Abstract of "Query Strategies for Directed Graphical Models and their Application to Adaptive Testing" by Sam Saarinen, Ph.D., Brown University, May 2021.

Educational assessments are crucial for both instructors and education researchers to measure learning, troubleshoot student problems, evaluate pedagogy, and improve education. Unfortunately, creating and administering reliable assessments is a labor-intensive process. This dissertation frames assessment creation from a pool of assessment items as a machine learning problem and tackles this problem by learning directed graphical models of topic prerequisite relationships. It is shown on a variety of real datasets that these models can be learned in a computationally- and data- efficient manner from records of student responses, can be queried efficiently, and produce accurate predictions about student knowledge. This technique is used to develop and administer novel computer science assessments of instructor-defined specificity using student-authored questions.

Query Strategies for Directed Graphical Models and their Application to Adaptive Testing

by
Sam Saarinen
B. Sc., University of Kentucky, 2016
Sc. M., Brown University, 2018

A dissertation submitted in partial fulfillment of the
requirements for the Degree of Doctor of Philosophy
in the Department of Computer Science at Brown University

Providence, Rhode Island
May 2021

This dissertation by Sam Saarinen is accepted in its present form by
the Department of Computer Science as satisfying the dissertation requirement
for the degree of Doctor of Philosophy.

Date _____ _____

Michael L. Littman, Director

Recommended to the Graduate Council

Date _____ _____

Stephen Bach, Reader

Date _____ _____

Shriram Krishnamurthi, Reader

Approved by the Graduate Council

Date _____ _____

Andrew G. Campbell
Dean of the Graduate School

Vita

Sam Saarinen received a Bachelor's of Science degree (with majors in Computer Science and Mathematics, and a minor in Physics) from the University of Kentucky in 2016. He received a Master's of Science degree (in Computer Science) from Brown University in 2018. He is a recipient of the Goldwater Scholarship (2014), a finalist for the Computing Research Association Outstanding Undergraduate Researcher award (2016), and an honorable mention for the National Science Foundation Graduate Research Fellowship Program (2016).

While at Brown, Sam Saarinen lectured parts of Machine Learning (CSCI 1420) and Introduction to Discrete Structures and Probability (CSCI 0220). He also served as the Faculty-Graduate Liaison in the Computer Science Department from Fall 2019 to Spring 2021. While at Brown, he published two papers at conference (Saarinen et al., 2019, 2020) as well as two workshop papers with mentored students (Honda et al., 2019; Lin et al., 2020). Work from this dissertation is included in four additional publications in submission.

Preface: Why did I write this?

Why have I spent the last 5 years working on this? Why should I do anything at all? While I have a slim chance of persuading you, my reader, of my moral preconceptions in only a few paragraphs, I hope I can give you a lens that brings the work in this dissertation into focus. I hope that you will sympathize with my view both that the problem I have been working on is both profoundly important and that significant progress can be made in a lifetime. The rest of this dissertation will hopefully convince you that I have made some significant progress on this problem already. While the rest of the dissertation will rely on a much more technical treatment, I hope you will bear with a few paragraphs of merely evocative language as I strive to situate this work in the context of a vast universe.

If evolution can be said to produce an end result, I claim it is sustainability. I could appeal to natural ecology as justification, but this is really a mathematical observation. Even in a chaotic and expansive ecosystem, transient phenomena are doomed to be replaced by sustainable systems in the same way that a Markov chain converges to a stationary distribution. While we hope those stationary distributions will be non-trivial (for example, not merely the proposed eventual heat-death of the universe), a universe in which a more sustainable system is possible is one in which that system is statistically inevitable.

In view of this belief, I elect to be a pragmatic optimist: I choose to believe that there is a highly sustainable order of the world (and the universe) that is much better than our current system, and I also believe that we have a good deal of influence over when it arrives.

I believe that there is a future world where there is less suffering, more wisdom, and a place for everyone. I think that world has no war, few misunderstandings, and flourishing science. I think that world has a responsible stewardship of the ecosystem and climate of the planet, a proliferation of new kinds of art, and a great capacity for compassion.

Personally, I think there is something beautiful about participating in the inevitable — to be a small cog on which the clockwork of the universe turns. So how can I usher in this bright future? One thing that I know for certain is that I cannot do it alone. I could simply recruit others to the cause, but because of our mortality, we will always be recruiting others to the cause. Humanity suffers under a great burden of generational amnesia — knowledge has to be painstakingly transferred to children who can't remember what came before. So any attempt to bring about this bright future that takes more than a generation must inevitably contend with the difficulty of education.

Most experienced educators believe that high-quality education is possible. Some are even arrogant enough to believe that they can provide it. (By high-quality, I roughly mean that the student learns as much as they possibly can in a given amount of time.) I think this belief is sound if we talk about educating a single person. Truly great teachers are able to provide education at that same level of quality to dozens, or sometimes even hundreds of students per year. But what about thousands of students? Millions of students? What if we could provide education (at nearly the same quality as the best one-on-one instruction) to billions of students? Current educational models don't allow great education to scale. The quality of education available to you is largely a function of where you are born and to whom.

I don't think that it's possible for a single person to double the amount of climate science produced per year, or to double the amount of energy research, cultural tolerance, or economic development in the world. But I do think that a single person, working very diligently, could create mechanisms that make education more scalable, potentially doubling the number of climate scientists.

What I might hesitate to say in the more rigorous work that follows, I will say quite openly here: Education research is a field with many ideas and weak science. Measurement is a prerequisite of science, and measurement is expensive, unreliable, and ill-defined in education, where the object of interest (student knowledge) cannot be directly observed. There are some outstanding works that have succeeded despite these limitations. But imagine how many more there might be if measuring student knowledge were cheap, easy, and precise. If assessment were easy, we could begin to answer fundamental educational questions, like "What is the best way of teaching that topic to this student right now?" If these questions could be answered easily and systematically, we could develop scalable approaches to education around what we know to be effective.

The goal of this dissertation is to describe a step forward on the grand journey to make education research more scientific, educational scalability greater, and that bright future sooner.

Contents

List of Figures

Chapter 1

Introduction

This dissertation documents a body of work involving the application of computational methods to the creation and administration of educational assessments (tests, quizzes, etc.)[1]. Education can be a somewhat subjective field, so this introduction will provide context on the qualitative objectives associated with assessment creation in prior work, and the thesis statement of this dissertation will be elaborated. In the following Chapter, related work in assessment generation and knowledge modeling will be summarized.

The main scientific contributions of the author begin in with a quantitative operationalization of an assessment creation objective. Arguments for the utility of prerequisite maps (a directed graphical structure modeling propaedeutic relationships between topics) are presented in chapter 4, followed by chapters detailing how these structures can be learned (chapter 5), queried (chapter 6), and applied to generating real assessments (chapter 7). Finally, limitations and opportunities for future work will be described in chapter 8.

1.1 Why Education?

Education is an enormous industry, a useful tool for scientific and technological development, a vehicle for increasing understanding and tolerance between cultures, and arguably fundamental to the human experience. But it is also an area ripe with scientific questions. Cognitive Science, Psychology, Artificial Intelligence, and Education Research are all interested to some degree in modeling human learning. When do students learn? How do they learn? Why do they learn? While Cognitive Science, Psychology, and Artificial Intelligence all have a role to play in uncovering the mechanisms of human learning, Education Research is more concerned with predicting the phenomenon of learning, itself. This gives it a unique role and perspective in the scientific medley.

[1] Note that this is a static copy of this document that may not reflect any changes or corrections made after release. A "living" version is linked from the author's website at `SamSaarinen.com/artifacts`

1.2 Why Machine Learning?

Most existing educational research has inherited the statistical and experimental methods of Psychology. So why machine learning? Machine learning methods are ideally suited for modeling black-box functions, processes, or phenomena (that is, processes that are not understood in detail). For example, machine learning has seen great success in predicting complex interactions, modeling the difference between images of different kinds of objects, and identifying explanatory patterns in data . The downside relative to smaller-parameter modeling approaches is that these algorithms require large amounts of data. This is a good fit to modeling educational problems, where precise models are lacking but there is a surfeit of data available.

1.3 Context: Assessment

An ***assessment instrument*** is a tool for determining what a student knows (conceptually or practically) and is typically composed of several ***assessment items***, which often take the form of a prompt or question and a rule for evaluating the response given by the student. Although the formality of the term "assessment instrument" typically evokes a series of forced-choice questions, assessment items can also be short-answer free-response questions, tasks graded by rubric, etc.

Generally speaking, the purpose of assessments is to estimate what students know or can do in relation to a particular topic. This information can be used to answer a variety of questions like:

- What does the student know now?
- What should the student learn next?
- How much has the student learned?
- Is technique X or technique Y a better way of teaching topic Z?

Great assessments can also allow more specific questions to be answered, like:

- How does student A think about topic Z? (In other words, what is the student's conception of the topic?)
- Is technique X or technique Y a better way of teaching topic Z *to student A*?

These latter questions have to do with generating insight into student ***mental models***, that is, the internal representation that a student uses to reason about a concept. While mental models cannot be observed directly, a pattern of responses consistent with a particular mental model suggests that that model may be a useful starting place for instruction. Partially-incorrect mental models, or ***misconceptions***, are mental models that allow students to answer some, but not all questions correctly. Misconceptions are often difficult to detect, but contribute to the phenomenon of ***fragile student knowledge*** (Perkins and Martin, 1986) in which a student appears to understand a concept, but is only able to answer very particular questions about it.

1.4 Qualitative Objectives for assessment

While there are many possible assessments, not all of them will be equally useful in answering the questions in the previous section. In particular, we are interested in assessments that are ***valid***, ***reliable***, ***efficient***, ***interpretable***, ***actionable***, and, finally, ***inexpensive***. These criteria are elaborated following.

Valid and Reliable: Assessments should be useful as measurements. Two criteria discussed in the assessment literature are ***validity*** and ***reliability***. Although there are a variety of definitions for these two terms (Hammersley, 1987), validity generally refers to whether the assessment measures what we want to measure, and reliability refers to how little variance there is in that measurement. (Validity and reliability loosely correspond to accuracy and precision in the physical sciences.) Validity for new assessments is typically established qualitatively through a combination of student introspection, expert review, and repeated trials, although it can be established quantitatively in the case that a validated benchmark assessment is available. Reliability, on the other hand, can generally be measured or estimated statistically.

Efficient: Student time is valuable. There is an opportunity cost to using student time on assessments, so, all else being equal regarding the other criteria, assessments that take less time are better. Instructor time is also valuable, so assessments that take less instructor time to administer and to grade are also preferable.

Interpretable: High-quality assessments provide insights not only into what students know, but how they think about a topic. ***Concept Inventories***, such as the well known Force Concept Inventory (Hestenes et al., 1992), associate each possible response on a forced-choice assessment item with a distinct mental model. The responses can suggest to the instructor not just what the students are thinking (through the answer), but why (through the association with a mental model). Assessments should also help the instructor perceive the conceptual relationships between questions.

Actionable: Good assessments should facilitate effective teaching. Assessments that produce a cumulative score but that do not pinpoint topics a student should focus on have low actionability. Interpretability and actionability often coincide, although they are different objectives. An instrument like a concept inventory is actionable largely due to its interpretability; responses that suggest a misconception also suggest that pedagogy that corrects the misconception would be effective. Actionable assessment is critical to several of the researched educational interventions with the largest effect sizes, such as individualized instruction (Bloom, 1984; Corbett, 2001) and mastery-based learning (Guskey, 2007).

Aggregate data across students in a classroom could also be used to provide the following actionable recommendations:

- Address Bottleneck Concepts: The term ***bottleneck concepts*** is introduced here to refer to concepts that cause a sharp fall-off in student mastery, and are thus excellent candidates for instructors to spend additional time addressing. ("Bottleneck concepts" are related to, but less specific than, the idea of "threshold concepts" (Meyer and Land, 2003).)
- Pair Complementary Students: A student who has mastered A but not B could be paired with one who has mastered B but not A, each teaching the missing concept to the other. Pairwise work often builds peer camaraderie and increases learning gains without the need for direct instructor intervention (Crouch and Mazur, 2001; Porter et al., 2011; Rao and DiCarlo, 2000).
- Work with Targeted Groups: Small groups with shared misconceptions could be pulled-out for efficient instruction as a group.

Inexpensive: Assessments with the prior four criteria (efficient, reliable, interpretable, and actionable) are quite difficult to construct, and thus usually require significant amounts of expert time and effort (and thus expense) to produce (Taylor et al., 2014). This expense limits the rate at which new high-quality assessments can be generated. As elaborated in the immediately following subsection, we believe that developing computational tools that reduce the cost of generating high-quality assessments will lead to significant benefits, especially in emerging curricula for which minimal time and resources have been available to develop such assessments.

1.5 A Scalable Approach to Assessment Generation

This dissertation focuses on a particular machine learning problem (see chapter 3) that is motivated by a desire to generate assessments that meet the criteria outlined in the previous section. Current methods for creating high-quality assessments (discussed in greater length in chapter 2) are very expensive, due to their significant cost in time and expert labor. In order to reduce the cost and expert labor required to generate assessments, the novel process presented here leverages three scalable resources that are being underutilized in most contemporary educational contexts: students; computation; and data. The importance of each of these ingredients is sketched here. See Figure 1.1 for a diagrammatic overview of the solution.

Students Many universities have been inundated by increasing computer science enrollments (and many have faced hiring shortages). Rather than bemoaning the increasing student-teacher ratios, the large amount of human intelligence students bring to the table could be leveraged at scale. As elaborated in chapter 7, students are frequently able to write useful assessment items and to articulate their own answers and rationales. By spreading work that traditionally has been done by experts (such as assessment item design, answer/distractor generation, and interpretation of rationales) across a large number of students and leveraging some novel ***class-sourcing*** mechanisms (facilitated collection of student and instructor contributions, that is, *crowd-sourcing in a class*),

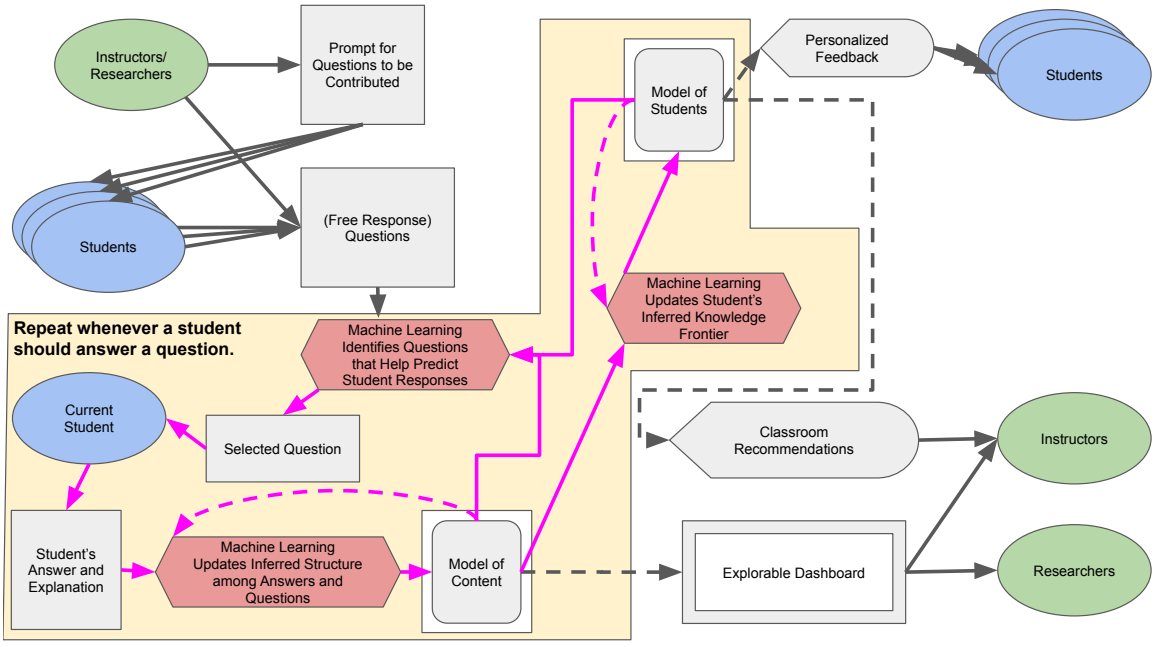

Figure 1.1: A flow diagram for a proposed assessment generation system. The inner loop (yellow background) is the subject of the research in this dissertation.

the cost in expert time to generate new assessments can be greatly reduced, perhaps creating novel pedagogical activities for students in the process.

Computation The ubiquity of computational devices can be leveraged to apply machine learning to some well-structured subproblems in the assessment-generation pipeline. For example, unsupervised or self-supervised learning can be used to discover structure among a pool of assessment items (see chapter 5). Supervised learning can be used to predict student responses to new assessment items (and measure uncertainty). And, reinforcement learning can be used to balance testing new assessment items with collecting data (see chapter 6). Thus, computation can regulate the class-sourcing processes, more effectively using student time and reducing the demands of the process on the time of experts.

Data There are many exciting ideas around using data to make assessments more efficient and reliable. For example, Computer Adaptive Testing (CAT) has a history dating back to 1985 Weiss (1985), bearing fruit in modern intelligent tutoring systems Corbett (2001). Statistical measures of reliability, such as Cronbach's alpha Cronbach (1951), have been around since 1951. But, all of these methods are applied after the generation and administration of assessment items on a test population, and lend themselves to an expensive batch process of iterating on an assessment.

By applying assessment data, such as answer-distributions and cross-question correlation statistics, *as the assessment is being generated*, the time and data required to generate high-quality assessments

can be greatly reduced.

1.5.1 Cui Bono?

Who benefits? There is benefit to several classes of people:

Students As elaborated in chapter 7 and illustrated by Figure 1.1, students engage with this process in a way that closely resembles existing quiz processes. The primary difference is that students are also asked to contribute the prompt for an assessment item (a form of contributing student pedagogy that may have value in itself (Luxton-Reilly and Denny, 2010; Hamer et al., 2008a)). At the end, however, students get personalized feedback comparable to what they might expect from a much less light-weight instrument.

Educators Educators can easily generate context-specific assessments, or administer existing assessments generated with this process. Either way, educators get clear insight into not only what students are thinking, but why.

Researchers To use this process to generate assessments, researchers only need to set the scope of the assessment (via prompts for the students to contribute assessment items and answers to other students' questions, respectively), and to engage a test population. This work acts as an enormous multiplier on researchers' efforts to create and administer context-relevant assessments.

The total time cost for each party is modest—much less than the cost for comparable results without using our process.

1.6 Thesis Statement

The main thesis of this document is this: Discrete graphical models of prerequisite relationships between assessment items can be learned from data, can be efficiently queried, and are useful for developing and administering novel computer science assessments of instructor-defined specificity using student-authored questions.

