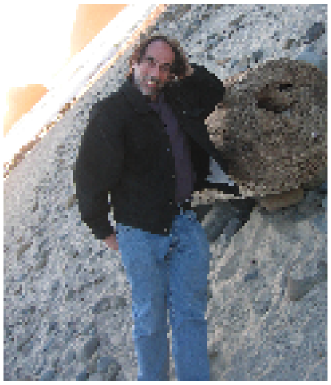

Research Statement

Henry Lieberman
MIT Media Lab

June 2014

Why do I do what I do?

Sometimes, people ask, how do I generate new research topics? One way is often from my own personal experience. Out of some simple frustrations of everyday life, I try to envision how it might be made better, and what computer technology could potentially do to help. I always ask what the fundamental problem really is, and try to attack it at a deep level, that leads me far beyond a quick fix for a particular application. What is the essential knowledge, reasoning, and problem solving processes, underlying the particular situation? How can people and technology cooperate?

A simple bureaucratic dispute between my bank and my credit card company over a mistyped address provided the inspiration that led to my working on end-user debugging for electronic commerce transactions. My inability to figure out how to use a complicated stereo system I just bought led me to work on goal-oriented interfaces for consumer electronics. A bicycle accident, in which I broke both my arms, led to my working on advances in knowledge-based speech recognition.

Basically, I want to make computers, and all technology, easy to use and more helpful to people. Sometimes I'm embarrassed when non-technical friends and family ask me to help with their computer problems. Mostly, these just shouldn't happen, and we should be able to do a lot better. Like many AI researchers, I'm also motivated by trying to use intelligence in computers as a tool for trying to understand the mystery of how people's minds work.

My academic field intersects HCI and AI

I think the best route to progress starts at the intersection of the fields of Human-Computer Interaction (HCI), and Artificial Intelligence (AI). HCI brings paramount concern about the user's needs and well-being, but its current course is not sustainable. We can't keep making progress simply by adding another icon or menu to our screens, nor another mobile device to our pockets. Amid all the talk of "smartphones" and "smart homes", somebody's got to be working on the smart part. That's where AI comes in.

Conversely, the academic field of AI has become too focused on algorithms and theory, or very long term goals such as intelligent robots, and has neglected opportunities to bring intelligent systems to everyday life.

I consider myself a leader in bringing AI and HCI together to create better technology to meet people's needs. See my recent manifesto in AI Magazine [Lieberman 2010], which has been influential in encouraging communication between these two fields. My work also expands out to neighboring fields, like computer graphics and visualization; software engineering; the arts; health, etc. How many researchers can say they've published at the major conferences in AI (AAAI, IJCAI), HCI (CHI), collaboration (CSCW), graphics (SigGraph), software engineering (OOPSLA/SPLASH, ECOOP), and others?

The most directly relevant academic community is the ACM Intelligent User interfaces Conference (IUI), in which I play a leading role. I was twice Program Chair of this conference, and continue to serve on its Advisory Committee and Program Committee. I recently helped launch, and am Associate Editor of, the new ACM journal *Transactions on Intelligent Interactive Systems*, devoted to this synthesis.

My work in Commonsense Reasoning

For most of the past decade, my work has concentrated on bringing Commonsense reasoning to a wide range of interactive applications. As a longtime student of Marvin Minsky, I took to heart his continual insistence that Commonsense knowledge is the central problem of Artificial Intelligence.

I believe that one of the main reasons technology is so difficult to use and ineffective is that computers simply don't know enough about people and everyday life. Just as having a base of Commonsense knowledge is essential for communication between people (a lesson learned the hard way in early work on natural language processing), so it is between people and computers. But why are so few people working on it?

I saw that the traditional field of Artificial Intelligence seemed stuck in two different ways. On one hand, it grew more theoretical, and was taken over by mathematicians who enjoyed optimizing algorithms that could be easily compared analytically and in "bake-off" contests. On the other hand, it became mired in more and more esoteric application areas, due to pressure to show "relevance" to funders. AI systems were built for military planning and configuration of electronic equipment, while the editors, browsers, and drawing programs that ordinary people used every day remained as stupid as ever.

