Learning for the Sake of Learning

1 Why is this work worth doing?

1.1 What is the project?

Over the past several decades, computers and more generally computing and related technologies have changed many aspects of our lives. Nearly every consumer device today is essentially a giant computer, e.g., a typical car today has as many as 50 microprocessors that continually collect and analyze many different sources of data. Many medical devices, likewise, use microprocessors to process various pieces of information (e.g., cutting edge-prosthetics can interpret neural activity in the brain to generate movement). In recent years, the reach of computing has broadened to more sophisticated tasks and are now competing with human-level cognitive abilities. For example, self-driving vehicles are being tested in many of our cities, and automated language translation services have taken huge strikes in producing high quality translations.

Not surprisingly, the field of education has experienced its share of technological impact. On the one hand various online tools such as *Learning Management Systems* or *LMSs* and *question-answer* systems have been developed to allow teachers to share lecture materials and communicate with their students more easily. On the other hand, deeper research efforts have been undertaken to on the ***human-learning problem:***

how to use computer science to help students learn better?

Earlier works on the human-learning problem, such as intelligent or cognitive tutoring systems [Anderson et al., 1995, Koedinger et al., 1997, Walker et al., 2014], relied on modeling human cognitive processes within the computer. The underlying cognitive models were expertly designed based on rigorous theories of how human mind works and applied to the specific topic such as high-school algebra or geometry Anderson [1983]. As a students uses such a tutor, they would be given various questions. The tutor would then treat the responses as semantically meaningful ***data*** and interpret the data to guide the student to better learning outcomes. Cognitive tutors, however, are difficult, especially economically, to apply to more complex and unstructured domains such as topics covered in higher education [Anderson et al., 1995].

More recent approaches took a diametrically opposing methodology and developed machine-learning and AI (Artificial Intelligence) techniques such as neural networks that learn from massive amounts of student data **?**. Such data, usually collected in the context of *Massively Open Online Courses* (a.k.a., *MOOCs*) or Learning Management Systems, include so called "click streams" generated by the interaction of the student with the system. Such data has the advantage of being trivial to collect. Its utility, however, is limited, because it usually lacks significant semantics, e.g., there could be many reasons for why a student clicked one button over another as they are watching a video. Furthermore, the innerworkings of models based on neural networks are opaque and are difficult to understand, even by area experts. These approaches therefore rightly raise an array of questions regarding their predictability, robustness, and fairness.

LAST SENTENCE, IT IS NOT CLEAR WHAT ROBUSTNESS MEANS. WE COULD DROP FOR SIMPLICITY.

In this project, we propose to bridge the gap between these structured and unstructured approaches to the human-learning problem. The basic high-level idea behind our approach is to use mathematically principled structured statistical models such as factor graphs **?** to represent the subject matter of interest and and train these models by using machine learning algorithms and high-precision learning data. At a high level this approach could be viewed as an attempt to find the "sweet spot" between structured and unstructured approaches.

- As with structured approaches, we wish to use a mathematical and statistical model that can be described, specified, analyzed, and explained. Unlike them, we choose our model to be a graph-based model that can represent a variety of domains. In other words, our models have just enough structure to become mathematically tractable and they afford plenty of freedoms to be broadly expressive.

- As with unstructured models, we will use plenty of data to train our graph-based models to be able to "machine learn" the domain knowledge.

Based on this methodology, we formulate the following specific questions that we aim to answer.

1. How should unstructured course materials such as textbooks, tests, student interactions etc. be structured to enable computers (machines) to understand **fine-grained semantics and relationships** between concepts in the subject?
2. Can this structure be **automatically inferred** by computer algorithms?
3. What kinds of data should be collected from the students interactions with the course materials and between each other. How should this be collection be conducted?
4. How can the computer assess the students' understanding of the material in a **highly accurate and provably fair and unbiased** manner?
5. How can the computer identify the specific gaps in students knowledge and provide **personalized feedback** to students?
6. How can the computer provide **feedback to instructors**, about their teaching and the weaknesses and strengths of their course materials?
7. How can we **generalize the techniques to a broad spectrum** of educational settings, e.g., from conventional classrooms to MOOCs, from high schools to graduate-level university courses, from physical sciences and engineering to social sciences?
8. How to guarantee that our techniques are **robust**, i.e., make the minimal possible assumptions on human behavior?
9. How to make our techniques **interpretable and explainable** to humans, meaning that the teachers and students can understand the underlying models as well as the reason for any decisions made by the algorithms?