Discrete graphical models of prerequisite relationships between assessment items — We will call these ***Prerequisite Maps***. The prerequisite relationships between questions will be represented by an acyclic directed graph where nodes represent an equivalence class of questions, and a directed edge indicates that the parent (or source node) is prerequisite to the child (or destination node).

learned from data — In contrast to existing methods, which are error-prone and labor-intensive, the methods presented in chapter 5 allow these curricular dependency maps to be constructed automatically from empirical student responses. These curricular dependency maps can be used to achieve state-of-the-art predictive accuracy on unseen student responses.

efficiently queried — With very few exceptions, contemporary classroom practice is to have all students answer all questions on each assessment. This is highly inefficient, as many of the questions could be inferred to be too easy or too hard given the student's initial responses. By minimizing the

number of nodes queried (number of questions students answer) from a curricular dependency map, the cost of administering an assessment (in terms of student time and classroom time) can be greatly reduced. These techniques are elaborated in chapter 6.

developing and administering novel computer science assessments — As a proof of concept for these techniques, this document considers several courses in computer science education, including CS1 (Introduction to Programming) and Logic for Systems. Chapter 7 shows that a generated assessment produces more useful population-level information than an extant expert-developed instrument, and a novel assessment instrument for linear temporal logic, a topic with no existing instruments, is also presented.

instructor-defined specificity — One of the challenges in pedagogical research is developing assessments that are granular enough to provide specific insight into individual pedagogical decisions. In chapter 7, it is demonstrated that instructors or researchers can create assessments with precise topical scope.

student-authored questions — Chapter 7 explores the quality of student-authored assessment items. Outside of the reduction in cost to generate assessments, there are two primary benefits to having students author assessment items (in addition to or instead of instructors). The first is that there is some evidence that students can benefit from participatory authorship of learning content, an approach called Contributing-Student Pedagogy. The second is that a large pool of students can generate a richer and more informative variety of questions than a small team of experts, as evidenced by chapter 7. This is because it is difficult in general for an individual to imagine all of the ways someone else might think about a topic, and this is especially hard for experts (the so-called "expert blindspot"). Across a large pool of students, there can be individuals who exhibit mental models that are literally "unthinkable" to experts.

Chapter 2

Related Work: Theories of Testing

The work presented in this dissertation represents a new direction in terms of theories of testing. In particular, the modeling assumptions used here are different from the modeling assumptions used by the prevailing theories of educational testing. Although the quantitative objective introduced in chapter 3 is compatible with any assessment and is, for the most part, model agnostic, the approach that it suggests, and the resulting algorithms, are substantially different from most of the educational testing literature.

This chapter will attempt to do two things. First, for readers unfamiliar with education research, this should provide a launching point for better understanding the context of the current work. Second, for readers who are familiar with education research, this chapter should illuminate the places where the present work diverges from the theoretical underpinnings of of the established models.

2.1 Constructivism

The work in this dissertation is mostly compatible with, but not derived from, a prevailing modern philosophy of education, called ***Constructivism***. At its simplest, constructivism posits that "knowledge", rather than being some symbolically communicated Platonic ideal, is actually highly subjective and actively *constructed* internally by the learner based on their experiences. The philosophy was significantly driven by the experiments of child psychologist Jean Piaget (Piaget and Cook, 1952), and is now relatively uncontroversial (at least in its main thrust) from a neurological and cognitive perspective. In terms of the philosophy's implications for assessment, however, there is less consensus. Some proponents of the theory propose holistic evaluation in authentic settings where the behavior of the student can be reasonably compared with that of an expert, and they reject on principle the validity of anything resembling a pen-and-paper test (Lutz and Huitt, 2004). Others, particularly in empirical or mathematical fields, contend that knowledge in their domains is not about social convention as much as about predicting the behavior of external systems (Osborne, 1996), and that those predictions can be made in highly abstracted contexts.

The work in this dissertation is informed by constructivism in the sense that it accepts the

possibility (and even likelihood) that there are multiple ways to think about a topic (multiple mental models, or conceptions), that knowledge cannot be *directly* transferred between individuals, and that knowledge can be only indirectly exhibited. But the work in this dissertation is also more pragmatic than the more extreme applications of constructivism to theories of assessment. In particular, the performance objective in chapter 3 is premised on the belief that assessment is valid to the extent that it is statistically predictive of other valid assessment, and that validity is a continuous (and not a binary) measure. The model of assessment used here precludes neither artificial assessment contexts (like forced-choice questions) nor holistic assessment in authentic contexts. (It should be noted that in practice, all of the experiments here have used free-response or forced-choice assessment items for reason of convenience of data collection.)

2.2 Concept Inventories

The "gold standard" for education research assessments are ***concept inventories***, which are validated forced-choice assessments with the special property that each response to an assessment item corresponds to single mental model (Hestenes et al., 1992). Unfortunately, the traditional process for developing assessment items that meet these requirements is very labor intensive (Goldman et al., 2008; Taylor et al., 2014). (See fig. 7.1.) As a consequence, very few of these instruments exist.

In terms of guiding philosophy, the work presented here is very well aligned with concept inventory development. In particular, highly informative assessment items that identify student mental models are likely to be prioritized by the formal objective presented in chapter 3, and also satisfy the qualitative design criteria of being valid, reliable, efficient, interpretable, and actionable. The only limitation to the existing work on concept inventories is the enormous expense required to produce assessment items of such high quality.

The work in this dissertation questions some of the assumptions around the accepted process by which concept inventories are created. For example, do questions really have to be written by experts if the quality of questions written by non-experts can be accurately measured? Do misconceptions have to be identified through lengthy one-on-one interviews, or can they be identified from a large number of written responses from students? Do assessment items have to be piloted in a batch process, or can computational tools be used to evaluate assessment items on a continuous basis?

2.3 Item Response Theory

In the typical classroom, assessments are administered statically - all students answer all questions, and the individual items are only evaluated after all the questions have been answered. While this approach is simple and near-universal in its use, it is highly suboptimal. If we assume that some questions are harder than others (an assumption that will be validated by experiments presented shortly), the average student will spend a lot of time on questions that are either too easy or too hard to be meaningful. As early as 1985 (Weiss, 1985), researchers proposed that adaptive testing

(or Computer Adaptive Testing, CAT) could be used to reduce the number of questions asked of each student by selecting questions as a function of the student's previous responses. Depending on the number of items and how they are related, the number of items assessed in an adaptive testing paradigm could be logarithmic in the total number of assessment items, representing an exponential improvement over the efficiency of assessment in common practice.

The prior state-of-the-art methods for adaptive testing all use the Item Response Theory model, which predicts the likelihood of a correct response as a function of several parameters. In Item Response Theory, the simplest model is known as the 1-parameter logistic model, or the 1PL model. In a 1PL model, the i'th learner is modeled by a single parameter θ_i called ability or proficiency, and the question/item is modeled by a difficulty parameter d_j. (All IRT models use only skill parameters for each student, so the order of the model refers to the number of parameters used to model each assessment item.) If we add a parameter a_j that specifies the discriminability of the question, the model is known as a 2PL model. If we incorporate a parameter c_j that specifies the likelihood of a guess, we have a 3PL model. Each question in IRT has an associated item response function, typically the logistic function. The difficulty, discrimination, and guess parameters reshape the logistic function as follows:

$$p_j(\theta) = c_j + \frac{1 - c_j}{1 + e^{-a_j(\theta - d_j)}}$$

If the ability and difficulty parameters are allowed to be multi-dimensional, the framework is called Multi-Dimensional Item Reponse Theory (MIRT). At the time of writing, no general framework for MIRT model learning operates directly from student response data with no expert input (Chalmers et al., 2016).

Plajner (2016) introduces a straightforward method for building CAT models with IRT. They use empirical bayesian estimates of the latent parameters based on answers, and compute the information provided by asking a given question, consequently picking the question that greedily maximizes the information at each timestep. The use of IRT in adaptive testing is well-established (Vie et al., 2016).

Fundamentally, IRT is rooted in the assumption that there are (at most) a small number of latent continuous skills that independently predict correctness on each item. This assumption of conditional independence among the items given the skills is very elegant, allowing efficient model inference and preserving the simplicity of the model. However, as will be seen in experiments presented in chapters 5 and 6, the strong independence assumptions of IRT are limiting when assessment items have explicit interrelationships.

Despite the enormous potential of adaptive testing, adoption in educational (and educational research) practice has been limited. Initially, this may have been due to the limited availability of computing machinery, but today it is likely due to limitations in the techniques presently used for adaptive testing. In particular, IRT models make strong independence assumptions that break down when assessment items have explicit dependency relationships, they typically model only one skill at a time, and the greedy query strategy typically used may be suboptimal. Additionally, existing techniques interface poorly with the resources available in the educational ecosystem, often requiring large amounts of expert time and effort to create new assessments, as well as significant

trial populations for iteration on the assessment in question.

IRT is trying to solve the same problem as the work in this dissertation; notably, it is explicitly optimizing (under the modeling assumptions) a myopic version of the objective presented in chapter 3. Thus, the oldest and most established statistical theory of testing is directly comparable to the work presented here. The largest difference between this work and IRT methods is in their modling assumptions. Fundamentally, IRT models assume that student knowledge can be represented on a low-dimensional (typically uni-dimensional) continuum, whereas the work in this dissertation assumes that student knowledge can be modeled as a discrete set of interrelated masteries. This difference in modeling assumptions leads to a difference in predictive performance, as shown in chapter 6.

Chapter 3

Formal Objectives for Assessments

The purpose of assessments is to measure what students know. Unfortunately, there is no source of ground truth for this measurement. Unlike physical properties that can be grounded in simple and highly-replicable procedures, such as measuring length using a ruler, knowledge is an abstraction we apply to an otherwise inscrutable mass of neuronal interconnections and active firing patterns, and is, for all practical purposes, impossible to observe directly.

To top it off, the "knowledge state" of a person may effectively be an infinite-dimensional object, and the coarse division of knowledge into a relatively small set of topics and skills is almost surely an arbitrary convenience that butchers the true subtlety of human thought. Even getting a group of experts to agree on what it means for a student to have mastered a given topic is a noisy and inconclusive process , suggesting that what we are aiming at is ill-defined.

Despite these challenges, assessments already fill an essential role in modern education, and they offer a promising direction for educational science to begin answering basic questions, like, "What is the most effective way of teaching topic X to student Y?" The goal of this chapter is to clarify and operationalize a reasonable set of mathematically formal objectives for a created assessment. While accepting that these objectives for assessment may be subject to opinion, they will enable the rest of this dissertation to rest on a well-defined and rigorous evaluation metric.

3.1 Inspirations

Since the objective function introduced in this chapter is novel to the field of education research and its value is subjective, this section will attempt to motivate its design by looking at several analogous choices of objective that have born significant fruit in their respective disciplines, including multiple examples from machine learning. These comparisons should suggest not only that the formal objectives in this chapter are likely to be useful, but also that they are entirely reasonable.

3.1.1 The Imitation Game, Cryptography, and Constructivism

Alan Turing, widely considered the founder of modern computer science, and by all accounts a genius, happens to have also done some of the first work in the study of artificial intelligence. In what has become a very famous paper, he poses the question "Can machines think?" Turing (1950). Rather than agonize over philosophical quandaries, Turing proposes the following (thought) experiment: Can an intelligent tester, using only written communication, distinguish between a person and a machine pretending to be a person? Since there are considerable practical difficulties in actually implementing this test, it makes a poor objective measure for artificial intelligence, but it suggests the following insight in our particular problem: Rather than ask whether a student knows something, it may be more useful to ask whether a given student can be distinguished from an expert.

This indistinguishability criterion has good company; a similar objective forms the basis of modern cryptography. Modern cryptographic guarantees rest on showing that it is computationally hard to distinguish between an encrypted 1 and an encrypted 0. (There are a few equivalent formulations of this condition.) This formulation sets up adversarial objectives for the encryption algorithm and a hypothetical codebreaker that can parallel the roles of teaching methods and assessment methods. Teaching methods aim to enable students to be indistinguishable from experts, and assessments aim to tell them apart.

Although education research does not currently have a rigorous holistic performance objective for assessment generation, the criterion of indistinguishability is well-aligned with the educational philosophy of constructivism. ***Constructivism*** posits that learning consists of students constructing knowledge for themselves based on their experiences. The philosophy was significantly driven by the experiments of psychologist Jean Piaget Piaget and Cook (1952) and has significant ramifications for assessment. Namely, since knowledge is internal to each student and largely of their own construction (in response to external stimuli), there is little point in assessing "knowledge" directly. Rather, proponents of the theory propose evaluation in authentic settings where the behavior of the student can be reasonably compared with that of an expert Lutz and Huitt (2004). Although the work of this thesis is less prescriptive on the issue of "authenticity" of the assessment, it thoroughly aligns with the goal that learning makes students indistinguishable from experts under the assessment of relevance.

The indistinguishability criterion will also show up in another way in this dissertation. Experiments presented later in this document will show that assessments designed to meet the evaluation criteria discussed in this chapter possess many of the qualitative characteristics one might hope for in assessments generated by traditional methods. In other words, if the assessments created by this work can pass for assessments generated by a traditional process for generating high-quality assessment instruments, they can be considered high-quality instruments, regardless of the process that generated them.

3.1.2 Time is What a Clock Measures

There is a quote commonly attributed to Einstein: "Time is what a clock measures." While this may initially seem a dodge of the question of our perception of time or the fundamental reality of time, it is in fact the basis of one of the most profound physical theories ever created. By defining time in terms of instruments that measure time, Einstein opens the door for the reasoning about time by considering the effect of observed physical law on the well-understood mechanisms of clocks. By considering the effects of a finite (and constant) speed of light on properties like the simultaneity of clock readings, the Special Theory of Relativity was created.

In a similar manner, rather than focus on the mental states an assessment should distinguish, it may be more beneficial to consider the outcomes of the assessment items themselves. While this work attempts to be as general as possible with respect to the nature of the assessment items used, many kinds of assessment item have outcomes that can be reasonably discretized (whether forced-choice questions, multiple checkbox selections, or even rubric-graded responses). Without any consideration for the underlying significance of the prompt of an assessment item, its outcome can be treated as a random variable and a source of data. This shift in focus will allow a mathematically rigorous treatment of an assessment in terms of random variables, joint distributions, and algorithms.

It should be noted that this shift in focus does not obviate the need for human involvement, philosophy, or interpretation. It is merely a separation of concerns. While assessment items and teaching methods likely require human expertise to design, the construction and administration of an assessment from a *human-constructed* pool of assessment items (even of variable quality) can be treated purely as a problem of operating on data.

3.1.3 Self-Supervised Representation Learning

Fields like computer vision and natural language processing have had enormous successes in recent years through the use of techniques that learn convenient representations for their data. In contrast to techniques that evaluate representations in terms of usefulness for some end task, some of the most successful techniques have used a form of self-supervision where the goal is to reconstruct some removed part of the data using the rest of it.

This form of self-supervision is also related to the performance metrics for semi-supervised, active, and meta- learning, all of which involve extrapolating information about some portion of the data to the rest of it. Note that Bayesian information gain is a commonly-used objective for active learning settings Anderson and Moore (2005). The goal is to select a set X to observe to minimize the entropy of the resulting belief distribution over the data $H(b(X))$ where H is the Shannon information entropy, b is the belief distribution derived from a set of observations, and X are the observed data. This formlulation is equivalent to maximizing the difference $H(D) - H(D|X)$, which is called Global Entropy Reduction Maximization (GERM) Yu et al. (2010).

3.2 Problem Inputs and Outputs

This thesis is particularly focused on the setting where many (potential) assessment items are available. Let $Q = \{Q_1, Q_2, ...Q_m\}$ be the set of assessment items[1] with corresponding outcome sets $O_i = \{O_1, O_2, ...O_{|O_i|}\}$. Let S be a population of n students drawn i.i.d. from some distribution over outcomes $\mathbb{P}[Q]$.

An observation history for a student $\omega_{s,t}, s \in S$, where t reflects a timestep, consists of a map of assessment items to outcomes for that student $\{Q_{t_1} : o \in O_{t_1}, ..., Q_t : o \in O_t\}$. A collection of observation histories for students 1 through i is denoted $\Omega_{[1 \rightarrow i]}$.

An assessment is an algorithm A that takes as input the set of assessment items Q, a collection of observation histories for previous students $\Omega_{[1 \rightarrow i]}$, the observation history for a current student $\omega_{(s,t)}$, and a maximum number of queries per student k. If $t < k$, it outputs the index of an assessment item to administer to student s. Otherwise, it outputs predicted probability distributions for a subset of the assessment items $\hat{P}_s = \{Q_j : \hat{\mathbb{P}}[Q_j] | Q_j \in Q', Q' \subseteq Q\}$. Whenever the assessment selects an assessment item, that item is administered to the student in question. The outcome is then added to the observation history provided as input to the assessment algorithm in the next time step.

3.2.1 Why not a batch problem?

In this section, assessments are defined in a way that is amenable to online learning (in the machine-learning sense) with respect to streams of both students and queries for each student. The motivation for the latter (adapting questions based on a history of interaction with the student) allows for adaptive assessment, which has potentially much higher data-efficiency than a static set of assessment items. The motivation for the former (learning as new student data is accrued) permits lifelong learning while still being compatible with a traditional batch process.

3.2.2 Why not just predicted outcomes?

In the context of this dissertation, assessment algorithms output a probability distribution over outcomes for a subset of items. This choice is uncommon for machine-learning problems, which more often simply output predicted values for all items, but it has several advantages. First, the probability distribution can reflect a degree of confidence in the results. It can be useful for instructors trying to distinguish between predictions that their class has split responses and predictions that simply lack data. Second, outputting predictions for only a subset of items reduces the burden on the assessment to validate every possible assessment item (the number of which might even exceed the number of available students). Third, when assessment items have more than two possible outcomes, the predicted probability distribution can provide information about which outcomes are *not* likely, even if the actual outcome is still uncertain.

[1] 'Q' is for question, since 'A' is used for assessment

3.3 A Data-Driven Objective

Suppose we wish to evaluate an assessment algorithm A on a pool of students S for a set of assessment items Q. A natural objective is to maximize the amount of data we gain about each student (in an information-theoretic sense). That is, given the assessment algorithm's choice of items for student $s \in S$, how much information was gained on the assessment items in Q as a whole?

Definition 3.3.1 (Expected Empirical Bayesian Information-Gain Objective)**.** *The objective function will be stated in equivalent forms, the first using Shannon information entropy and conditional cross entropy, and the second using expected surprisal. In this application domain, larger values are better:*

$$U(Q,S,A) = \frac{1}{|S|}\sum_{s_i \in S}\left[\sum_{Q_j \in \hat{P}_{s_i}}\Big(H(\mathbb{P}[Q_j]) - H(\mathbb{P}[Q_j|\omega_{s_i,k}], \hat{P}_{s_i}[Q_j|\omega_{s_i,k}])\Big)\right] \tag{3.1}$$

$$= \mathop{\mathbb{E}}_{s_i \in S}\left[\sum_{Q_j \in \hat{P}_{s_i}}\Big(-\log \mathbb{P}[Q_j = s(Q_j)] + \log \hat{P}_{s_i}[Q_j = s(Q_j)|\omega_{s_i,k}]\Big)\right] \tag{3.2}$$

where $\hat{P}_{s_i} = A(Q, \Omega_{[1\to i-1]}, \omega_{(s_i,k)}, k)$.

A proof of the equivalence of these two formulations is given in appendix B

3.3.1 Why information and not accuracy?

Many assessment items are uninformative with respect to a population of interest . For example, they may be too easy or only have one reasonable answer. From an assessment point of view, these questions are very wasteful, as they cannot be used to distinguish between a novice and an expert (a la Section 3.1.1). Under the proposed information-based objective, collecting data on these wasteful questions has no value. Under an accuracy-based metric, the assessment algorithm would be incentivized to identify as many of these easily-predictable assessment items as it can, quite possibly at the expense of identifying relationships between higher-entropy items.