In the early 2000s, I worked closely with the late Push Singh, whose Open Mind Commonsense Web site provided the first substantial and accessible collection of Commonsense knowledge. But how to use it in applications, when it was obviously incomplete, imprecise, and notoriously difficult to reason with using conventional logical tools?

I subsequently developed a methodology for using Commonsense reasoning in intelligent agent software that acts in an assistive role in user interfaces. This methodology involves using Commonsense for providing intelligent defaults, anticipating user needs and preferences, and making the interface more sensitive to the context of the user's life. It also involves judicious selection of where and how it is appropriate to use Commonsense. It may not be appropriate in too-critical applications, such as air traffic control where it might make life-or-death decisions. But in everyday, non-critical applications it can often be made "fail-soft" and enlist the user's cooperation in solving problems.

In the intervening years, my students and I have used this methodology to build prototype Commonsense systems in a vast array of application areas. These are detailed, of course, in the references, but some of the applications areas include: Speech Recognition, Photo annotation and sharing, Consumer electronics interfaces, Music, Games, Translation, Medicine, and more.

Advances in Commonsense Reasoning

In the past few years, especially, I have been blessed with the opportunity to work with some fantastic students, particularly Catherine Havasi and Rob Speer, and earlier with Hugo Liu. We have developed what I consider nothing short of breakthrough research in the theoretical mechanisms of how to effectively use Commonsense knowledge.

Traditional reasoning in AI uses the proof mechanisms of mathematical logic, which fare notoriously poorly when it comes to Commonsense knowledge. Commonsense knowledge doesn't admit of the precise definitions that logical and factual knowledge do, and it can lead to contradictions, which cause conventional theorem provers to crash.

Our fundamental new technique, AnalogySpace, uses an analogical reasoning technique rather than chaining logical statements with inference rules. It puts all the concepts into a multidimensional space that can self-organize according to semantically meaningful distinctions such as "good vs. bad" or "easy vs. hard". It can fill in missing knowledge, confirm or question knowledge, perform classification, or identify potentially unusual statements. It is tolerant of noise, contradictions, or differing opinions. Perhaps its strongest point is its ability to deal with less-well-defined concepts such as affect, relevance or point of view.

There are many extensions and offshoots of AnalogySpace that we are exploring in current and future work, such as "structure-mapping" analogy, case based reasoning, planning and meta-reasoning. We are still actively trying to understand what it all means and explore what its true capabilities and limitations are. This has been the subject of our recent research, and will constitute much of our research in the near-term future.

In my work, I like to balance conceptual and theoretical work with work on practical applications. Right now, as explained, I'm focused on the former, but in the medium term future, I would like to swing back my research more to the applications side. We now have new tools that will allow us to revisit some of the older applications areas mentioned above, with far more effective reasoning capabilities. I also would like to expand out into new application areas.

New application areas

Decision support and organizing online debate. Together with Christopher Fry, we are working on a system, Justify, to facilitate online deliberation and debate for decision-making. Much online conversation is just organized chronologically, and newcomers to a discussion have a hard time understanding what has happened and synthesizing it to be useful in making decisions. Justify is an interactive development environment (IDE) for decision support that uses a rich ontology of types to clarify the role of each contribution in an argument, and provide automatic summarization at every level. It also provides for experimenting with alternative negotiation and decision procedures, such as various kinds of voting.

- *"Not-so-Common Sense"*. This means using Commonsense techniques to encode and exploit knowledge in specialized areas like medicine or engineering fields. This is a little like the old Expert Systems in AI, but now Commonsense allows us to blend specialized and general knowledge, reducing the brittleness that plagued early expert systems. It makes the Knowledge Engineering task of collecting specialized knowledge easier, as it can be done directly in natural language. The systems can better explain their decisions in Commonsense terms. Since this technique "applies to anything", it can interest a broad range of sponsors and academics. We have a project underway to collect knowledge in Neuroscience, in cooperation with the Lab's Ed Boyden, and Sebastian Seung of Brain & Cognitive Sciences.