1.2 Research Outcomes

The answer to the research questions formulated above will show whether machine-learning techniques based on statistical learning with Bayesian Networks is a feasible approach to the human learning problem. The specific outcomes of the project will not only be research papers and knowledge but actual software artifacts that can serve many practising teachers and students today. Because these artifact will collect possibly large amounts of data, it will be possible to validate the theories developed and refine or fix them as needed. The data collected may be of independent interest to researchers and practitioners.

In addition do doing the research and building the tools to support it, we will use them to teach our students and support other instructors to use it. The project will therefore have ***direct and immediate impact*** on education. We envision thousands of students using this tool on a daily basis within a year's time from now. As our techniques and software artifacts mature, we will reach out to high schools in our area both to broaden the impact of our research and also to gain new insights into the problem that we are investigating in the broader context and beyond higher education.

We note that for the purposes of this proposal, our starting point will be STEM (Science, Technology, and Mathematics) in higher education setting but with applications to high schools, and possibly other areas.

1.3 Why Now?

The gap between the structured and unstructured approach to improving human-learning via computation stands as an important challenge in applying powerful computational techniques to the human-learning problem. This is therefore an important and timely problem and progress in solving the problem have important scientific, technological, and societal impacts. But is a solution feasible? The PIs believe that we can make significant progress on this problem because of the following recent fundamental advances in computer science.

1. Proliferation of the Internet and computing devices allow for rapid collection of diverse and detailed data.
2. Advances in machine learning and artificial intelligence (AI) make it possible to build statistical models of complex phenomena and to pair these model with data to solve complex problems. These techniques usually require large amounts of data.
3. Advances in "cloud computing" make it possible to build highly available and scalable systems that can collect huge amounts of data and process such data at interactive speeds. The PIs have built such a system and are using it in our teaching today.

1.4 Major Research Questions

In light of the advances discussed in Section **??**, we now describe the key concrete questions that we aim to answer in this project:

1. How should unstructured course materials such as textbooks, tests, student interactions etc. be structured to enable machines to understand **fine-grained semantics and relationships** between concepts in the subject?
2. Can this structure be **automatically inferred** by computer algorithms?
3. How to arrive at a broad assessment of students understanding of the material in a **highly accurate and provably fair and unbiased** manner?
4. How to effectively identify the specific gaps in students knowledge and make suggestions to guide the students to identify their areas of weaknesses, and provide **personalized feedback** to students?
5. How can the semantic-aware inference from this data be used to provide **feedback to instructors**, allowing them to better understand how well their students are learning and augment their teaching material to serve the students better?
6. How to ensure that these techniques **generalize to a broad spectrum** of educational settings, e.g., from conventional classrooms to MOOCs, from high schools to graduate-level university courses, from social sciences to engineering?
7. How to make these machine-based techniques **robust** by making the minimal possible assumptions on human behavior? How to make them **interpretable and explainable** to humans, meaning that the teachers and students can understand the underlying models as well as the reason for any decisions made by the algorithms?

2 Can we really do it?

We believe that our team has the necessary skills, expertise, and experience to conduct the proposed research. Also, our educational context, consisting of Carnegie Mellon University creates a motivating and supportive environment for this research. Last but not least, we have done fair amount of preliminary work to develop a prototype system that allowed us to test some of the hypothesis put forward in this proposal. We describe below this preliminary work, called the ***Diderot Project***.

2.1 Team

At a high level, this proposal can be viewed as broad and deep computer-science approach to the problem of education. The PIs together have the skills and expertise to cover the spectrum of skills needed to conduct the research. PI Acar is an expert on algorithms, programming languages, semantics, software systems. PI Ada is an expert on algorithms, complexity theory, and databases. The PI Shah is an expert on machine learning, statistical learning theory, game theory. The PIs together have published more than one hundred papers in highly selective venues of computer science. In addition to their research on theoretical areas, the PIs are experienced in building large software systems and have developed the Diderot project. The PI's are experienced in managing research project and have together managed multi-million dollar projects, primarily from government funds for basic research, and in directing and mentoring graduate and undergraduate students.

The PI's have also developed large undergraduate courses and together teach close to one thousand students per year. The PIs are day-to-day practitioners of education and closely interact with the students and are deeply aware of their needs. One of the PIs, Ada, is well known for his excellent teaching and has won important teaching awards (most notably the "Herb Simon Award for Teaching").

2.2 The Diderot Project

TODO: need to thread Nihar's work.

This proposal was primarily motivated by our findings in a project called ***Diderot***. The primary motivation behind the Diderot project is to offer an integrated education environment that allows the instructors, teaching assistants, and the students to study and discuss the course materials in a highly interactive and collaborative framework, and to collect lots of data during this process. Diderot is build to operate on the "cloud" and is able to support any number of users and courses without significant additional work. **Diderot is operational:** an initial version of Diderot was rolled out in Spring 2018, and is now being offered in three different courses reaching more than 500 students.