3.3.2 Why summed entropies instead of joint entropy?

A previous publication by the author Saarinen et al. (2020) explicitly suggests using mutual information between assessment item outcomes as a measure of relevance, and this and other information-theoretic measures of item reliability are correlated with the widely used Cronbach's Alpha Fokoué and Gündüz (2016). If, as is the case for work presented later in this dissertation , the assessment items are generated independently by a collection of people with subject knowledge in response to a fixed prompt or topical scope, each item can be thought of as a vote of relevance for the information captured by its responses. The summed entropies objective favors capturing the information that was significant to the largest number of assessment items, whereas a joint-distribution-based objective would comparatively favor questions capturing unrelated information. A joint-distribution-based objective could potentially lead to an assessment that only administers irrelevant yet high-information items.

3.4 Problems with Existing Assessment Statistics

One of the contributions of this dissertation is the introduction of a formally rigorous evaluation criterion to the problem of assessment generation. It represents the first holistic objective of its kind, although there are a few extant formal reliability measures for assessments or assessment items. By far the most popular McNeish (2018) of these measures is Cronbach's Alpha Cronbach (1951). Cronbach's Alpha is inapplicable in this setting, as it assumes assessment items produce (possibly continuous) outputs in a single dimension and cannot be used as is for the design of an effective assessment strategy. Since Cronbach's Alpha only considers the internal consistency of an assessment, it is maximized by using an assessment consisting only of repeated administration of the same single assessment item. All of the alternatives to Cronbach's Alpha are similarly designed for low-dimensional continuous scales and have similar drawbacks in their applicability to the development of assessment algorithms McNeish (2018); Hattie (1985).

Although there are no assessment-algorithm objectives specified in the machine-learning literature, it is worth highlighting that a few conventional supervised learning objectives have been used for empirically validating structure extracted from test-question data Chen et al. (2016). As just mentioned, these objectives are only for the problem of structure recovery, use outcome-prediction based performance, and are unsuitable for the data-collection paradigm considered here.

Chapter 4

Why Prerequisite Maps

Perhaps the simplest predictive model of student responses is one that only looks at population-level statistics for each assessment item, ignoring joint-distribution effects between questions. The query optimization strategy under that model is trivial: query the assessment items with the highest population-level entropy, in any order. While the problem of efficiently finding the highest-entropy assessment items is interesting (and addressed by this dissertation), this approach by itself leaves a lot to be desired. For example, semantically identical assessment items would either be all queried or completely unused. In the extreme, the entire assessment might consist only of repeats of the same question; a model that models inter-item relationships could equal its performance on the objective measure using only a single assessment item!

On the other hand, finding and using the best joint model of question relationships is likely computationally difficult (in fact, Bayesian network inference is NP-Hard (Dagum and Luby, 1993), as is network optimization (Chickering et al., 2004)). This leaves an opportunity for some model with intermediate complexity to approximately model the joint distribution of assessment item responses in a computationally-efficient manner. Enter prerequisite maps.

4.1 Prerequisite Map Assumptions

A prerequisite map is an acyclic directed graph structure where a node represents a component of knowledge and a directed edge indicates that the child (the destination node) requires knowledge of the parent (the source node). The prerequisite map model assumes that knowledge objects are independent outside of the (transitive) dependency relationships.

Despite being relatively unused for assessment, prerequisite maps are a natural and commonly used model for human knowledge (Adorni et al., 2019; Bayer et al., 2012; Botelho et al., 2016; Brunskill, 2011; Chang et al., 2015; Chen et al., 2016, 2015; Han et al., 2017). Anecdotally, it seems that every accredited educational institution has a notion of topical ordering or knowledge prerequisites, so the idea that prerequisite relationships exist in education is not a controversial idea.

What is more controversial is whether the prerequisite relationships are a phenomenon in their

own right, or the by-product of some deeper description. For example, Item Response Theory (the only predictive statistical model of assessment currently in widespread use) models prerequisite relationships implicitly as differences in difficulty with respect to some continuous underlying skill. Under an IRT model, it is possible yet unlikely that a student correctly answers difficult items more frequently than easier items. This is impossible (outside of statistical noise due to guessing) under a prerequisite map.

4.2 Benefits of Prerequisite Maps

Because prerequisite maps are constructed using pairwise relationships, it is feasible to design algorithms to learn such structures with polynomial time- and data- complexity. Furthermore, the semantic significance of the pairwise prerequisite relationships lends a level of interpretability to the resulting structure. In addition to being efficient, prerequisite maps generally improve on the accuracy of other tractable models. As shown in chapters 5 and 6, prerequisite maps demonstrate higher predictive accuracy than IRT models on a variety of assessments.

4.3 Contributions of this Work to the Prerequisite Map Model

While prerequisite maps are a natural model and have been studied in a variety of contexts, they have mostly been used descriptively, and not for statistical prediction. To that end, this work introduces a few innovations that do not appear in earlier literature. First, the prerequisite map is constructed over equivalence sets of assessment items. Natural extensions of transitive closure and transitive reduction algorithms are used to accommodate equivalence relationships. Second, the pairwise relationships are modeled with statistical allowances for two types of noise in the correctness of outcomes ("mistakes"/"slips" and "guesses").

In addition to these extensions to the underlying prerequisite map model, this work presents novel construction, inference, and querying algorithms, as detailed in the following chapters.

Chapter 5

Prerequisite Maps can be Learned from Data

This chapter expands on work first published in Saarinen et al. (2020).

5.1 Introduction

This chapter attempts to bridge the gap between two communities of knowledge-modeling research. The chapter is specifically built around the question, "When does modeling assessment item interdependence improve predictive accuracy?" This introduction will provide context for the chapter and distinguish this work from related work in the literature.

5.1.1 We are Discovering Prerequisite Structures

Although this chapter uses an adaptive testing evaluation framework, the techniques are most closely related to the Prerequisite Inference literature. There are many educational uses for identifying dependencies between topics, concepts, or questions. These uses include defining constraints on curricular order (what order should topics be taught in to maximize student learning) Brunskill (2011), providing course recommendations Bayer et al. (2012) , designing adaptive testing systems (and inferring student knowledge) Lynch and Howlin (2014), and efficiently validating new test questions. Although the exact form of such relational structures has varied across the literature, this chapter will call all such devices **dependency maps**. Prior work has attempted to deduce such dependency maps from a variety of data sources using a variety of techniques and evaluation methods. See table 5.1 for a summary.

This work is partly motivated by the problem of detecting student knowledge efficiently using student-sourced questions, a promising approach to scalable assessment generation and adaptation Saarinen et al. (2019). Due to minimal expert oversight, there is no ground-truth source of skill-labelings for questions assessing the same skill or knowledge, nor is there a ground-truth dependency

Table 5.1: Approaches to prerequisite map inference are grouped broadly by approach to validation, then by exact validation method, then by source of data. This paper introduces a new evaluation framework for dependency maps and evaluates a novel technique inspired by several existing ones.

Data Source	Validation Method	Technique	Reference
Expert (or Simulated) Dependency Map Recovery			
Student Answers to Test Questions	Plausible Structure Recovery	Expectation Maximization on Pairwise Relationships	Brunskill (2011)
Pairwise Interaction Features	Human Evaluation	Various Regression Algorithms	Chang et al. (2015); Botelho et al. (2016)
Course Enrollment and Grades	Reproducing Existing Course Prerequisites	Ranking by Conditional Success Ratios	Bayer et al. (2012)
Probabilistic Student Knowledge States from Test Questions	Rediscovery of Simulated and Expert Structure	Probabilistic Association Rules Mining	Chen et al. (2015)
Student Answers to Test Questions	Rediscovery of Simulated and Expert Structure	Bayesian Model Selection	Han et al. (2017); Chen et al. (2016)
Data Self-Supervision			
Student Answers to Test Questions	Leave-One-Out Cross Validation	Structural EM for Bayesian Model Selection	Chen et al. (2016)
Student Answers to Test Questions	Data Reconstruction Error	Restricted Bayesian Inference (DIDACT)	**this chapter**

map to validate against (so it cannot be used to measure performance of algorithms designed for this problem). Furthermore, because the student-contributed questions are often written without global awareness of the other questions available, many questions are related or equivalent. This motivates an adaptive testing system that attempts to minimize the number of questions needed to accurately predict student performance. (Note that even with expert-authored questions, experts may wish to validate their own dependency maps empirically, or to save themselves the effort of creating one manually.)

This work aims to learn a dependency map on the basis of explaining (or predicting) the observed data, so the works closest to this paper are the attempts to use Bayesian inference to infer prerequisite relationships among latent skills, given the mapping from assessment questions to required skills Brunskill (2011); Han et al. (2017); Chen et al. (2016). Although those approaches are promising and able to reproduce small artificially-generated or expert-defined structures, they suffer from two primary limitations. First, the ground-truth mapping from questions to measured latent skills is not available in the problem domain considered here. Second, Bayesian inference methods are generally both approximate and slow, limiting their scalability. This paper considers structures with an order of magnitude more nodes than those studied in prior work.

The algorithm explored here, DIDACT, also bears resemblance to the prior Probabilistic Association Rules Mining work Chen et al. (2015). The work presented here differs primarily in that this paper explicitly considers the problem of predicting or filling in values in the dataset, and the algorithm has been generalized to allow item equivalence.

There is also a fascinating body of work into Dependency Map learning from natural language sources (Adorni et al. (2019), for example), but those techniques require a large text corpus (such as a textbook), are not designed for relating assessment items, and the evaluation method presented here is fundamentally different.

There is also work on predicting student responses using supervised learning Liao et al. (2017),

but that work only applies to predicting responses to a fixed set of questions given responses to a different fixed set of questions, making it inapplicable for either detecting prerequisite relationships or facilitating adaptive testing.

Finally, we also note that the methods presented here exploit algorithms on directed acyclic graphs (DAGs) to explicitly simplify the output and enforce global constraints in the dependency map, a technique that has not appeared in the prior literature.

5.1.2 We do *NOT* use a Q-Matrix

Many approaches to inferring dependency maps aim to simplify the problem through use of a **Q-matrix**, which maps a number of assessment items to a smaller number of latent knowledge variables. Q-matrix Q has $Q_{ij} = 1$ if question i uses skill j, and 0 otherwise. If an exam is built by experts, a Q-Matrix may be hand-coded. In our setting however, we do not use a Q-matrix. Instead, we design an inference algorithm that scales well to large numbers of assessment items.

5.1.3 We are Doing Adaptive Testing

Computer Adaptive Testing (CAT), or simply Adaptive Testing, has a rich history in the literature, dating back to 1985 (Weiss (1985)). In recent years, many innovations in Knowledge Modeling have been carried over to an Adaptive Testing setting Plajner (2017). We continue this tradition, but with a novel evaluation framework for adaptive testing that provides rich information around the tradeoff between data-efficiency and accuracy.

5.1.4 We compare to Item Response Theory

Although it should now be apparent how IRT and Dependency Map inference can be both used within an adaptive testing framework, it may be beneficial to clarify their differences. Fundamentally, IRT is rooted in the assumption that there are (at most) a small number of latent continuous skills that independently predict correctness on each item. This assumption of conditional independence among the items given the skills is very elegant, allowing efficient model inference and preserving the simplicity of the model. In contrast, Dependency Map inference is fundamentally premised on the idea that assessment items exhibit interdependence.

5.1.5 We are *NOT* Doing Knowledge Tracing

Knowledge tracing is the task of modelling student knowledge *over time* to accurately predict future student performance Piech et al. (2015). When systems can accurately model student knowledge, content can be suggested to students based on individual needs. In the literature it is common to use a Bayesian model of the knowledge of a student, updating learner's latent knowledge using a hidden Markov model as learners interact with exercises Vie (2018). Recent models propose using recurrent neural networks to predict student responses based on their past activity Piech et al. (2015);

Minn et al. (2018). The fundamental difference between knowledge tracing (KT) and CAT is that while in KT system designers are trying to maximize the student's knowledge through exercises that teach concepts, CAT is focused on *testing* a student's knowledge, as accurately and efficiently as possible. This is not to say that the two tasks are unrelated—both KT and CAT use models of student knowledge. For example, the use of IRT and MIRT models for knowledge representation, Bayesian networks, and Q-Matrices are used throughout both the KT and CAT literatures.

5.1.6 Contributions

This chapter presents three primary contributions. First, a quantitative evaluation framework for adaptive testing is introduced that allows control of the tradeoff between data efficiency and accuracy through a settable parameter γ. Second, a fast algorithm for mining dependency relationships and doing adaptive testing is presented. This algorithm does a restricted form of Bayesian reasoning that achieves high accuracy, brief runtime, and high data-efficiency. Third, experiments on real and simulated data suggest that modeling of item interdependencies has a significant impact on predictive power when the assessment is narrow in scope.

5.2 Validation Method

The value of a model should ultimately be measured by how well it predicts unseen/new data. This perspective is inherently captured by the adaptive testing problem, where the goal is to ask questions until the student's responses to the remaining items can be predicted with high accuracy. There are two primary objectives involved in adaptive testing systems. The first is efficiency—to minimize the number of questions asked. The second is robustness. Adaptive testing suffers from asymmetrical error conditions whereby asking unnecessary questions is much less expensive than mislabeling student knowledge of an item. These two kinds of error are difficult to compare directly in terms of, for example, total cost in student time, so we use a proxy condition: All inferred student responses should be provided with at least some minimum **accuracy threshold** denoted γ. For example, $\gamma = .95$ indicates a model should only predict the student's response to a question if it is at least 95% likely to get it right. This requires the model to both have high accuracy and to *know* that it has high accuracy. This setup motivates the following active-learning-style problem:

1. Train on a dataset of previous student correctness scores on a variety of assessment items, possibly with missing values.

2. For each (test set) student, repeatedly select a question to ask and then receive a response, or issue a stop command.

3. After the stop command, predict the student's responses to any remaining questions.

4. For every predicted response that is correct, give score 1. For every predicted response that is incorrect, give score $-\frac{\gamma}{1-\gamma}$. This penalty gives expected score 0 when the algorithm has exactly

confidence γ. Note that questions that were asked (not predicted) receive score 0.

This scoring scheme is simple and allows traditional train/test splits, cross validation, or online learning evaluations. It is in the best interest of the tested algorithm to only predict responses that it believes it will get correct with probability greater than γ and to ask the question if its confidence is less than γ. If its confidence is exactly γ, guessing or asking yield the same score in expectation.

This metric allows us to explore the tradeoff between data efficiency and accuracy by adjusting γ. With γ equal to 0, the score is the number of questions that were inferred correctly without being asked. If the score is normalized by the total number of questions, this is a fairly direct measure of the "efficiency" of the adaptive testing system — how many questions (on average) the system is able to predict responses to without asking them. Here, the baseline to compare to is an algorithm that just guesses that each student will do what the majority do on each item (get it correct or incorrect). This baseline asks no questions of new students, so the only way to improve over its score is by using responses to some questions to improve the accuracy of predictions made on the rest (by modeling student ability or inter-question relationships, for example). Note that it is difficult to achieve scores near 1 when there are only a small number of assessment items, due to the proportional cost of gathering information. However, as the number of assessment items grows, the opportunities for modeling to accurately predict responses to a large fraction of the items increases.

At high γ, the model is primarily concerned with accuracy; with such a steep penalty for wrong predictions, the model will be willing to ask many questions in order to ensure that each remaining inference is correct. Here, the baseline is an algorithm which asks all questions, achieving score 0 every time. While this baseline is not very efficient, it is perfectly accurate, and so suffers no penalties. The danger for algorithms based on models is that the models must not over-estimate their confidence of a student's response - otherwise they stand to suffer large negative penalties for incorrect guesses. At $\gamma = 1$, there is an infinite penalty for even a single incorrect inference, so any score above 0 is highly impressive. Note that if questions can be guessed (or mistakes made) with some probability ϵ (the maximum noise in the observation), models should simply ask most questions when $\gamma > 1 - \epsilon$. Along this line of thinking, the most practically relevant range on these plots is the range from $\frac{1}{2} \leq \gamma \leq 1 - \epsilon$. In this range, there will be questions for which majority rule is no longer a safe guessing strategy, but careful modeling still has a chance of inferring responses accurately. In terms of interpretation, $\gamma = \frac{1}{2}$ is the point at which asking a single question has about the same cost as simply teaching that content.

5.3 A Fast Discrete Model

Bayes Nets are quite general and very data efficient, but can be computationally slow to learn and do inference in as the number of variables grows. This chapter therefore presents a discrete model that balances the flexibility of interdependence modeling with the speed of inference under assumptions of independence. The algorithm considers the influence of observed responses on a given item as independent, *subject to the learned dependency map.* In other words, redundant evidence (from a

prerequisite of an observed prerequisite, for example) is filtered out, as is irrelevant evidence (evidence from items not transitively connected to the query item by the dependency map). This combines some of the best of Bayesian Networks (expressive capability and dependency modeling) with IRT (fast inference due to independence assumptions). This algorithm is called Directed Item-Dependence And Confidence Thresholds (DIDACT). Construction of the dependency map proceeds by 4 steps.

1. Prepare statistics on all pairs of test items. How many students are correct on both, only the first, only the second, or neither?
2. Sort prospective edges for the dependency map by the mutual information between that pair of questions.
3. For each prospective edge, determine if it is an equivalence relation, prerequisite relation, or other.
4. Add equivalence and prerequisite relations according to their sorted order, using a DAG structure over equivalence classes to enforce non-circularity of the dependencies.

For Step 3, a globally estimated guess parameter g is used to construct a test for the different relations. Let $\hat{a}$ be the estimated proportion of students who answer both questions incorrectly and let $\hat{b}$ and $\hat{c}$ be the estimated proportion who answered one question correctly or the other, respectively. If, with probability at least γ, $b > \frac{g}{1-g}a$ and $c > \frac{g}{1-g}a$, there is no relation. If exactly one of the inequalities is true with confidence greater than γ, then there is a directed relationship (one of the things can be known without the other, but not the other way around). Finally, if neither is greater, then the two are treated as equivalent.

Doing inference with DIDACT is likewise straightforward. Given a vector of previously observed answers:

1. For each node x, construct a partial reduction (all observed nodes with a transitive dependence on x that do not have another observed node on any of their paths to x).
2. Treat all observed variables as exerting independent influences on x. Take the product of their conditional likelihoods for $x = 1$ and $x = 0$ (we use Bayesian pseudo-counts to prevent probabilities of 1 or 0), multiply by the base answer rate for x, and then normalize over the two possible outcomes.

Finally, DIDACT uses a myopic active learning (item selection) rule dependent on γ—given its current predictions, see which question will increase its expected score the most in the next round. If the expected increase is non-positive, stop asking questions and predict the responses to all of the remaining questions. Although DIDACT is just one possible combination of dependency inference and independence assumptions, the plots in fig. 5.1 and 5.2 show that it is very fast and fairly accurate.