- *Second Language learning, and multi-lingual communication.* Now that we have substantial Commonsense collections in other languages, I think a tremendous opportunity exists in using them, not only for simple translation, but for language learning, understanding cultural similarities and differences, dealing with bad translations, communication between people who have only imperfect knowledge of each others' languages, etc.

Programming by Example and End-User Programming

Commonsense has not been the only theme that I have pursued during my time at the Media Lab.

I have a deep desire to make the full power of computers accessible to nontechnical users. Right now, there is a barrier between "programmers", who can make computers do almost anything, and ordinary users, who are stuck with whatever so-called "applications software" the programmers produce. I have explored a variety of technologies aimed at bringing the procedural power of programming to users who are not professional programmers.

The principal technique is Programming by Example (also called Programming by Demonstration), the idea that you can teach new behavior to a computer by showing it examples of what you want to do and explaining how they are to be generalized. It is like a "macro recorder" that records steps in the interface, and uses machine learning techniques to produce a generalized program.

I am considered a pioneer in this topic. My book, "Your Wish is My Command: Programming by Example" [Lieberman 01] is the standard reference. More recently I also co-edited a book, "End User Development" [Lieberman, Paterno and Wulf 05] which covers this topic and many related areas concerned with helping users make and modify their own computer applications.

Programming in Natural Language

More recently, I have been exploring the possibility of programming in natural language, bypassing the programming language entirely. This would remove a major barrier to accessibility of programming. While it still not possible to understand arbitrary English, recent advances in natural language processing could potentially make this possible in reasonably constrained situations [Lieberman and Ahmad 10], [Lieberman and Liu 05], [Milhacea, Liu and Lieberman 06]. This is an extremely ambitious goal, but I think I am making significant progress and hope to continue it in the future.

What happens when things go wrong?

I am also very interested in providing better support for debugging, or, more generally, understanding what to do when any technology doesn't do what you want or expect. Work directly on program debugging [Lieberman and Fry 97] has led to work more generally on end-user debugging for e-commerce transactions [Wagner and Lieberman 04], as well as help and explanation [Lieberman and Kumar 05]. This work has potentially extremely high value applications in industry, as fixing problems and customer service are major issues for many companies.

Early Software Agents Group work

Though, for space reasons, I won't cover the earlier parts of my career in as much detail as above, I'll briefly review some highlights. Before taking over the Software Agents group of the Media Lab around 2003, I worked with Pattie Maes, who directed the group. There we developed many of the principles and prototypes that underlie much of today's modern interface agent software, particularly for agents on the Web. Highlights include Letizia, {Lieberman 95], probably the first Web agent for personalized browsing; Aria [Lieberman, Rosenzweig, Singh 01], which introduced intelligent assistance into storytelling with media libraries, and some of whose features appear in modern applications such as iPhoto; finding help from experts [Vivacqua and Lieberman 00]; and my work in the early Semantic Web, where I co-edited the first technical book on the Semantic Web [Fensel, Hendler, Lieberman, Wahlster 03] after Berners-Lee's introduction.

The Visible Language Workshop

From about 1987 to 1994, I worked with Muriel Cooper and Ron MacNeil at the MIT Media Lab's Visual Language Workshop, where we developed interactive tools for visual design, visual thinking and graphical programming. I wrote a retrospective of the work of this time period in Communications of the ACM, following Muriel's death in 1994 [Lieberman 96]. It was during this time period that I developed much of my work in visual programming, programming by example, and graphics and visualization techniques.

My work at the MIT AI Lab

Prior to joining the Media Lab, I worked for twelve years at the then MIT Artificial Intelligence Lab (now part of MIT's CSAIL, Computer Science and Artificial Intelligence Lab).