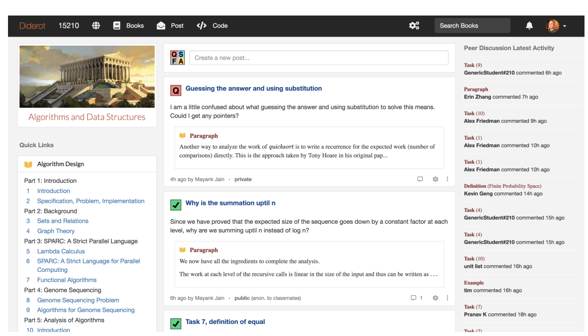
Diderot has many components but at a high level, it can be viewed as a combination of

1. a personal account,
2. an *interactive and online textbook*,
3. an *online test center*,
4. an *authoring system*.

The personal account for each user (student, teaching assistant, instructor, etc) provides each user an identity and provides a personal space for keeping track of personal notes, communications, grades and other personal data.

The online textbook allows the student to study a subject material by using an interactive and collaborative interface, e.g., a student can ask a question about the material as they are studying the material. The online test center allows the instructor to administer online tests and the students to take such tests.

The authoring system enables the content creators, typically the instructors and the teaching assistants, to create interactive and online lecture materials. The key to this system is a *translator* (a.k.a, a compiler) that takes a document written as plain text and converts it into the highly interactive format supported by Diderot. The authoring system is very effective in transforming "dead" textual material into highly interactive materials suitable for modern computing systems. The PIs have used the system to translate their lecture notes written in LaTeX (a text-based format for technical writing) and other materials (close to thousand pages in total) to Diderot's highly interactive format.



During this process, the translator breaks up the document up into relatively small, paragraph-level (or "thought"-level) segments, which we call *knowledge-atoms* or simply as *atoms* and determines the various dependency relationships between these atoms. For instance, a theorem atom may use a previously proved lemma atom, or an "example" atom may provide an example for a "definition" atom. These dependency relationships are usually apparent from the document (e.g., in technical and scientific writing we use references for this purpose), and if absent could be established by simple indexing algorithms. For example, it is usually is simple do detect that a theorem uses a lemma, or that an example illustrates a particular definition.

Data collection facilities of Diderot revolve around users and atoms. Nearly any action of a user usually naturally can be associated with an atom and usually generates data relevant to that action. For example, if a student asks a question about a definition, Diderot records that student, the question itself, and the definition associated with the atom for the definition being asked about. This allows the system to collect precise data about the learning effort made by the student. Similarly, test problems usually naturally relate to specific atoms, usually a small set of them.

In computer science lingo, all of this data can be represented as a graph (network) structure whose elements consist of users and atoms (vertices), the various relationships between them (edges), and the labels along the elements (questions asked, sentiments expressed, points earned or lost, etc).

As a system that is already up and running and is already used by hundreds of students, Diderot creates the opportunity for us to collect massive amounts of data on learning. In this project, we will

TODO: TALK ABOUT HOW WE WILL DETERMINE WHAT DATA TO COLLECT AND WHAT ANALYSIS WE WILL PERFORM.

Methods uned to answer questions. Hypothesis, prove theorem, test.

Sourec of data Interaction and assessment data.

Sample of study More than a 1000 students.

Data collection instruments Diderot itself.

Analyses to be conducted ???

3 Summary and Conclusion

Due to recent advances in computer science, the times are ripe of major innovations in education. We believe specifically that computers can revolutionize education by learning to work with instructors and with the students. This proposal aims to take an important step in this direction. It brings together experts in the areas of computer science, who are also practicing educators. The proposal builds on the success of a proof of concept (Diderot Project) that is already running and collecting data in and out of our classrooms. We seek support from Lyle Spencer foundation to help us base this work on a more altruistic foundation that can increase the greater good to our society and to humanity as a whole.

References

John R. Anderson. *The Architecture of Cognition.* Harvard University Press, 1983. URL `https://hal.archives-ouvertes.fr/hal-00699788`. Excerpts available on Google Books (see link below). For more information, go to publisher's website : http://www.routledge.com/books/details/9780805822335/.

John R. Anderson, Albert T. Corbett, Kenneth R. Koedinger, and Ray Pelletier. Cognitive tutors: Lessons learned. 1995.

K. R. Koedinger, John R. Anderson, W. H. Hadley, M. A. Mark, and Others. Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education (IJAIED)*, 8:30–43, 1997.

Erin Walker, Nikol Rummel, and Kenneth R. Koedinger. Adaptive intelligent support to improve peer tutoring in algebra. *International Journal of Artificial Intelligence in Education*, 24(1):33–61, Jan 2014.