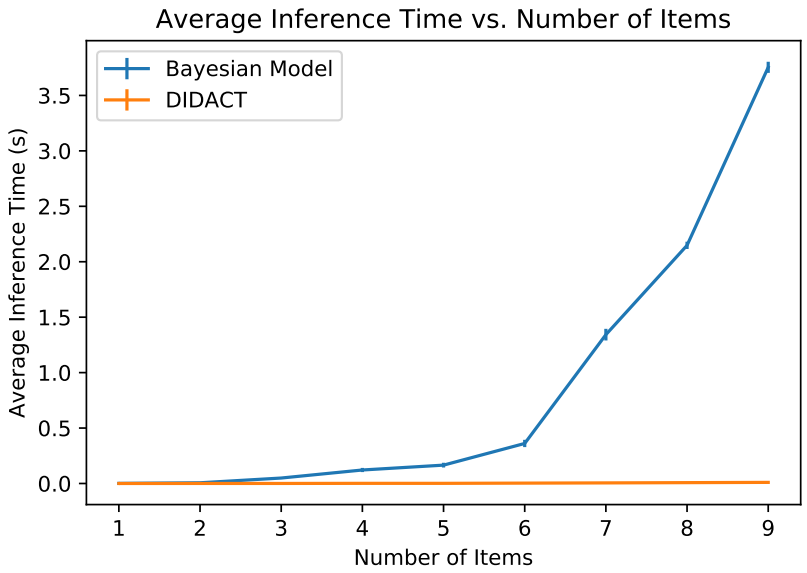

Figure 5.1: Exact Bayesian Inference quickly grows intractable, motivating the efficient DIDACT algorithm.

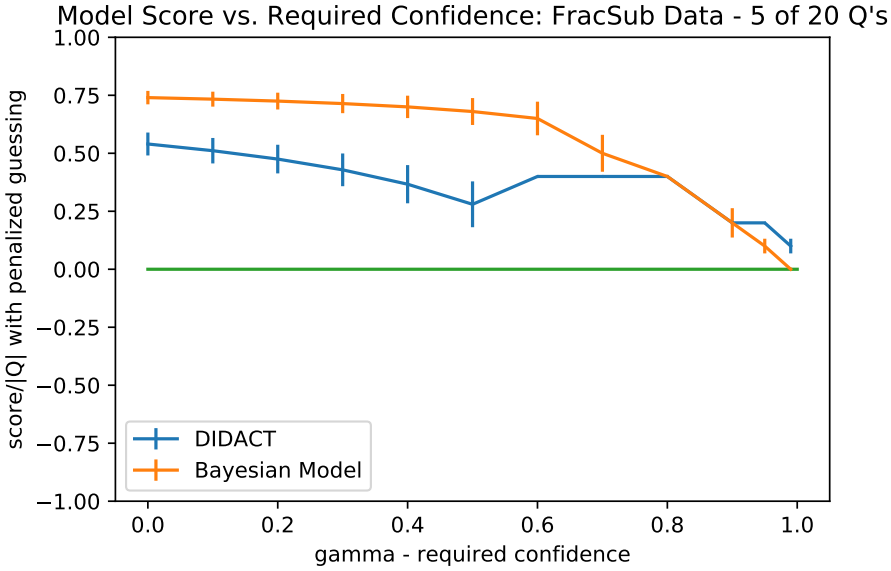

Figure 5.2: Although DIDACT is far from perfect, it still achieves good performance on real data, and with a much shorter runtime than exact Bayesian Inference.

5.4 Pairwise Inference without Noise Parameters

Although the approach given above is fast and works well in practice (see the following experiments), it is potentially unsatisfying that we must provide estimated bounds for the guess and mistake likelihoods. Is it possible to identify the latent relationship and estimate the noise parameters simultaneously? If we approach this problem naïvely, say by trying to minimize the KL-Divergence between our predicted pairwise statistics and the observed ones, we find that we cannot accurately identify known latent models, because the problem is over-parameterized. With 3 free parameters (4 numbers with the constraint that they add to 1), there are many ways to fit the data using one of 4 latent models, 2 knowledge parameters, and up to 4 noise parameters. But, by introducing priors that provide a monotonically increasing penalty for models with extreme knowledge-distribution parameter values or very large noise parameter values, we can get good accuracy in detecting pairwise relationships.

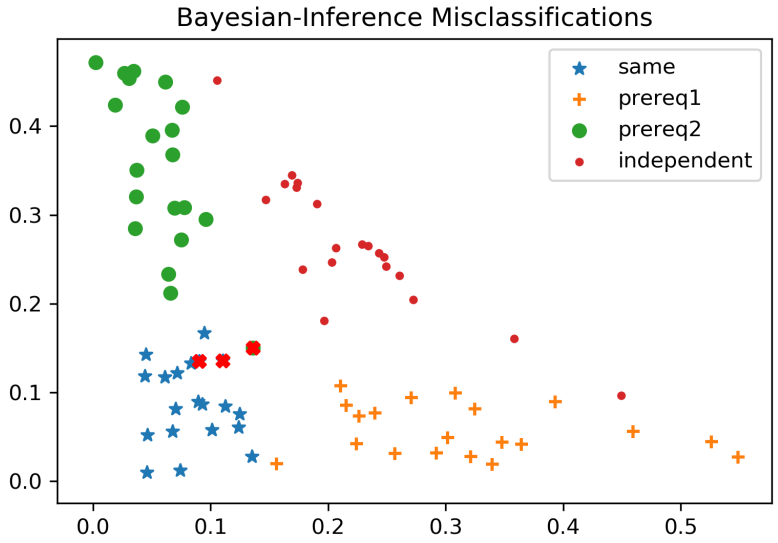

Figure 5.3: A Bayesian inference method using priors to regularize the solutions produces high accuracy in distinguishing known latent relationships on a synthetic experiment. Axes represent the probability a student gets one question wrong and the other right, or vice-versa.

5.5 Experimental Results

In this section, the results of using an IRT-based model and using DIDACT are compared.

5.5.1 We Use Human Data

Results below are based on two publicly available real-world datasets: the FracSub dataset (fig. 5.4); and the SAT dataset (fig. 5.5). The FracSub dataset includes graded responses from 536 students to 20 middle-grade math questions and was first published in conjunction with Wu et al. (2015).[1] The SAT dataset consists of responses from 296 students to 40 questions across multiple subject areas, and was first published in conjunction with Desmarais et al. (2011), and is available through the adaptive testing repository made available by Vie[2], which we also use for our IRT baselines.

5.5.2 Experiments show Benefits from Modeling Interdependence

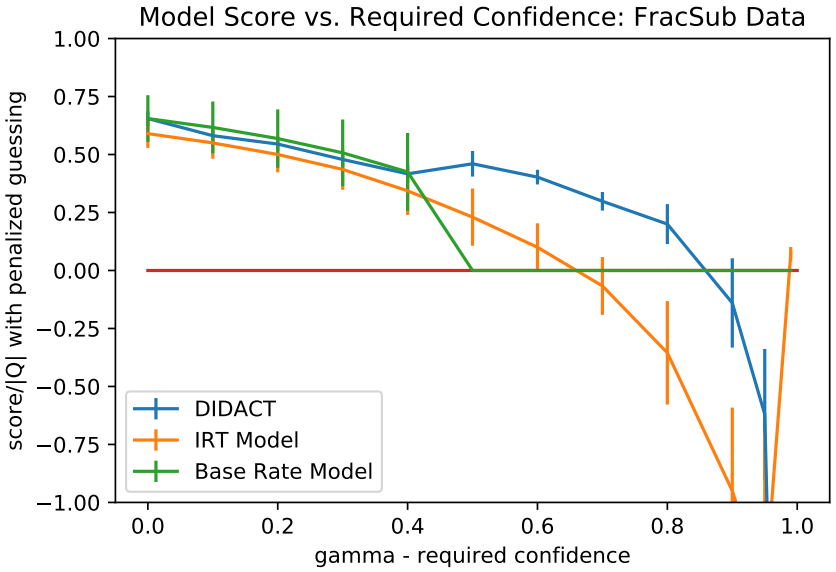

Figure 5.4: Adaptive Testing Performance subject to required confidence threshold γ on the FracSub dataset.

On the FracSub dataset (fig. 5.4), DIDACT and IRT begin with very similar performance, but the increasing γ shows that DIDACT has more accurate estimates of the likelihood of inferred answers. For convenience, a baseline algorithm is also plotted. The Base Rate algorithm estimates the base likelihood (without seeing any other answers) of each item being correctly answered. As long as an item's likelihood (or its complement) exceed γ, that item's response is inferred. Otherwise, the item is specifically queried. DIDACT achieves significantly higher accuracy once the base rate is no longer informative, although both models overestimate their own accuracy, as revealed at γ very close to 1.

[1] http://staff.ustc.edu.cn/ qiliuql/data/math2015.rar

[2] https://github.com/jilljenn/qna

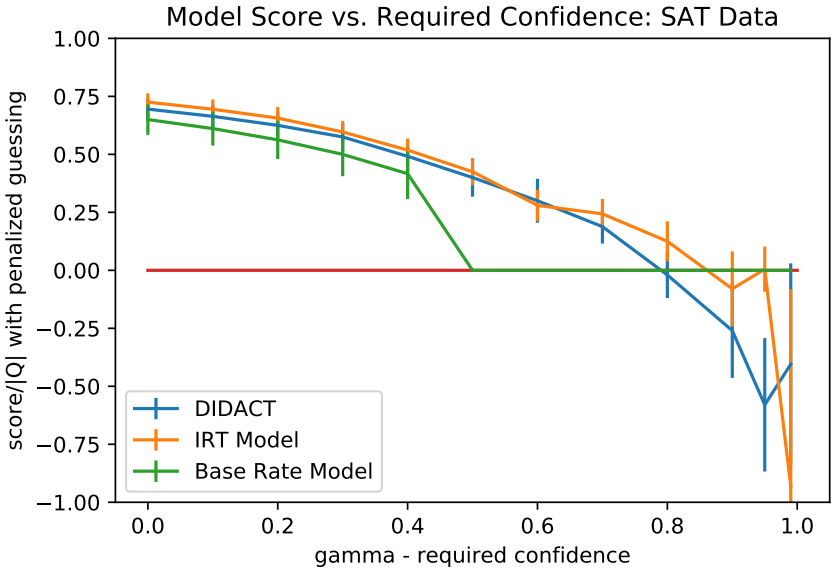

Figure 5.5: Adaptive Testing Performance subject to required confidence threshold γ on the SAT dataset.

In contrast, on the SAT dataset (fig. 5.5), the performance of these two models are roughly equal. (If anything, DIDACT performs slightly worse, although no statistically significant conclusion can be drawn given the error bounds.) What accounts for the difference in results between these two datasets? We note that the FracSub dataset involves many questions that are closely related semantically, whereas the SAT dataset includes questions from multiple unrelated subject areas. This suggests the following hypothesis: on closely related questions, question inter-dependence violates the conditional independence assumption of IRT, leading to worse performance than when the questions are nearly independent.

To test this hypothesis, experiments were run on two synthetic datasets. One is based on a prerequisite structure with very high interdependence; 10 items are placed in a single chain of prerequisite dependencies. (See fig. 5.6.) In the other dataset, (fig. 5.7) 40 items are placed in a large sparse DAG structure where many items do not have any transitive relationship. The results align with expectations. On the Chain Dataset, DIDACT performs excellently, querying only 2-4 items (on the order of $\log_2 10$) across all levels of γ. This performance is possible only because DIDACT explicitly captures the transitive dependence relationships between items. On the same dataset, IRT is forced to query many more nodes and suffers from inaccurate probabilities (revealed as γ approaches 1). In contrast, both models achieve good (and nearly indistinguishable) performance on the Broad DAG dataset, where the conditional independence assumption of IRT is a reasonable simplification of the true structure of the data.

These results suggest two interesting findings: first, for closely-related questions, models that are able to capture the interdependence of test items have higher predictive power; second, this

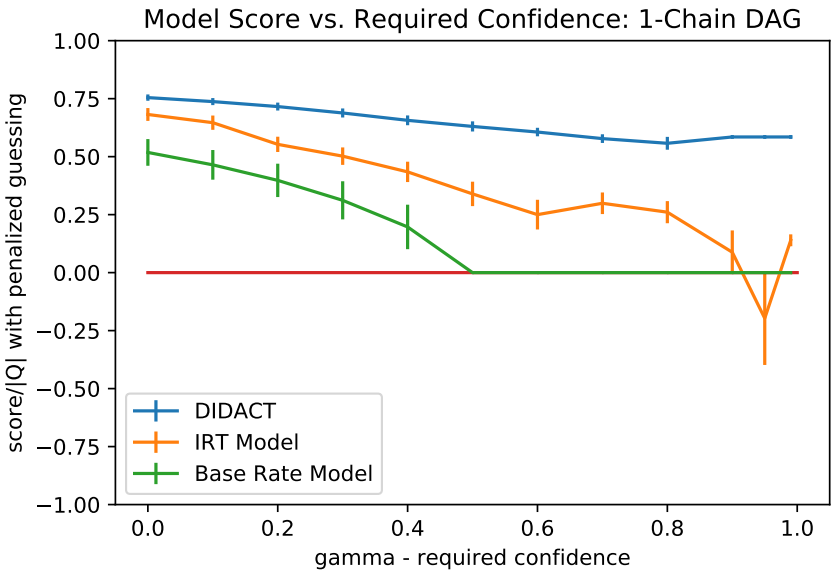

Figure 5.6: Adaptive Testing Performance subject to required confidence threshold γ on a synthetic dataset where dependencies form a single chain.

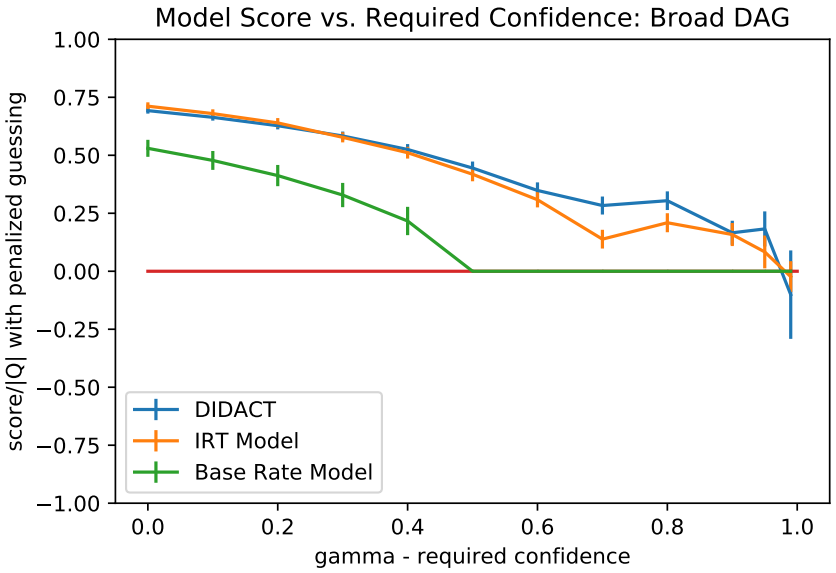

Figure 5.7: Adaptive Testing Performance subject to required confidence threshold γ on a synthetic dataset where dependencies form a broad but connected DAG.

phenomenon may not be discoverable from data based on comprehensive or broad assessments, because in these settings the two models are indistinguishable.

5.6 Evaluation on Information-Gain Criterion

We also evaluate DIDACT on the performance-gain objective from definition 3.3.1. In contrast to the γ-based objective above, where the accuracy threshold is fixed and the number of questions is variable, the information-gain objective we consider here fixes the number of questions and the accuracy (information gain) is variable. Comparatively, our algorithm does even better on this metric, likely because it de-emphasizes the effect of irrelevant questions on the score and doesn't penalize irreducible noise. Our discrete model has a much higher information capacity (due to its larger number of parameters) than the unidimensional IRT model, so it's perhaps unsurprising that IRT levels off sooner than DIDACT. These score differences may also reflect a philosophical difference in the treatment of observed student responses: to DIDACT, an observed response is fixed (once observed), and provides a noisy observation of a latent knowledge state; to IRT, an observed response is a sampled outcome based on latent parameters—that is, repeating the question (or a very similar question) may result in a different observation. This difference is fundamental in the interpretation of observations under each model.

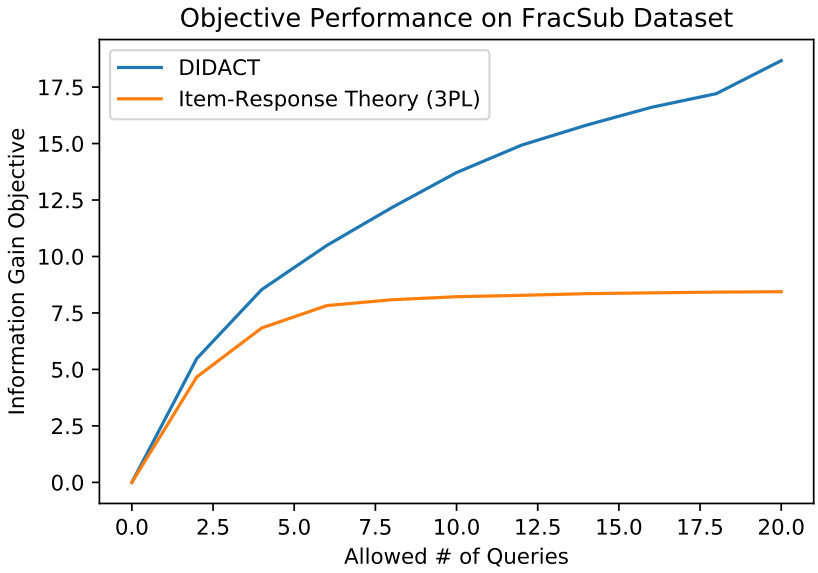

Figure 5.8: DIDACT significantly outperforms IRT on the information-gain objective for the FracSub dataset.

5.7 Leveraging Interdependent Models

Part of why IRT models (such as the Rasch model), have been so popular over the last decades is their auxiliary uses based on interpretation of the model. For example, questions can be ranked based on how well they fit the model defined by the other questions (a form of internal validity and the basis of measures like Cronbach's Alpha). Students can be evaluated. And, questions (and related topics) can be ordered by their difficulty, leading to a natural curriculum. The goal of this section is to illuminate how some of these use cases can benefit from modeling the interdependence of assessment items.

5.7.1 How to Make Exams More Reliable

Large assessments can often be made more reliable by removing questions that have little relevance to the rest of the exam. In the ideal of the adaptive test setting, the minimal number of questions are asked to accurately predict responses to the remaining questions, so a natural way of ranking items (represented as random variables X_i) is by the following:

$$R(X_i) = \sum_j I(X_i, X_j),$$

where $I(X_i, X_j)$ is the mutual information between X_i and X_j. Note that this score includes the amount of mutual information the variable has with itself, which is just the entropy of the random variable $H(X_i) = I(X_i, X_i)$. It slightly favors questions that are neither too easy nor too hard for most students.

Given a means of ranking questions, assessments can be designed subject to budget constraints for a particular γ. This goal can be accomplished by adding questions in order of decreasing rank until the mean score at γ begins to decrease.

5.7.2 How to Evaluate Students

Student abilities can be represented as a vector indicating whether the student has mastered each item. Given a dependency map, this vector space has a partial ordering that captures possible learning trajectories for each student. It also allows for fine-grained student diagnostics - perhaps the student isn't lacking practice, but specific prerequisite knowledge that would allow them to succeed. Although there are many possible ways to collapse the student mastery vector into a single grade or score, the fine-grained vector may hold more utility for practical classroom use.

5.7.3 How to Infer a Curriculum

Just as student ability vectors define a partial order over students, the dependency map defines a partial order over content. By the assumptions of the model, content appearing in the dependency map cannot be mastered before the content it is dependent on. Thus, all prerequisite topics should occur in a curriculum before the topic that depends on them.

