 Extended Abstracts on Human Factors in Computing Systems*, pages 1103–1106. ACM, 2004.

B. Scassellati, H. Admoni, and M. Mataric. Robots for use in autism research. *Annual Review of Biomedical Engineering*, 14:275–294, 2012.

E. Short, K. Swift-Spong, J. Greczek, A. Ramachandran, A. Litoiu, E. C. Grigore, D. Feil-Seifer, S. Shuster, J. J. Lee, S. Huang, et al. How to train your dragonbot: Socially assistive robots for teaching children about nutrition through play. In *IEEE International Workshop on Robot and Human Interactive Communication (RO-MAN), 2014*. IEEE, 2014.

R. Slavin. Cooperative learning. *Learning and Cognition in Education Elsevier Academic Press, Boston,*, pages 160–166, 2011.

D. Szafir and B. Mutlu. Pay attention!: designing adaptive agents that monitor and improve user engagement. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 11–20. ACM, 2012.

F. Tanaka and S. Matsuzoe. Children teach a care-receiving robot to promote their learning: Field experiments in a classroom for vocabulary learning. *Journal of Human-Robot Interaction*, 1(1), 2012.

J. K. Torgesen, R. Wagner, and C. Rashotte. Towre–2 test of word reading efficiency. *Austin, TX: Pro-Ed*, 1999.

B. van De Sande. Properties of the bayesian knowledge tracing model. *JEDM-Journal of Educational Data Mining*, 5(2):1–10, 2013.

K. VanLehn. Intelligent tutoring systems for continuous, embedded assessment. *The future of assessment: Shaping teaching and learning*, pages 113–138, 2008.

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

K. VanLehn, W. Burleson, M.-E. C. Echeagaray, R. Christopherson, J. G. Sanchez, J. Hastings, Y. H. Pontet, and L. Zhang. The affective meta-tutoring project: How to motivate students to use effective meta-cognitive strategies. In *19th International Conference on Computers in Education, Chiang Mai, Thailand*, 2011.

B. Woolf, W. Burleson, I. Arroyo, T. Dragon, D. Cooper, and R. Picard. Affect-aware tutors: recognising and responding to student affect. *International Journal of Learning Technology*, 4(3):129–164, 2009.

Y. Xu, K.-m. Chang, Y. Yuan, and J. Mostow. Eeg helps knowledge tracing. In *Proceedings of the 12th International Conference on Intelligent Tutoring Systems Workshop on Utilizing EEG Input in Intelligent Tutoring Systems*, 2014.

M. V. Yudelson, K. R. Koedinger, and G. J. Gordon. Individualized bayesian knowledge tracing models. In *Artificial Intelligence in Education*, pages 171–180. Springer, 2013.