5.8 Conclusion

This chapter provided empirical evidence that assessments involving closely related items are likely to benefit from interdependence modeling. To facilitate these experiments, a novel evaluation framework was introduced that explicitly navigates the tradeoff between data-efficiency and accuracy in adaptive testing. Additionally, a novel algorithm for interdependence modeling, DIDACT, was introduced, which achieves high performance while remaining computationally efficient. Finally, these results were connected to related educational problems, including assessment creation, adaptive pedagogy, and curriculum design. These results can be applied directly in future work expanding the use of dependency modeling in adaptive testing, which may be of particular use when assessment items come from nontraditional sources or the pool of items changes over time. Future work may consider how to extend these models to more general models of assessment than binary correct/incorrect items.

5.9 Challenges and Limitations

A significant challenge not fully addressed by this work is that unbiased collection of pairwise data may not lead to correct structure recovery. For example, in a long chain, two adjacent nodes (with a prerequisite relationship) will be labeled the same way the vast majority of the time, leading to the erroneous collapse of non-equivalent nodes. (See figs. 5.9 to 5.11.) The pairwise inference methods outlined above assume that the outcome of the two assessment items is not already determined by an external observation. If the structure were known, data could be filtered for datapoints where the pairwise relationship isn't determined by the other observations, but this path introduces a bootstrapping problem and a data-efficiency problem. Despite its state-of-the-art performance in the above experiments, DIDACT could be potentially improved to recover known structures more reliably.

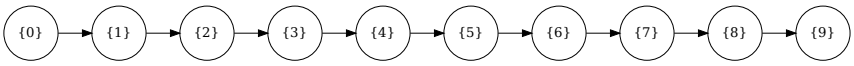

Figure 5.9: A latent DAG structure. Synthetic observations of student outcomes are generated with a small amount of noise.

This work is also limited experimentally. Although the FracSub and SAT datasets are common benchmarks for assessment and cognitive modeling studies, they are each relatively small (only a few hundred students) and each represent a fairly narrow (and standardized) slice of education. Although there is not an abundance of publicly available datasets for education, data collected from existing courses and subjects in a variety of settings could be used to further validate (or invalidate) these techniques.

Finally, there are some inherent barriers to the interpretability (or certainly, some non-intuitive

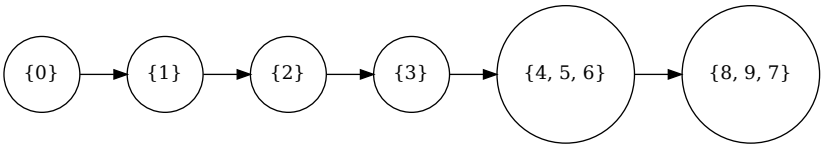

Figure 5.10: The structure recovered by DIDACT is imperfect.

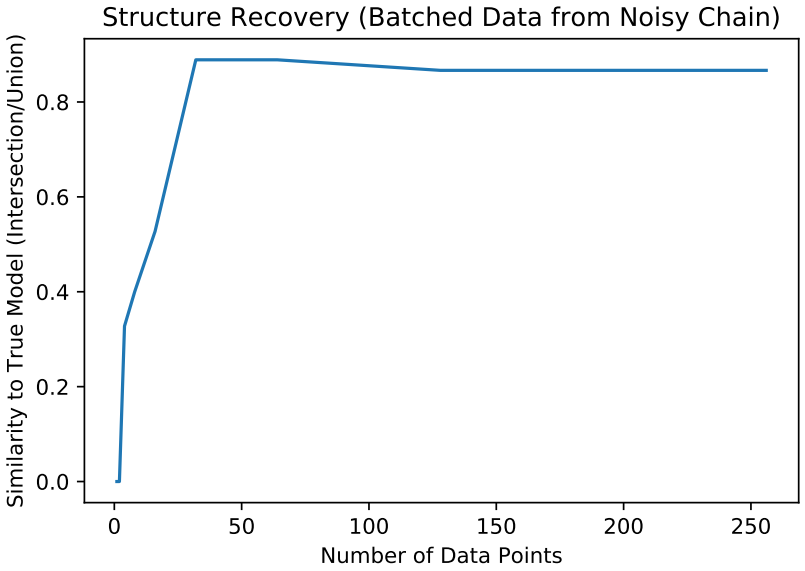

Figure 5.11: DIDACT quickly achieves an approximately correct structure, but may not converge to the exact latent structure. The y-axis is the commonly used "Intersection over Union" similarity metric, where the set of learned prerequisite relationships is compared with the complete set of true prerequisite relationships.

aspects) of the prerequisite maps discovered by DIDACT. For one thing, nodes in the prerequisite map represent equivalence sets over assessment items, which can lead to complicated dependencies between items requiring different subsets of a pool of skills to answer correctly. Many instructors may be more used to thinking about the dependencies between the factorized set of skills used (as evidenced by the use of the Q-matrix) than the relationships between individual assessment items. Thus, there may be a more compact interpretable structure than the prerequisite maps extracted by our algorithm.

For another thing, the statistical relationships between assessment items may reflect artifacts of the population being assessed, rather than semantic prerequisite relationships. For example, if all of the students are going through a course of instruction that artificially imposes a prerequisite relationship between otherwise unrelated topics (for example, a linear sequence where students cannot move on before mastering the previous topic), those spurious prerequisite relationships will be reflected in the data. Furthermore, if assessment items are collected sparsely over a very large educational domain, statistically-detected prerequisite relationships may be more reflective of assessment item "difficulty" or "complexity" than an explicit semantic relationship. For example, language ability and mathematics knowledge may be statistically related, even though it is *theoretically possible* to teach many math topics without the student knowing how to read. One would hope that as the density of assessment items increases relative to their breadth, many of these large-scale trends would be replaced with more semantically meaningful proximate relationships in the transitive reduction, but a much larger and broader educational dataset would be required in order to confirm that.

Chapter 6

Prerequisite Maps can be Efficiently Queried

Unfortunately, the assessment creation problem defined in Chapter 1 is computationally intractable. Even ignoring issues of learning probability distributions from data, planning the optimal sequence or combination of queries is computationally difficult. See Theorem 6.0.1.

Theorem 6.0.1. *Maximizing the objective given in Definition 3.3.1 subject to a maximum number of queries per student is NP-Hard, even if the underlying joint distribution of all the random variables is known.*

The proof of theorem 6.0.1 can be found in appendix C.

6.1 A Greedy Lower Bound

Since a polynomial-time algorithm to optimize the given assessment generation objective function is unlikely to exist, this section considers a slight modification that lower-bounds the objective and permits a myopic querying strategy.

Observe that when the predicted unconditioned distribution for an assessment item agrees with the underlying distribution, the objective function can be nicely telescoped across queries to a single student:

$$U(A,Q,S) = \mathop{\mathbb{E}}_{s_i \in S}\left[\sum_{t=1}^{k}\sum_{Q_j \in \hat{P}_{s_i}}\left(-\log \hat{P}_{s_i}[Q_j = s(Q_j)|\omega_{s_i,t-1}] + \log \hat{P}_{s_i}[Q_j = s(Q_j)|\omega_{s_i,t}]\right)\right].$$

This objective can be locally maximized by greedily maximizing the single-step gain in information:

$$U(A,Q,S,s_i,\Omega_{[1\to i-1]},\omega_{(s_i,t)},t) = \sum_{Q_j \in \hat{P}_{s_i}}\left(-\log \hat{P}_{s_i}[Q_j = s(Q_j)|\omega_{s_i,t-1}] + \log \hat{P}_{s_i}[Q_j = s(Q_j)|\omega_{s_i,t}]\right).$$

6.2 Interlude: Greedy Queries of Deterministic Prerequisite Maps

What reason is there to believe that a greedy strategy performs well relative to the optimal possible performance? Can optimal performance be achieved in special cases? This section considers a deterministic relaxation of the query optimization problem where the goal is to minimize the number of queries required to infer the outcome of every item.

This problem is of independent theoretical interest. There are some results for related problems, but this problem, to the author's knowledge, has not been studied in the literature. It is worth noting that acyclic directed graphs (DAGs) are isomorphic to partially-ordered sets (POSets). Partial Order Theory is of significant interest in computing and mathematics disciplines including distributed computing, concurrency theory, combinatorics, number theory, logic, and group theory.

6.2.1 The DAG-Partition Problem

Let $G = (V, E)$ be an acyclic directed graph consisting of a set of vertices V and set of directed edges E.

Define $p \to q$ to mean that $(p, q) \in E$, and define $p \xrightarrow{*} q$ to mean that p is a (transitive) ancestor of q. In other words,

$$p \xrightarrow{*} q \iff (p \to q) \vee \exists r : (p \to r) \wedge (r \xrightarrow{*} q) \tag{6.1}$$

Define the set of possible labelings

$$\mathcal{L} = \left\{ l : V \to \{F, T\} \mid (p \xrightarrow{*} q) \implies (l(q) \implies l(p)) \right\} \tag{6.2}$$

. In other words, a given labeling is a function that maps each vertex to True (left set) or False (right set) and is monotonic with respect to directed edge relationships — a True node cannot have a False parent.

Define the set of possible markings as

$$\mathcal{M} = \left\{ \mu_S : S \to \{F, T\} \mid (S \subseteq V) \wedge \left((p, q \in S) \implies \left((p \xrightarrow{*} q) \implies (\mu_S(q) \implies \mu_S(p)) \right) \right) \right\} \tag{6.3}$$

.

Then a query algorithm for solving the DAG partition problem on G is a function

$$A_G : \mathcal{M} \to V \tag{6.4}$$

that selects a next query node given the current markings. (Note that we will presently deal with the trivial termination condition for the algorithm, which is to stop querying when all the nodes' labels are known or can be inferred.)

Whenever a label is observed, it implies labels for either its transitive children (if the label is False) or for its transitive ancestors (if the label is True). We define the umbra function U on a

vertex v and label λ

$$U(v,\lambda) = \begin{cases} \left\{ p \mid v \xrightarrow{*} p \right\} & \text{if } \neg\lambda \\ \\ \left\{ p \mid p \xrightarrow{*} v \right\} & \text{if } \lambda \end{cases} \tag{6.5}$$

And we define the closure c of a marking as

$$c(\mu_S) = S \cup \bigcup_{v \in S} U(v, \mu_S(v)) \tag{6.6}$$

.

Define the marking sequence and query sequence from running A_G on G with labeling l

$$Q(A_G, l) = [v_1, v_2, ..., v_n] \tag{6.7}$$

where

$$Q(A_G, l)_1 = A_G(\mu_\varnothing) \tag{6.8}$$

$$Q(A_G, l)_k = A_G(\mu_{\{\bigcup_{i=1}^{k-1} v_i\}}) \tag{6.9}$$

and

$$c(\mu_{\{\bigcup_{i=1}^{n} v_i\}}) = V \tag{6.10}$$

Then define the objective performance measure $\mathcal{O}$ for A_G as:

$$\mathcal{O}(A_G) = \max_{l \in \mathcal{L}} \|Q(A_G, l)\| \tag{6.11}$$

6.2.2 Related Work

This DAG query problem may be thought of as a generalization of searching a sorted list. In particular, we are interested in partitioning the nodes in a DAG into a "left set" and "right set" such that all nodes in the left set are less than or equal to a given query point, and all the nodes in the right set are greater than the given query point. Our problem is unusual in that although not all pairs of nodes in the prerequisite DAG are comparable, our query point (the student) is comparable to all nodes in the graph (for any assessment item, we can get an observation of student correctness/incorrectness). This problem is relatively unstudied, although it is isomorphic to the problem of learning a monotonic predicate over a partially-ordered set (POSet), for which there is some prior work (Eiter et al., 2008). However, that prior work is primarily concerned with highly structured lattices (such as the subset lattice over a powerset of elements), and is thus inapplicable to the smaller but less structured DAGs that motivate this proposal. This problem is also, unfortunately, somewhat different from the prior work on sorting using a partially comparable set Daskalakis et al. (2011).

There is an alternate generalization of searching a sorted list, which is somewhat different from our problem: determine if a given query element exists in a DAG, given the ability to make queries (which may return "incomparable"). This latter problem has been studied in a paper that shows that

minimizing the number of queries is NP-hard in that setting (Carmo et al., 2004). However, the same paper shows that approximately optimal query strategies are possible in some fairly general domains.

This leaves a gap in current computer science theory that would be interesting and useful to fill. Several primary results are introduced in the following subsection. Note that common practice in the application domain is to simply query all nodes ($O(n)$ queries), leaving a large opportunity for practical improvement.

6.2.3 Main Results

Lemma 6.2.1 (Optimal Performance on a Disjoint Set)**.** *Let there be a disjoint set (also called an anti-chain) S of size w in graph G. Then for any algorithm, $\mathcal{O}(A_G) \geq w$.*

Proof. Consider the set of labelings where all ancestors of nodes in S are labeled T (so they do not imply anything about nodes in S), and all other nodes not in S are labeled F. Since the nodes in S are a disjoint set, there are no constraints on the labels of the w nodes, and thus 2^w labelings in this set. By a decision-tree argument, since each query has only two possible outcomes, it must take at least w queries to distinguish between these 2^w possibilities. □

Lemma 6.2.2 (Optimal Performance on a Chain)**.** *Let there be a chain graph of length n. For any algorithm, $\mathcal{O}(A_G) \geq \lceil \log_2(n+1) \rceil$.*

Proof. The argument is similar to the previous lemma. The "boundary" between the T set and the F set can only occur in one of $n+1$ possible locations (in between two nodes, or at one of the two end-points), so there are $n+1$ possible labelings. The height of a decision tree implementing a querying algorithm must be at least $\lceil \log_2(n+1) \rceil$. □

It is an established fact that acyclic directed graphs can be decomposed in polynomial time into strongly-ordered chains that cover the set of nodes and preserve the ordered relationship between items Ikiz and Garg (2004). A well established theorem of Dilworth (1950) says that the minimum-cardinality chain decomposition of a DAG has cardinality equal to the largest cardinality of an antichain in that DAG. This is a measure of the "width" of a DAG (not to be confused with DAG-width, which is a different thing), and will typically be represented with the letter w.

This suggests a straight-forward algorithm (not necessarily optimal) for the DAG partition problem: decompose the DAG into chains, and then perform binary search on each chain. This algorithm will be called "chain search" (C_G). Another straight-forward algorithm is a myopic algorithm that simply maximizes the worst-case number of newly-labeled nodes with each query, with random tie-breaking. That algorithm will simply be called "greedy" G_G.

Lemma 6.2.3 (Chain Search and Greedy are strictly suboptimal)**.** $\exists G, \exists A_G : \mathcal{O}(A_G) < \mathcal{O}(C_G) \wedge \mathcal{O}(A_G) < \mathcal{O}(G_G)$

Theorem 6.2.4 (Query Complexity of Chain Decomposition Binary Search)**.** $\mathcal{O}(C_G) \leq w\lceil \log_2(n+1) \rceil$

Proof. The proof is straightforward - the cost is w times the maximum cost of querying a single chain, which has length at most n. □

Corollary 6.2.4.1 (Bounded Suboptimality of Chain Decomposition Binary Search)**.** *For any G and any algorithm A_G, $\frac{\mathcal{O}(C_G)}{\mathcal{O}(A_G)}$ is $O(\log n)$*

Proof. This follows easily from lemma 6.2.1 and theorem 6.2.4. □

Theorem 6.2.5 (Query Complexity of Greedy Querying)**.** $\mathcal{O}(C_G) \leq 2w\lceil \ln n \rceil$

The proof of theorem 6.2.5 is in appendix D.

Corollary 6.2.5.1 (Bounded Suboptimality of Greedy Querying)**.** *For any G and any algorithm A_G, $\frac{\mathcal{O}(G_G)}{\mathcal{O}(A_G)}$ is $O(\log n)$*

Proof. This follows easily from lemma 6.2.1 and theorem 6.2.5. □

6.3 Exploration-Exploitation Tradeoffs

As an assessment is being built, data is being actively collected from a set of students, but the students are also being assessed. How should an assessment algorithm balance the short term goal of accurately assessing the current student (using assessment items it knows to be informative) with the long term goal of evaluating additional assessment items and potentially improving its assessment strategy? This is a classic exploration/exploitation tradeoff, and closely resembles a multi-armed bandit problem. The approach here takes inspiration from Thompson Sampling, a well-established algorithm that solves the multi-armed bandit problem with near-optimal regret bounds.

The observed item outcomes can be tabulated individually and pair-wise and treated as a prior in a beta distribution or dirichlet distribution (depending on the number of outcomes). The algorithm used here involves sampling a prerequisite map structure by first sampling probabilities from the pairwise dirichlet distributions and then running the standard DIDACT construction algorithm using those probabilities. Then, the expected single-step information-gain reward can be estimated from the structure, and the highest-value item selected to query. The item outcome will be recorded, and the query process repeated using the same model.

This process has many of the qualitative benefits of Thompson sampling. For example, statistically uncertain relationships are more likely to be left out of a sampled DIDACT model, providing additional incentive to query both items. Conversely, statistically reliable relationships will be exploited more frequently, leading to better objective performance.

Unfortunately, this algorithm does not lend itself to easy extension of the existing guarantees for Thompson sampling. In particular, the collected pairwise statistics are biased by the developing prerequisite map structure, which does not cause problems in practice, but does render the existing regret bounds for Thompson sampling inapplicable.

6.4 Challenges and Limitations

Although the near-optimal proof for a greedy query strategy is enormously validating for the DIDACT algorithm presented in the previous chapter, there are still many open questions around the DAG Partition Problem and the application to adaptive testing. For one thing, we do not currently have a computational complexity result for the DAG Partition Problem, although NP-hardness results for similar problems certainly suggest exact optimization may be hard. While moot in the context of the larger performance objective (which has already been shown to be NP-hard in its own right), it is an open question with respect to the theory in the deterministic case.

It is also unclear whether our near-optimality bounds extend to the stochastic setting where observations are noisy. While the earlier experiments showed that it worked well in practice, there is still a gap in the theory.

Finally, it is uncertain whether or not the theoretical guarantees of Thompson Sampling apply in our setting, where we are estimating entropy from previous observations, and may have an effectively infinite number of arms. Empirical results suggest that Thompson Sampling performs well in these settings (see fig. 6.1), but the theoretical guarantees are difficult to generalize.

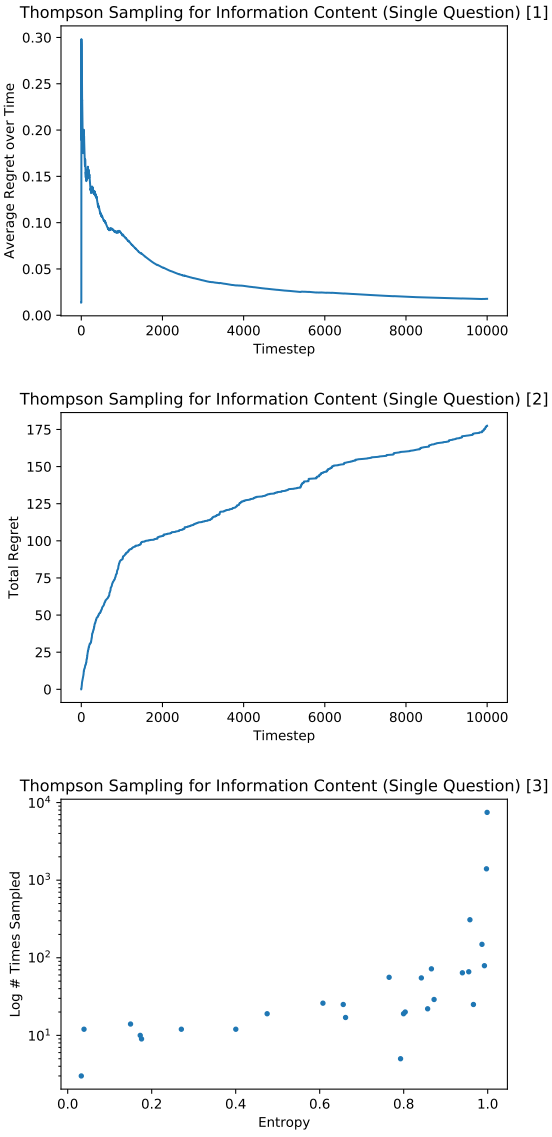

Figure 6.1: Thompson Sampling for information content quickly converges to near-optimal information gain.

Chapter 7

Prerequisite Maps can be Constructed from Student-Sourced Questions

Student conceptions and learning benefit from a diversity of assessment instruments. In an ideal world, educators would have access to a concept inventory (CI) for each topic they cover. A CI Hestenes et al. (1992) is a *validated*, *robust*, and *interpretable* instrument that pinpoints student conceptions. It can examine the knowledge of students coming into a course Henderson (2002), measure their change across the course Henderson (2002), and help identify misunderstandings on the spot, e.g., in conjunction with clickers Simon et al. (2010). Therefore, there is value to having a large number of CIs across the whole spectrum of educational topics.

Unfortunately, generating a CI or similar instrument is painstaking. Furthermore, a CI for a certain topic may not be very useful in a particular instructor's setting due to differences of covered material, prior preparation, choice of notation or language, and numerous other factors. Finally, considering computer science education specifically, the number of actual CIs is fairly small Kaczmarczyk et al. (2010) Taylor et al. (2014) and are relatively concentrated in introductory computing, failing to cover large parts of the field. Thus, CIs appear to be difficult-to-attain holy grails.

Absent such a rigorous instrument, an instructor might have to make do with a quiz of their design, which lacks any of the properties listed above. Due to the expert blind spot Nathan et al. (2001) the quiz may fail to cover some topics. It may also miss many mistakes that students have, may not necessarily clearly line up mistakes with misconceptions, and so on.

In this chapter we attempt to find a happy middle between these extremes. We define the process of *Adaptive Tool-Driven Conception Generation* (ATCG), which uses a pair of techniques in conjunction:

crowdsourcing to obtain a large number of contributors with diverse views, but using students in a class; and,

1. **experts** set the scope of the assessment (using a **Delphi Process** to choose important and difficult topics)
2. students articulate mental models and experts identify misconceptions (through **interviews** to find interpretations for student responses)
3. **experts** develop questions (to distinguish mental models)
4. **experts** validate the assessment (using statistical and **qualitative analysis** across **several trial administrations** to iterate towards robust and validated questions)

Figure 7.1: A traditional process for creating new assessments Goldman et al. (2008) based on recent practice in several disciplines Evans et al. (2003).

1. **experts** write a prompt setting the scope for contributions
2. students contribute questions
3. students contribute answers and rationales (by answering other students' questions and providing a brief justification)
4. a machine-learning algorithm prioritizes questions with informative response distributions
5. **experts** select questions for relevance (thus validating the questions)
6. **experts** identify misconceptions (by reviewing student rationales for each answer and interpreting the responses)

Figure 7.2: The novel class-sourcing process reduces the burden of expert labor.

machine learning to efficiently infer which contributions are most robust.

It is instructive to compare the ATCG process with a traditional CI process. Both are shown in fig. 7.1, with CI on the left and ATCG on the right. The CI steps are clearly much more heavy-weight, resulting in valuable instruments generated at great human (especially expert) expense. The ATCG steps are cheap and benefit from automation. They do not offer the same guarantees, but can be applied in many settings easily.

This naturally raises the question, what is the quality of the results from ATCG? We present a first set of results in this direction. First, we describe a tool, quizi.us, that implements ATCG (section 7.2). We have used quizi.us in three settings, two of which have nothing remotely resembling a computing CI: higher-order functional programming (which is used in many contexts, and even supported by the Snap! block programming language), and quantifier use in an upper-level course on mathematical logic in computer science. Due to space limitations we will focus on the third: generating an instrument for arrays in a Java CS2 course. We pick this because we can compare the result against an existing CI Kaczmarczyk et al. (2010). Perhaps somewhat surprisingly, the ATCG output fares quite positively (section 7.3), showing that this process has potential and is worth studying further.

7.1 Related Work

Concept Inventories in Computer Science Taylor et al. (2014) provide a survey of the concept inventory efforts in computer science. They acknowledge that these are expensive to run due to the strong validation demands. In this chapter we focus specifically on the instrument of Kaczmarczyk et al. (2010), based on extensive student interviews on topics validated by experts as important and difficult for beginning programming students, resulting in 16 questions, with most referencing snippets of Java code. We focus on the array portion of it, which is represented as four questions in Figure 7.3 (reproduced here with permission).

E1. What should the values of x and y be, in order to fill all elements of the following array with values of -1? [Distractors are based on thinking that arrays start at index 1 and/or a for loop stepping through an array will need to go until ¡ the array's length - 1]

```
int[] myArray = new int[10];
int i;
int x;
int y;
for (i = x; i < y; i++) {
    myArray[i] = -1;
}
```

a. x = 1, y = the length of myArray [1x]

b. **x = 0, y = the length of myArray** [109x]

c. x = 1, y = the length of myArray - 1 [0x]

d. x = 0, y = the length of myArray - 1 [22x]

E2. How many elements are in myArray? [Distractors are based on thinking that an array's length is off by 1 or equal to the number of letters in the array's name]

a. 7 [0x]

b. 9 [1x]

c. **10** [124x]

d. 11 [0x]

E3. Which of the following answers most accurately describes the parts of the declaration of myArray? [Distractors are based on thinking the type of an array is just an array or a general object and/or declaring the length of an array instead sets the first element or all elements in the array to the value of the length] [remove answer e for non-Java languages]

```
int[] myArray = new int[5];
```

a. **type is an integer array, length is 5** [124x]

b. type is an array, each element has a value of 5 [0x]

c. type is an integer array, each element has a value of 5 [0x]

d. type is an integer array, the value of the first element is 5 [0x]

e. type is an object array, the constructor is passed a value of 5 [2x]

E4. Which of the following answers most accurately describes the memory allocation of myArray in E3? [Distractors are based on thinking memory is not allocated for the elements in the array or only the 1st element or plus an extra element]

```
int[] myArray = new int[5];
```

a. memory is allocated for 6 spaces [7x]

b. **memory is allocated for 5 spaces** [120x]

c. no memory is allocated [0x]

d. memory is only allocated for the first element [0x]

Figure 7.3: Four questions from the expert instrument. Out of the 16 total questions, these related somehow to arrays. Questions, answers, and interpretations (mental models) are from Kaczmarczyk and collaborators (private correspondence, extending Kaczmarczyk et al. (2010)). Answers are followed by the number of *our* respondents who selected that answer. Note that not all respondents answered all questions.

Contributing Student Pedagogy ATCG can be viewed as a form of Contributing Student Pedagogy (CSP) Hamer et al. (2008b). This literature suggests that writing questions and answering other students' questions has a positive effect on course outcomes. ATCG has most of the attributes of a CSP, though simply generating an instrument provides only limited opportunity to "value the contribution of others", which for now has to be done through other channels. (ATCG also shares elements with *action research* Brydon-Miller et al. (2003), including a focus on action and reflection, and students as the main driver of scope.)

quizi.us shares much with the CSP system PeerWise Denny et al. (2008), but:

- When answering questions in our system, all answers are free-response so that the student answers will be unbiased.
- Rather than using student ratings of problem difficulty and usefulness, we empirically measure the informativeness of each question by using machine learning to examine the diversity of answers and optimize answer collection.
- After quizi.us has run, an expert curates the most informative contributed questions for the purpose of creating an instrument, whereas no such engagement is necessary with PeerWise, where the activity is the goal in itself.

Bandit Processes and Machine Learning To construct a quality instrument, we need robust questions. But measuring robustness accurately requires collecting responses from many students for that question. Additionally, out of consideration both for the students and the quality of data, we would like to not waste students' time on questions that are not likely to be interesting. One approach is to continually estimate the likelihood that a question is interesting, and prioritize data collection for questions that are already suspected to produce interesting responses. But we should not prematurely rule out a question because the first few responses were all the same (suggesting the question is uninteresting).

The problem of choosing an action (*a question*) out of many, where taking the action produces some reward (*we might get an interesting response*) and also gives us more information about the action (*we have a better estimate of how interesting the question is*), is a well-studied reinforcement learning problem called the Multi-Armed Bandit. Efficient solutions have been derived that are guaranteed to converge to selecting the best action as more data are acquired (Auer et al. (2002), for example), which we use in quizi.us.

7.2 The quizi.us Tool

To facilitate ATCG, we created the Web-based quizi.us tool. It allows students to join classes, participate in quiz activities, write questions, answer questions with rationales, and review question and answer distributions after the quiz ends. quizi.us also allows instructors to contribute their own

questions, to review the distribution of student responses, and to automatically grade students based on participation and late-day use.

In addition to streamlining the data collection process, the tool implements the critical machine learning step to filter out informative questions from the large pool, spending as few student answers as possible on questions that are not very informative. quizi.us models picking questions for students to answer as a many-armed bandit problem (a multi-armed bandit with far more actions available than time to test them all). We define the reward (the objective we are seeking to maximize) for the bandit problem as the sample-specific information (point-wise entropy) of duplicated responses (first-time/unique responses have reward 0). We believe this is a good proxy for questions that elicit diverse student mental models (section 7.3.5). Note that the objective of the ATCG method is to identify misconceptions for the purpose of evaluating pedagogical efficacy or tailoring instruction, not to place students on a scale of ability, as in item response theory Lord (2012). This necessitates a different measure of question reliability than is typically used for so-called high-stakes assessments.

Ordinarily, upper-confidence-bound algorithms (what we are using to solve the multi-armed bandit problem) converge to selecting the single best option. Since we are trying to find not the best question but rather a small set of questions, we select from among the top $\sqrt{n}$ questions, where n is the number of students who have used the system so far.

One difficulty is that students may write the same answer in different formats (e.g., "1000" versus "1,000"). We found it annoying to manually cluster these. Therefore, after students answer a question, they are shown previous representative answers and asked whether theirs is the same as any of these. In effect, we ask the students to cluster their answers themselves. This is easy for them, and in practice catches most duplicates. Of course, on seeing the previous answers, students may change their response (e.g., if they feel they were mistaken). Thus, quizi.us also lets them explicitly mark that they are changing their answer. This switch provides a very useful signal about student understanding that we have not yet exploited.

When the student process is done, the expert is given: a collection of questions; for each question, a collection of answers; for each answer, a collection of rationales for that answer. The expert then selects the questions that appear most interesting to assemble into an instrument. Examining the rationales gives insight into the mental models responsible for the student responses. In practice, we find that most answers arise for the same reason, but a careful reading finds a few other "outlier" reasons. If ATCG is being used to help build a CI, such questions would have to be modified so that different (mis)conceptions do not result in the same answer.

The ATCG protocol has a bootstrapping problem: What questions do the first students answer? quizi.us permits multiple options: let them simply finish early, or use pre-populated questions. The latter may come from various sources: instructor-provided, from an existing CI, or even from previous iterations of using quizi.us (e.g., if the process is re-run across years—which is what we did in our study in section 7.3).

7.3 A Comparative Study

We now compare an ATCG-generated instrument for CS2 Java arrays (the *generated* instrument), against Kaczmarczyk et al.'s CI (the *expert* instrument). G1-G6 in fig. 7.4 show the student-generated questions. At a high level, we find:

- The generated instrument contains most of the same topics and distractors as the expert instrument.
- The generated instrument misses some of the content of the expert instrument; we conjecture why this is so.
- The generated instrument contains array use topics—and distractors for them, along with explanations of misconceptions—*not* included in the expert instrument.

While performing this comparison, we also identify possible points of confusion in the expert instrument.

7.3.1 The Study Population and Iteration

We generated this instrument from two iterations of a CS2 class at a mid-size competitive private research US university. Although the course covered algorithms and data structures typical of CS2, it was also a first course using Java and a first course with object-orientation (the students had previously studied functional programming). Each time, the study was run midway through the course, after students had completed a lab assignment on each of objects and arrays. Just over 120 students participated in each iteration of the study, and we note (based on results from an entry survey) that less than half of these students had high school programming experience in Java.

The study was structured as an online homework assignment, and students were given a week to complete it. The first time the study was run, students were required to write one question (consisting of a short program in Java whose printed output demonstrated something they considered interesting about arrays), then to answer 15 questions written by other students in free-response style (to predict the printed output of the written programs without running them). They were also asked to answer the 4 array-related questions from the expert instrument. Figure 7.3 shows their counts for each answer in brackets; note that many answers were chosen by no students.

In the second iteration, we seeded the system with contributions from the previous year. We note that we observed qualitatively similar distributions of mental models between the two years. To reduce student effort, we made question submission optional (23 out of 128 students submitted one voluntarily anyway). Because of improvements to the question allocation algorithm and a tighter semester schedule, we only asked them to answer 10 questions. Section 7.3.6 gives more results on engagement and student time. *All* questions and responses in fig. 7.4 are student-generated; the responses are all from the second iteration of the study.

The provided Position class could be used freely:

```
public class Position {
    public double x = 0, y = 0;
    Position(double x, double y) {
        this.x = x;
        this.y = y;
    }
}
```

G1.

```
public class Interesting {
    public static void main(String[] args) {
        int[][][][][] arr = {{{{{1, 2}, {3, 4}}, {{5, 6}, {7, 8}}},
                             {{{9, 10}, {11, 12}}, {{13, 14}, {15, 16}}}},
                           {{{{17, 18}, {19, 20}}, {{21, 22}, {23, 24}}},
                             {{{25, 26}, {27, 28}}, {{29, 30}, {31, 32}}}}};
        System.out.println(arr[1][1][1][1][1]);
    }

}
```

a. 32 [24x]

b. 1 [9x]

c. 16 [4x]

d. `{3,4} {11,12}, {19,20}, {27, 28}` [2x]

e. `{{{{1, 2}, {3, 4}}, {{5, 6}, {7, 8}}}`
`{{{{1, 2}, {3, 4}}, {{5, 6}, {7, 8}}}`
`{{{{1, 2}, {3, 4}}, {{5, 6}, {7, 8}}}`
`{{{{1, 2}, {3, 4}}, {{5, 6}, {7, 8}}}`
`{{{{1, 2}, {3, 4}}, {{5, 6}, {7, 8}}}`
[1x]

f. illegible answer in console [3x]

g. other [10x, including 30, 28, 12, 14, and 31]

G2.

```
public class Interesting {
    public static void main(String[] args) {
        int x = 3;
        int[] test = {x,1,1,1};
        x = 1;
        System.out.println(test[0]);
    }
}
```

a. 3 [42x]

b. 1 [8x]

G3.

```
public class Interesting {
    public static String posString(Position p) {
        return "(" + p.x + ", " + p.y + ")";
    }
    public static void main(String[] args) {
        Position[] whatIsJava = new Position[5];
        Position[] otherArray = whatIsJava;
        otherArray[4] = new Position(5.5, 2.2);
        System.out.println(posString(whatIsJava[4]));
    }
}
```

a. (5.5, 2.2) [23x]

b. null [14x]

c. error [10x]

d. (0, 0) [7x]

e. other [2x]

G4.

```
public class Interesting {
    public static void reverse(int[] input) {
        int lastPlace = input.length - 1;
        int middlePlace = input.length / 2;
        for (int x = 0; x <= middlePlace; x++) {
            int temp = input[x];
            input[x] = input[lastPlace - x];
            input[lastPlace - x] = temp;
        }
    }
    public static void main(String[] args) {
        int[] arr = { 2, 16, 8, 42, 89 };
        reverse(arr);
        System.out.println(arr);
    }

}
```

a. [89, 42, 8, 16, 2] [24x]

b. Some memory address [19x]

c. `{89, 42, 8, 16, 2}` [7x]

d. other [4x]

G5.

```
public class Interesting {
    public static void main(String[] args) {
        int[] first = {1, 2, 3, 4};
        int[] second = {1, 2, 3, 4};
        System.out.println(first==second);
    }
}
```

a. false [31x]

b. true [11x]

c. error [10x]

G6.

```
public class ArraysExample {
  public static void main(String[] args) {

    Position[] array1 = new Position[2];
    Position[] array2 = new Position[2];

    Position posn1 = new Position(2.5, 3.5);
    Position posn2 = new Position(2.5, 3.5);

    array1[0] = posn1;
    array1[1] = posn2;

    array2[0] = posn2;
    array2[1] = posn1;

    System.out.println(array1.equals(array2));
  }
}
```

a. False [39x]

b. true [10x]

c. Error [2x]

d. other [3x]

G7. [Note that quizi.us ranked this question as having low information (limited diversity in answers). We include it here for comparison with answers to G1.]

```
public class Interesting {
    public static void main(String[] args) {
      try {
       Integer[] intArray = new Integer[] {1, 2, 3, 4, 5};
       System.out.println(intArray[5]);
        // print out the 5th element of the array
      } catch (ArrayIndexOutOfBoundsException e) {
       System.out.println("intArray does not contain an element at index 5");
      }
    }
}
```

a. "intArray does not contain an element at index 5" [53x]

b. Error [2x]

Figure 7.4: A selection from the most informative questions generated using ATCG. Each question is a program, whose output students were asked to predict. Each answer is followed by the number of respondents who agreed with it. Some answers are combined as "other" for brevity.

7.3.2 Similar Questions — Indexing and Length

To begin, we note that both the generated and the expert questions contain questions involving indexing (is Java 1-indexed or 0-indexed) and array length (E1 and E2 in fig. 7.3 (expert instrument) and G1, G2, G4, and G7 in fig. 7.4 (generated instrument)). We discuss 1-indexed misconceptions quantitatively in section 7.3.6.

7.3.3 Missing Questions — Memory and Type

Although there is a diversity of questions in the generated instrument, we note that there are no questions about memory allocation or type of an array object similar to E3 and E4 (fig. 7.3). We conjecture that this was precluded by the prompt used in this study, which required students to write a short Java program whose output would be printed. We consider it likely that most CS2 students do not have access to the reflection capabilities necessary to phrase such questions as Java programs. Perhaps if given a different prompt, students might write questions regarding type or memory allocation, but it is also possible that a less-structured activity would be harder for students. Thus, an open-ended CI interview offers possibilities that a closed, computer-based process may make more difficult.

7.3.4 Novel Questions — Assignment, Aliasing, and Equality

Finally, we note that there are a number of concepts tested by the generated instrument that are not present in the expert instrument. For example, G1 (fig. 7.4) involves nested arrays, G3 tests aliasing after assignment of one array to another, G4 involves mutation of an array passed to a method, and G5 and G6 test array equality. These are all concepts that are not addressed by the expert instrument. The authors were surprised at some of the things students thought to write questions

about, and we conjecture that student-authored questions may help to get beyond expert blind spots. In particular, G2 (which reveals a mutation and aliasing misconception—this is discussed in section 7.3.6) is not a question that occurred to us.

7.3.5 Novel Mental Models

This study allowed us to identify new mental models related to multiple indexing, equality comparison, and initialization of arrays. Additionally, by analyzing over a hundred student responses to expert instrument question E4 (fig. 7.3), we identified a new mental model relative to those previously identified: one student believed that 6 spaces were allocated for a `new int [5]`, because they were aware that Java caches the length of an array as an integer, and reasoned that that was allocated in addition to the 5 spaces for the 5 elements of the array. Although it is a rare mental model in our population, this suggests a refinement to that question. This mental model may have been lost in the much smaller sample size for which it is feasible to conduct extensive interviews.

7.3.6 Quantitative Analysis

We enumerate some salient points:

1. The expert instrument (fig. 7.3) is able to detect up to 12 misconceptions (either or both of 2 for E1, 3 for E2, 4 for E3, and 3 for E4) related to arrays. Our population exhibited a ceiling effect on the expert instrument, however, as only 5 of those misconceptions appeared in our population, and the majority of students got all the CI questions correct. (Figure 7.3 contains detailed incidence counts.) The generated instrument (considering only programs 1-6, as presented in fig. 7.4) detected 13 misconceptions in our student population.

2. The top-ranking generated questions are ones that produce a spread of student responses. Although this spread is not always due to a difference in mental models (G4 in fig. 7.4), some questions capture misconceptions in a nuanced way. In particular, on G1 on the generated instrument, 9 out of the 53 students exhibited a 1-indexing misconception (answer b). Interestingly, this 1-indexing misconception did not occur on a student-generated question involving a 1-dimensional array (G7), and on E1 from the expert instrument (fig. 7.3), only one student chose an answer consistent with a 1-indexing misconception. This is likely an instance of the well-known phenomenon of fragile student knowledge Perkins and Martin (1986).

3. For our student population, the generated questions detected misconceptions more frequently than the CI questions. There are many possible reasons for this, including differences in student population referenced during assessment creation. We also note that we do find interesting mental models in answers that claim a program is erroneous. We speculate this may not happen as frequently if students believe the programs were written by an expert.

4. We measured the amount of time each student spent on the question authoring page and on the answer writing page, which is a reasonable proxy for student time, although some large

outliers (one answer took 8.8 hours) suggest that students may have left the page open while doing other things. The median time to write a question was just over 11 minutes, and the median time to answer a question was 1 minute and 17 seconds. Furthermore, just over 30% of our students voluntarily answered more questions than required to complete the homework assignment—some by a significant margin, which suggests some of them may even have found this process educational or fun.

Across all students, about 90 hours of student time were used in total. About 8 expert hours over the course of a week were used to design and refine the protocols for the study, and about 15 expert hours over the course of two weeks were spent on selecting the relevant questions and interpreting student rationales. In total wall clock time, the second iteration of this study took 4 weeks to run end-to-end, which is substantially faster than any CI-creation process.

7.4 Application to a New Subject: Linear Temporal Logic

Although comparison to an established standard is a valuable scientific endeavor, replacing an existing instrument is of less practical value than creating one for an entirely new topic. To that end, we applied the ATCG process to create an instrument to detect learner misconceptions of Linear Temporal Logic (LTL), an extension of first-order logic with operators that quantify expressions over an infinite sequence of states. LTL was an ideal topic of study due to: its novelty in the education-research literature; its unambiguous syntax; its growing importance in industrial applications of formal methods and logical specifications for programs.

Students in a class on "Logic for Systems" were prompted to provide a brief English description of an LTL specification, which other students were then asked to "translate" into an LTL formula. Students were asked to provide answers to 10 such prompts, and the task was administered as a homework assignment in that class.

The results are very preliminary, and a follow-up study could provide greater statistical validity to the interpretations that follow. That said, we were able to identify quite a few misconceptions in our initial study, and we believe that these insights can inform pedagogical design around this difficult topic.

The observed misconceptions include:

Implicit Global (believing an expression without a temporal quantifier applied globally)

Next implies converse in the current step

Implication also means Next

Next means Finally/Eventually

Retroactive Global (believing that Global applies to past states in addition to the present and future)

Temporal operators can't apply to composite expressions

Next "spreads" the expression (believing that multiple applications of Next means the expressions holds for that many time steps)

As illustration, see examples of two of these misconceptions in figs. 7.5 and 7.6.

Correct Expression	Misconception Expression
G(x1 -> X(!x1 U x2))	x1 -> X(!x1) U x2 *
G(x1 -> (X(!x1) and X(X(x1)))	x1 -> X(!x1) and X(X(x1))
G(x1 -> X(!x1 U x2))	x1 -> X(!x1) U F(x2) *
F(x1) and G(!x2)	(!x2) and F(x1)
G(x1 -> (F(x2) or !X(x3)))	x1 -> (F(x2) or X(!x3))
F(x1) and G(x1 -> !F(x1))	!x1 U x1 and (x1 -> !F(x1))
G (!(x1 and X(x1) and X(X(x1))))	!(x1 and X(x1) and X(X(x1)))

Figure 7.5: 7 of the top 12 questions in our study detected the implicit global misconception. (Answers with a red star exhibited other issues as well.)

Correct Expression	Misconception Expression
F(x1 -> X(G(x2)))	F(x1 -> G(x2))
G(x1 -> X(!x1Ux2))	G(x1 -> (!x1)Ux2)
G(x1 -> X(!x1 U x2))	G(x1 -> !x1 U x2)
G(x1 -> (F(x2) or !X(x3)))	x1 -> F(x2) or !x3 *

Figure 7.6: 4 of the top 12 questions in our study detected the "Implication also means Next" misconception. (Answers with a red star exhibited other issues as well.)

7.5 Discussion

It is not clear whether it is better to ask students to write questions before, during, or after answering the questions of other students. Having them write questions before answering any occasionally results in questions that are nearly isomorphic, although the written questions are unbiased by the work of other students. Having them write questions during or after allows students to benefit from the examples they have seen, and deliberately write different questions, but it may artificially constrain their responses based on what they have seen; more studies will need to be done to evaluate the relative strength of these effects quantitatively.

We found that some students submitted questions unrelated to arrays in Java, although they did produce a broad spread of answers. (In particular, a question about object equality comparison left even the authors searching carefully through the Java language specification to understand the correct answer.) Although the tool could be altered to allow students to flag off-topic questions, the

instructor may wish to keep those questions that reveal a spread of mental models, even if they are off-topic.

Though we did not encounter instances of abusive use, it is also important for users to be able to flag offensive content (as they can in quizi.us). Malicious users might write an abusive answer that others would have to read; more subtly, they might submit a program whose output is offensive, which everyone else would then be forced to type. Working in a graded course, rather than on the open Internet, probably helped us avoid such incidents.

7.6 Conclusion

This chapter presents ATCG as an intriguing mid-point, and potential sweet spot, between the heavy-weight CI process and entirely instructor-generated evaluation instruments. Augmenting instructor work with that of students can reveal a rich set of models, working around expert blind-spots. In a comparative study, we find that *purely* student-generated questions and answers, with relatively little effort (and with some of the auxiliary benefits of CSP), can compare very favorably to ones generated with considerable effort by a team of experts. Our tool, quizi.us, helps run the ATCG process, presenting a Web interface and embodying a machine learning algorithm. We have already begun to apply ATCG in several other contexts, and we feel this process should be applied to all of computer science (indeed, quizi.us is not even computing-specific), especially in areas where it may take a very long time and considerable expense to cover with a CI.

7.7 Challenges and Limitations

Although the work presented in this chapter is very promising, there are some significant difficulties in working with students (or humans in general) that leave some future work to be done. For one thing, computer science is a convenient domain to consider because of its artificiality—determining the "correct" response is usually straightforward and unambiguous. But, even computer science frequently broaches more subjective topics, like ethics, design, interpretation and intuition. The ability to statistically evaluate large numbers of non-expert assessment items requires the ability to automatically group similar answers, which requires some forethought in the design of the assessment scope and answer collation process for students. Even then, for a student to identify equivalent answers may require domain expertise (as in the LTL study), producing variation in student responses that is orthogonal to the content knowledge being nominally assessed by the prompt.

Even outside of those considerations, there are limitations to the generality and applicability of this work. First, these techniques require fairly large numbers of participants to produce good results. All of the experiments in this chapter were conducted on classes with at least 100 students, which includes many introductory university classes, but may not apply to more advanced topics or classes in other institutions which may be smaller (or, at least, might require aggregation across multiple instances of the class). Second, these live experiments with students were all conducted in

the Computer Science Department at Brown University (as have any other pilot experiments to date not reported here), so more experimentation is necessary to determine how well these results will generalize to other populations or other subjects. Third, unexpected challenges have arisen in all of our experiments with people (some relatively minor), so some iteration may be necessary to adapt this process even to a similar context.

Chapter 8

Conclusions

This work presented in this dissertation documents a major step forward in methods for the generation and administration of assessments. This work began with a quantitative objective for assessment generation and then showed that prerequisite maps are a useful model for solving this problem. Efficient algorithms for learning and querying these models were presented and evaluated on real educational data. Finally, techniques for sourcing questions and responses from students were presented, as well as proof-of-concept experiments generating assessments in computing education contexts. This chapter will elaborate some of the implications of this work as well as its limitations.

8.1 A New Theory of Testing

Education has inherited a theory of testing (and its only statistical models for assessment) from psychometric theory, and not all of the assumptions translate well across the two contexts. For example, in psychometric theory, unidimensionality of the model is a desirable trait. In education, an assumption of a unidimensional latent skill is a poor model for knowledge, which can be spread non-monotonically over many possible subjects and topics. This dissertation has demonstrated that in the context of topical assessments, Item Response Theory does not have the same predictive power as prerequisite map models (and performs more poorly on the objective metric). This new theory of testing (the Bayesian information objective and prerequisite map models) represents a fundamentally different philosophy of knowledge (as discrete masteries, rather than continuous skills) and opens up the possibility of further innovation in this space and cross-fertilization with developments in cognitive science.

8.2 Implications for Active Sensing

As stated in section 3.1.3, the Bayesian information gain objective used by this work is similar to objectives in active learning and active sensing. Although the predictive model based on prerequisite

maps may be more useful in the educational setting than in a more general active sensing context, the application of a greedy bandit algorithm to this problem using information gain as a reward is novel with respect to the literature. Other work optimizing information gain generally assumes that the Bayesian model is available (see Kreucher et al. (2005) for a literature review). A bandit approach to maximizing information gain objectives can eliminate the need for a provided model in active sensing problems. Conversely, improvements in active sensing algorithms may be applicable to the assessment-creation problem considered here.

8.3 Limitations and Future Work

Optimizing the performance objective given in definition 3.3.1 is an NP-complete problem, and this dissertation has only presented heuristic approaches to maximizing the objective. So, one direction of research is to further explore models and algorithms that may achieve better performance on practical assessment-generation tasks.

On the side of practical application, this dissertation was only evaluated on problems that were conducive to unambiguous discrete groupings of output answers, such as program outputs or logical formulae. In principle, the models presented in this dissertation could be applied to a much wider variety of assessment items (for example, writing graded by peers according to a scoring rubric), but there are likely additional practical difficulties to extracting and administering such assessment items in a scalable and reliable fashion.

Outside of these limitations to the work presented in this dissertation, there are many exciting extensions and applications deserving of further research.

Novel Assessment Generation An enormous area of future study (for education research) is the application of these techniques to generating novel assessments for new topics. The stark paucity of existing instruments, particularly in non-introductory courses, severely hampers rigorous pedagogical research in these areas. A convenient side effect of generating assessments for new topics is that an ATCG-like process also generates an inventory of common misconceptions about the topic, which can potentially inform the design of pedagogical experiments.

Long-term, personalized educational recommendations could use performance on one of these assessments as a reward signal, evaluating pedagogical efficacy on a per-student basis. This can be cast as contextual-bandit problem , and existing techniques could be applied transparently.

Temporal Extensions to Prerequisite Maps A significant assumption in the work presented here is that a student's knowledge state does not change over the duration of the assessment (or as a result of administering any assessment item). In practice, breaking this assumption might allow the work to be applied to some additional tasks. First, the problem of ***knowledge tracing*** involves predicting student knowledge states through a sequence of instructional activities (which can include content presentation, practice problems, and assessments). The author speculates that

prerequisite maps could be helpful in identifying topics in the ***zone of proximal development*** , and thus predicting whether or not a particular instructional activity is likely to change the student's knowledge state.

Second, the problem of ***rehearsal scheduling*** involves modeling human memory and scheduling review and practice opportunities to maximize student retention given a budget on time. Although there are some heuristic models being used in areas like foreign language vocabulary acquisition (Settles and Meeder, 2016), these models typically assume that the retention of knowledge items are mutually independent. The author speculates that prerequisite maps might be helpful in identifying knowledge items that are being implicitly rehearsed by more difficult skills, thus eliminating the need to review them individually.

Hierarchical Extensions to Prerequisite Maps Although the algorithms presented in this dissertation are all polynomial-time, DIDACT does not scale to thousands of equivalence sets of assessment items. This means that it is not currently feasible to build a prerequisite map covering, for example, the content for a complete university curriculum. Humans frequently deal with this problem of scale by applying convenient abstractions to the organization of content, splitting it into "units", "courses", and "sequences". A hierarchical extension to the prerequisite map model might allow efficient querying and inference across much larger knowledge spaces.

Answer-Changing, Explanations, and Consensus Generation In the work presented in this dissertation, it is assumed that the correctness of student answers is given. That is, the assessment item author is able to reliably provide the correct answer, or an expert is available to verify the correctness of a given answer. Ideally, a reliable and inexpensive process will not rely solely on the assessment item author, nor require expert involvement.

In the ATCG setting, after students group their given answer with similar ones, they are presented with a representative set of other students' answers and explanations, and given the opportunity to change their answer for partial credit. The author conjectures that the correct answer is not always the most popular, but it usually has a good explanation. (Otherwise, it would be very hard to establish that it is, in fact, the right answer.) By selecting for explanations that attract more changed answers (other work has explored the possibility of using bandit processes to select good explanations for assessment item answers based on student ratings (Williams et al., 2016)), it may be possible to identify which answer is the most justifiable (has the most persuasive explanation) from student answer-changing behavior. Inspired by work on "Bayesian Truth Serum" (Prelec, 2004), the author suspects that this novel signal may be useful in other forms of opinion aggregation.

Chapter 9

Acknowledgements

In my experience, very few people read the acknowledgements, unless they expect to see their name there. That means that, because you are reading this right now, you are either the rare person who cares enough about this work to understand how it came about, or you are someone who feels you contributed something to this work. Either way, thank you.

9.1 Faculty

First and foremost, I owe a great debt to **Michael Littman**, my PhD advisor. While I've found great joy in our similarities, such as our proclivity toward educational parody music, enjoyment of juggling, and delight in puzzles and word play[1], I've grown the most because of our differences. As a person, Michael is patient, gentle, and humble, despite a lifetime of well-recognized accomplishment. As a researcher, Michael is focused, thorough, and skeptical. As an advisor, Michael does an excellent job scoping projects to the interests and capabilities of the students he works with. Michael has played a unique and mountainous role in developing my intellectual, scientific, and personal maturity. In Michael's own words, advising me has been "like advising two PhD students" (in terms of work, if not results). I am incredibly grateful that he made that investment.

In that vein, I owe a tremendous amount to my undergraduate research advisor, **Judy Goldsmith**, and to **Claus Ernst** and **Uta Ziegler**, with whom I first published. This dissertation would not have come to be without your rich mentorship and constant encouragement.

In the Brown Computer Science Department, I am also incredibly grateful to **Kathi Fisler** and **Shriram Krishnamurthi**, who were an integral part of the research presented here. They have set a tremendous example for me in terms of scientific rigor, interdisciplinary research and collaboration, and clarity of presentation. I am also grateful to **Tim Nelson**, who facilitated collection of some of the data presented here, and to **Stephen Bach**, whose insight and work were significant influences on this dissertation.

[1]This work was partially funded by Ulysses S. Grant #MF41128250A.

I want to give a special thanks to **Stefanie Tellex** and **George Konidaris**, who have been a huge part of my graduate experience. I especially want to thank Stefanie for spearheading the weekly robotics lab meetings and creating such a welcoming and fruitful space for researchers of all experience levels. Both George and Stefanie have had many helpful research conversations with me, offered personal advice, and have patiently smoothed out several of my many rough edges.

Paul Valiant has also had an enormous influence on my graduate experience, particularly in refining my standard of rigor and the quality of my intuition for computer science theory.

Finally, I have learned a great deal from many faculty members through their courses, conversation, and bodies of work. Thank you especially to **James Tompkin**, **Eli Upfal**, and **Amy Greenwald**.

9.2 Peers

I have learned a lot from grad students who came before me, including **Dave Abel**, **Kavosh Asadi**, and **Nakul Gopalan**. I know there's a lot that I never saw, but your calm, rational, and kind approach to research convinced me that I, too, could make it through. Thank you for your many helpful conversations.

To my friends, **Amir Ilkhechi**, **Lucas Lehnert**, **Preston Tunnell Wilson**, **Eric Metcalf**, **Andy Jones**, and **Cam Allen**, I'm grateful for the frequent and detailed conversations we've had about everything from research to humor to philosophy to physics to life. And to all the students I've interacted with in the department (it's a long-tailed distribution — so many of you have had an impact on me) thank you for making this work so much fun.

A truly special thank you goes to **Evan Cater**, who did research with me for all 4 years of his undergraduate education, coauthored a paper with me, and taught me so much. This dissertation would not be where it is today if it weren't for his enthusiasm, knowledge, talent, and encouragement. Thank you, Evan, for being my friend.

I am also grateful to the many other students who have done research with me, including **Eric Choi**, **Alberta Devor**, **Kevin Du**, **Madeline Griswold**, **Fumi Honda**, **Mingxuan Li**, **Zhiyin Lin**, **Naveen Srinivasan**, and **Zora Zhang**.

9.3 Family

Outside of Michael, the person most directly involved in supporting the creation of this dissertation has been my wife, **Jiayin Saarinen**. Her patient listening, kindness, encouragement, honest questions, and care have been instrumental in the completion of this work. I am also incredibly grateful to my sister, **Emma Saarinen**, with whom I share a passion for mathematics, a love of Avatar: The Last Airbender, and our parents. On that note, I'm deeply indebted to my parents, **Tim and Anne Saarinen**, without whom none of this would have ever happened.

There are too many other people with intersecting lives to name everyone who has helped bring this work to completion (directly or indirectly), but I want to at least give credit to **Steve**

and Alexis Schnell, **Greg and Jasmin Chery**, **Admir and Carine Monteiro**, **Brandon and Ashley Moye**, **Jimmy and Anita Allen**, **David Sam**, **Asia Stevens**, and **Sibi Rajendran**.

This dissertation would not exist without **Jesus**.

Bibliography

Adorni, G., Alzetta, C., Koceva, F., Passalacqua, S., and Torre, I. (2019). Towards the identification of propaedeutic relations in textbooks. In *International Conference on Artificial Intelligence in Education*, pages 1–13. Springer.

Anderson, B. and Moore, A. (2005). Active learning for hidden markov models: Objective functions and algorithms. In *Proceedings of the 22nd international conference on Machine learning*, pages 9–16.

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. (2002). The nonstochastic multiarmed bandit problem. *SIAM journal on computing*.

Bayer, J., Bydžovská, H., and Géryk, J. (2012). Towards course prerequisites refinement. *IMEA 2012*, page 4.

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

Botelho, A. F., Adjei, S. A., and Heffernan, N. T. (2016). Modeling interactions across skills: A method to construct and compare models predicting the existence of skill relationships. *International Educational Data Mining Society*.

Brunskill, E. (2011). Estimating prerequisite structure from noisy data. In *EDM*, pages 217–222. Citeseer.

Brydon-Miller, M., Greenwood, D., and Maguire, P. (2003). Why action research? *Action Research*.

Carmo, R., Donadelli, J., Kohayakawa, Y., and Laber, E. (2004). Searching in random partially ordered sets. *Theoretical Computer Science*, 321(1):41–57.

Chalmers, R. P. et al. (2016). Generating adaptive and non-adaptive test interfaces for multidimensional item response theory applications. *Journal of Statistical Software*, 71(5):1–39.

Chang, H.-S., Hsu, H.-J., and Chen, K.-T. (2015). Modeling exercise relationships in e-learning: A unified approach. In *EDM*, pages 532–535.

Chen, Y., González-Brenes, J. P., and Tian, J. (2016). Joint discovery of skill prerequisite graphs and student models. *International Educational Data Mining Society*.

Chen, Y., Wuillemin, P.-H., and Labat, J.-M. (2015). Discovering prerequisite structure of skills through probabilistic association rules mining. *International Educational Data Mining Society*.

Chickering, M., Heckerman, D., and Meek, C. (2004). Large-sample learning of bayesian networks is np-hard. *Journal of Machine Learning Research*, 5.

Corbett, A. (2001). Cognitive computer tutors: Solving the two-sigma problem. In *International Conference on User Modeling*, pages 137–147. Springer.

Cronbach, L. J. (1951). Coefficient alpha and the internal structure of tests. *psychometrika*, 16(3):297–334.

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

Dagum, P. and Luby, M. (1993). Approximating probabilistic inference in bayesian belief networks is np-hard. *Artificial intelligence*, 60(1):141–153.

Daskalakis, C., Karp, R. M., Mossel, E., Riesenfeld, S. J., and Verbin, E. (2011). Sorting and selection in posets. *SIAM Journal on Computing*, 40(3):597–622.

Denny, P., Hamer, J., Luxton-Reilly, A., and Purchase, H. (2008). Peerwise: students sharing their multiple choice questions. In *ICER*. ACM.

Desmarais, M. et al. (2011). Conditions for effectively deriving a q-matrix from data with non-negative matrix factorization. In *4th international conference on educational data mining, EDM*, pages 41–50.

Dilworth, R. (1950). A decomposition theorem for partially ordered sets. *Annals of Mathematics*, pages 161–166.

Eiter, T., Makino, K., and Gottlob, G. (2008). Computational aspects of monotone dualization: A brief survey. *Discrete Applied Mathematics*, 156(11):2035–2049.

Evans, D., Gray, G. L., Krause, S., Martin, J., Midkiff, C., Notaros, B. M., Pavelich, M., Rancour, D., Reed-Rhoads, T., Steif, P., et al. (2003). Progress on concept inventory assessment tools. In *Frontiers in Education*. IEEE.

Fokoué, E. and Gündüz, N. (2016). An information-theoretic alternative to the cronbach's alpha coecient of item reliability. *Journal of Advances in Mathematics and Computer Science*, pages 1–9.

Goldman, K., Gross, P., Heeren, C., Herman, G., Kaczmarczyk, L., Loui, M. C., and Zilles, C. (2008). Identifying important and difficult concepts in introductory computing courses using a delphi process. *ACM SIGCSE Bulletin*.

Guskey, T. R. (2007). Closing achievement gaps: revisiting benjamin s. bloom's "learning for mastery". *Journal of advanced academics*, 19(1):8–31.

Hamer, J., Cutts, Q., Jackova, J., Luxton-Reilly, A., McCartney, R., Purchase, H., Riedesel, C., Saeli, M., Sanders, K., and Sheard, J. (2008a). Contributing student pedagogy. *ACM SIGCSE Bulletin*, 40(4):194–212.

Hamer, J., Cutts, Q., Jackova, J., Luxton-Reilly, A., McCartney, R., Purchase, H., Riedesel, C., Saeli, M., Sanders, K., and Sheard, J. (2008b). Contributing student pedagogy. *SIGCSE Bull.*

Hammersley, M. (1987). Some notes on the terms 'validity'and 'reliability'. *British educational research journal*, 13(1):73–82.

Han, S.-Y., Yoon, J., and Yoo, Y. J. (2017). Discovering skill prerequisite structure through bayesian estimation and nested model comparison. In *EDM*.

Hattie, J. (1985). Methodology review: assessing unidimensionality of tests and ltenls. *Applied psychological measurement*, 9(2):139–164.

Henderson, C. (2002). Common concerns about the force concept inventory. *The Physics Teacher*.

Hestenes, D., Wells, M., Swackhamer, G., et al. (1992). Force concept inventory. *The Physics Teacher*.

Honda, F., Saarinen, S., and Littman, M. L. (2019). Exploration under state abstraction via efficient sampling and action reuse. In *Reinforcement Learning and Decision Making*.

Ikiz, S. and Garg, V. K. (2004). Online algorithms for dilworth's chain partition. *Parallel and Distributed Systems Laboratory, Department of Electrical and Computer Engineering, University of Texas at Austin, Tech. Rep.*

Kaczmarczyk, L. C., Petrick, E. R., East, J. P., and Herman, G. L. (2010). Identifying student misconceptions of programming. In *SIGCSE*. ACM.

Karp, R. M. (1972). Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer.

Kreucher, C., Kastella, K., and Hero Iii, A. O. (2005). Sensor management using an active sensing approach. *Signal Processing*, 85(3):607–624.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86.

Liao, P., Sun, Y., Ye, S., Li, X., Su, G., and Sun, Y. (2017). Predicting learners' multi-question performance based on neural networks. In *2017 International Conference on Behavioral, Economic, Socio-cultural Computing (BESC)*, pages 1–6. IEEE.

Lin, Z., Saarinen, S., and Littman, M. L. (2020). Combining aleatoric and epistemic uncertainties for robust healthcare decision-making. In *Trustworthy AI for Healthcare (AAAI Workshop)*.

Lord, F. M. (2012). *Applications of item response theory to practical testing problems*. Routledge.

Lutz, S. and Huitt, W. (2004). Connecting cognitive development and constructivism: Implications from theory for instruction and assessment. *Constructivism in the Human Sciences*, 9(1):67–90.

Luxton-Reilly, A. and Denny, P. (2010). Constructive evaluation: a pedagogy of student-contributed assessment. *Computer Science Education*.

Lynch, D. and Howlin, C. P. (2014). Real world usage of an adaptive testing algorithm to uncover latent knowledge. In *7th international conference of education, research and innovation (ICERI2014 proceedings). IATED, Seville, Spain*, pages 504–511.

McNeish, D. (2018). Thanks coefficient alpha, we'll take it from here. *Psychological methods*, 23(3):412.

Meyer, J. and Land, R. (2003). *Threshold concepts and troublesome knowledge: Linkages to ways of thinking and practising within the disciplines*. Citeseer.

Minn, S., Yu, Y., Desmarais, M. C., Zhu, F., and Vie, J. J. (2018). Deep Knowledge Tracing and Dynamic Student Classification for Knowledge Tracing. *Proceedings - IEEE International Conference on Data Mining, ICDM*, 2018-Novem:1182–1187.

Mitrinovic, D. S. and Vasic, P. M. (1970). *Analytic inequalities*, volume 1. Springer.

Nathan, M. J., Koedinger, K. R., Alibali, M. W., et al. (2001). Expert blind spot: When content knowledge eclipses pedagogical content knowledge. *ICCS*.

Osborne, J. F. (1996). Beyond constructivism. *Science education*, 80(1):53–82.

Perkins, D. and Martin, F. (1986). Fragile knowledge and neglected strategies in novice programmers. In *Empirical Studies of Programmers*.

Piaget, J. and Cook, M. (1952). *The origins of intelligence in children*. International Universities Press, New York.

Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L., and Sohl-Dickstein, J. (2015). Deep knowledge tracing. *Advances in Neural Information Processing Systems*, 2015-Janua:505–513.

Plajner, M. (2016). Student Skill Models in Adaptive Testing. *Journal of Machine Learning Research*, 52:403–414.

Plajner, M. (2017). Probabilistic Models for Computerized Adaptive Testing. *arXiv preprint*.

Porter, L., Bailey Lee, C., Simon, B., and Zingaro, D. (2011). Peer instruction: do students really learn from peer discussion in computing? In *Proceedings of the seventh international workshop on Computing education research*, pages 45–52.

Prelec, D. (2004). A bayesian truth serum for subjective data. *science*, 306(5695):462–466.

Rao, S. P. and DiCarlo, S. E. (2000). Peer instruction improves performance on quizzes. *Advances in physiology education*, 24(1):51–55.

Saarinen, S., Cater, E., and Littman, M. L. (2020). Applying prerequisite structure inference to adaptive testing. In *Proceedings of the Tenth International Conference on Learning Analytics & Knowledge*, pages 422–427.

Saarinen, S., Krishnamurthi, S., Fisler, K., and Tunnell Wilson, P. (2019). Harnessing the wisdom of the classes: Classsourcing and machine learning for assessment instrument generation. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, pages 606–612. ACM.

Settles, B. and Meeder, B. (2016). A trainable spaced repetition model for language learning. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 1848–1858.

Simon, B., Kohanfars, M., Lee, J., Tamayo, K., and Cutts, Q. (2010). Experience report: peer instruction in introductory computing. In *SIGCSE*. ACM.

Taylor, C., Zingaro, D., Porter, L., Webb, K., Lee, C., and Clancy, M. (2014). Computer science concept inventories: past and future. *Computer Science Education*.

Turing, A. (1950). Computing machinery and intelligence. *Mind*, 59:433–460.

Vie, J.-J. (2018). Deep Factorization Machines for Knowledge Tracing. *arXiv preprint*, pages 370–373.

Vie, J.-J., Popineau, F., Bourda, Y., and Bruillard, É. (2016). Adaptive testing using a general diagnostic model. In *European Conference on Technology Enhanced Learning*, pages 331–339. Springer.

Weiss, D. J. (1985). Adaptive testing by computer. *Journal of consulting and clinical psychology*, 53(6):774.

Williams, J. J., Kim, J., Rafferty, A., Maldonado, S., Gajos, K. Z., Lasecki, W. S., and Heffernan, N. (2016). Axis: Generating explanations at scale with learnersourcing and machine learning. In *Proceedings of the Third (2016) ACM Conference on Learning@ Scale*, pages 379–388.

Wu, R., Liu, Q., Liu, Y., Chen, E., Su, Y., Chen, Z., and Hu, G. (2015). Cognitive modelling for predicting examinee performance. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Yu, D., Varadarajan, B., Deng, L., and Acero, A. (2010). Active learning and semi-supervised learning for speech recognition: A unified framework using the global entropy reduction maximization criterion. *Computer Speech & Language*, 24(3):433–444.

Appendix A

Helpful Definitions

This appendix contains definitions for expectation, Shannon information entropy, cross entropy, and the Kullback-Leibler Divergence.

Definition A.0.1 (Expected Value)**.** *Let X be a random variable with a discrete set of exclusive outcomes $x_1, x_2, ..., x_k$ occurring with probabilities according to $\mathbb{P}$. Then the expected value (with respect to X) of a function of X is defined as*

$$\underset{X\sim\mathbb{P}}{E}[f(X)] = \sum_{i=1}^{k} f(x_i)\mathbb{P}[X = x_i]$$

.

Definition A.0.2 (Shannon information entropy)**.** *Let X be a random variable with outcomes $x_1, x_2, ..., x_k$. The entropy, $H[X]$, is defined as*

$$\begin{aligned}\underset{\mathbb{P}}{H}[X] &= -\sum_{i=1}^{k} \mathbb{P}[X = x_i] \log_2 \mathbb{P}[X = x_i] \\ &= \underset{X\sim\mathbb{P}}{E}[-\log_2 \mathbb{P}[X]]\end{aligned}$$

. The latter formulation is sometimes called the "expected surprisal" and corresponds to the average number of bits required to communicate an event.

Definition A.0.3 (Kullback-Leibler Divergence)**.** *The Kullback-Leibler Divergence (KL-Divergence, or D_{KL}) is a measure of the difference between two distributions P, Q over the same outcome space X.*

$$\begin{aligned}D_{KL}(P \parallel Q)[X] &= \sum_{i=1}^{k} P[X = x_i] \log_2 \left(\frac{P[X = x_i]}{Q[X = x_i]}\right) \\ &= \underset{X\sim P}{E}\left[\log_2 \left(\frac{P[X]}{Q[X]}\right)\right]\end{aligned}$$

Lemma A.0.4 (Gibbs' Inequality)**.** *The KL-Divergence is nonnegative and only zero when the distributions are identical.*

$$\forall P, Q :$$
$$[D_{KL}(P \parallel Q)[X] \geq 0] \wedge$$
$$[D_{KL}(P \parallel Q)[X] = 0 \leftrightarrow (\forall x_i \in X : P[X = x_i] = Q[X = x_i])]$$

Gibbs' Inequality is a well-known result with a variety of straightforward proofs Mitrinovic and Vasic (1970).

Definition A.0.5 (Cross Entropy)**.** *The cross entropy is defined with respect to two probability distributions $\mathbb{P}, Q$ over the same outcome set X.*

$$\begin{aligned} \underset{\mathbb{P},Q}{H}[X] &= -\sum_{i=1}^{k} \mathbb{P}[X = x_i] \log_2 Q[X = x_i] \\ &= \underset{X \sim \mathbb{P}}{E}[-\log_2 Q[X]] \\ &= \underset{\mathbb{P}}{H}[X] + D_{KL}(\mathbb{P} \parallel Q)[X] \end{aligned}$$

The last equality follows easily from definition A.0.3.

Appendix B

Proof of Objective Function Equivalence

Recall Definition 3.3.1:

Definition 3.3.1 (Expected Empirical Bayesian Information-Gain Objective)**.** *The objective function will be stated in equivalent forms, the first using Shannon information entropy and conditional cross entropy, and the second using expected surprisal. In this application domain, larger values are better:*

$$U(Q,S,A) = \frac{1}{|S|}\sum_{s_i \in S}\left[\sum_{Q_j \in \hat{P}_{s_i}}\Big(H(\mathbb{P}[Q_j]) - H(\mathbb{P}[Q_j|\omega_{s_i,k}], \hat{P}_{s_i}[Q_j|\omega_{s_i,k}])\Big)\right] \tag{3.1}$$

$$= \mathop{\mathbb{E}}_{s_i \in S}\left[\sum_{Q_j \in \hat{P}_{s_i}}\Big(-\log \mathbb{P}[Q_j = s(Q_j)] + \log \hat{P}_{s_i}[Q_j = s(Q_j)|\omega_{s_i,k}]\Big)\right] \tag{3.2}$$

where $\hat{P}_{s_i} = A(Q, \Omega_{[1\to i-1]}, \omega_{(s_i,k)}, k)$.

Theorem B.0.1. *The two expressions for the Expected Empirical Bayesian Information-Gain Objective are equivalent.*

Proof. This result follows easily from the definitions for entropy and expectation, and the fact that s_i is drawn from distribution $\mathbb{P}$. □

Appendix C

Proof of NP-hardness

Theorem 6.0.1. *Maximizing the objective given in Definition 3.3.1 subject to a maximum number of queries per student is NP-Hard, even if the underlying joint distribution of all the random variables is known.*

Proof. The proof consists of a reduction from Vertex Cover, an NP-complete problem Karp (1972). The decision version of Vertex Cover is: Given a graph G and constant k, can no more than k vertices be selected such that every edge in G has at least one of the selected vertices as an endpoint?

Observe that Equation 3.1 can be rewritten using the well-known KL-Divergence Kullback and Leibler (1951), an information-theoretic measure of the distance between two distributions:

$$U(A,Q,S) = \frac{1}{|S|}\sum_{s_i \in S}\left[\sum_{Q_j \in \hat{P}_{s_i}}\Big(H(\mathbb{P}[Q_j]) - H(\mathbb{P}[Q_j|\omega_{s_i,k}]) - D_{KL}(\mathbb{P}[Q_j|\omega_{s_i,k}] \parallel \hat{P}_{s_i}[Q_j|\omega_{s_i,k}])\Big)\right].$$

If the true distribution is known, then the KL-divergence term at the end becomes 0, and the objective function becomes:

$$U(A,Q,S) = \frac{1}{|S|}\sum_{s_i \in S}\left[\sum_{Q_j \in \hat{P}_{s_i}}(H(\mathbb{P}[Q_j]) - H(\mathbb{P}[Q_j|\omega_{s_i,k}]))\right].$$

Let $G =< V, E >$ be an instance of Vertex Cover with number of cover nodes k. Construct an instance of the assessment-optimization problem with $|V|$ random variables labeled $Q_1, ...Q_{|V|}$ such that each random variable Q_i has as many bits of entropy as it has edges, is independent of any random variables Q_j where $(i,j) \notin E$, and Q_i and Q_j share exactly 1 bit of information otherwise, mutually independent of all the other variables. These constraints can be satisfied by assigning a coin flip to each edge (whose outcome is a part of each of the attached vertices).Let there be a single student s with outcomes sampled from this joint distribution. If there is a vertex cover using only k nodes, then the performance objective value of $2|E|$ is achievable. (In other words, k queries are able to reveal $2|E|$ bits of information.) If there does not exist a vertex cover using only k nodes, then the performance objective value of $2|E|$ is not achievable. □

Appendix D

Proof of Bounded Suboptimality for a Greedy Query Strategy

Theorem 6.2.5 (Query Complexity of Greedy Querying)**.** $\mathcal{O}(C_G) \leq 2w\lceil \ln n \rceil$

Proof. Similar to the proof of theorem 6.2.4, begin by considering the chain decomposition of G into w chains. By the pigeonhole principle, at least one of those chains has length at least $\frac{n}{w}$. A binary-search strategy on that chain can eliminate at least $\frac{n}{2w}$ nodes, and the greedy strategy by definition must be able to eliminate at least as many, thus leaving at most $\frac{2w-1}{2w}n$ unlabeled nodes after the query. A similar argument applies at any step of the greedy algorithm's execution, so the total number of queries:

$$\mathcal{O}(C_G) \leq \lceil \log_{\frac{2w-1}{2w}} \frac{1}{n} \rceil \tag{D.1}$$

$$= \lceil \log_{\frac{2w}{2w-1}} n \rceil \tag{D.2}$$

$$= \lceil \frac{\ln n}{\ln \frac{2w}{2w-1}} \rceil \tag{D.3}$$

$$\leq \lceil \frac{\ln n}{\frac{1}{2w}} \rceil \tag{D.4}$$

$$= 2w\lceil \ln n \rceil. \tag{D.5}$$

The second-to-last step makes use of a first-order Taylor series approximation to the natural logarithm near 1: $\ln x \geq 1 - \frac{1}{x}$.

□