Logo and Education

My first job there was with Seymour Papert's original Logo group, which ignited my passion for computers and education. I developed our Lisp version of Logo, and the first bitmap, color, and 3D graphics systems for Logo.

Back in SigGraph 1978, [Lieberman 78] I published the "flood fill" curve-filling algorithm, which is now part of every drawing program. Former Media Lab director Walter Bender always used to introduce me as, "this is the guy who invented flood fill!".

Actors

From about 1979 to 1987, I worked with Carl Hewitt on the Actor model of parallel, object-oriented computation for AI. Along with Smalltalk, this was one of the first truly object-oriented languages. I still believe this is the best way to view parallel computation. It was way ahead of its time, but as we approach the era where parallelism will be commonplace even in cheap computers, its time may now be upon us.

I'm perhaps best known in software engineering for my introduction of prototype object systems [Lieberman 86] at the first OOPSLA conference. Prototype object systems now appear in many modern languages, including Javascript.

My 1983 paper on the first real-time garbage collector was the key development that made dynamic-storage languages like today's Java and Python practical on a large scale [Lieberman and Hewitt 83].

My 1980 Tinker system [Lieberman 80] was one of the first real Programming by Example systems, and perhaps still the most general.

Teaching and Mentoring

Though my position has not been as a "professor", and I do not have an official academic teaching load, I very much enjoy teaching and mentoring students. The Media Lab has permitted me to teach courses and act as a thesis advisor to students in much the style of a professor. I take pride in my teaching abilities, and I receive consistently positive student evaluations

Educational philosophy

As I started out my MIT career with the original Logo group at the AI Lab, I am a disciple of Seymour Papert. I fully subscribe to the Constructionist philosophy of education, and the Media Lab's "atelier" style. I was both a student, and taught, in the MIT Experimental Study Group (ESG), a bastion of alternative education at the Institute. People learn by doing, and thinking about what they do with what Papert calls "powerful ideas". I believe in student-centered, problem-based learning, and critical thinking about scholarship and active participation in the academic community. I believe, with Howard Gardner, that students have many different cognitive styles for learning, and it is necessary to adapt teaching styles (being more or less directive, for example), to best fit the student's style. Teaching best takes place by example.

Media Lab teaching

I have taught a Media Lab graduate course, "Commonsense Reasoning for Interactive Applications" since 2002, which has taught our Commonsense methodology to students. I start out the course by telling students, "The purpose of this course is to trick you into doing research". This trick often works, much to the delight of the tricked. This course has generated many course projects that have developed, way beyond the end of the course, into MS and PhD thesis projects, inter-group collaborations, and published papers.

In addition, I taught a graduate course in AI and HCI as a visiting professor at the University of Paris, and have taught numerous conference tutorials, short courses, etc. I also especially enjoy co-teaching, and have taught with other professors, including Ted Selker, Pattie Maes, Hiroshi Ishii, Mitchel Resnick, and Hal Abelson, and postdocs Catherine Havasi and David Maulsby.

Lab and Sponsor Collaboration

I've thoroughly enjoyed collaborations with other Media Lab groups. We have put a considerable amount of effort in making our Commonsense database and software tools available to other groups in the Lab, and to sponsors. Anyone who wants to "beat a little common sense" into their applications is welcome to use our tools, which are often useful for natural language interaction with the user, reasoning about what the application should do, providing intelligent defaults, tracking user interaction and determining user intent and preferences. Details of many of the numerous collaborations appear in the references. I would say the majority of the roughly 25 Media Lab research groups have used our Commonsense tools for one project or another.

We also enjoy many external academic collaborations elsewhere at MIT (for example, CSAIL, where I used to work), and other universities here and abroad. Notably, we are working with collaborators in other countries, particularly Taiwan, Japan and

Brazil, on foreign-language versions of our Commonsense knowledge base.

Sponsor collaborations

We also enjoy an enthusiastic response from our industry sponsors, especially recently, as our tools have reached a level of maturity where companies can begin to consider using them in practical situations. Some of our software is being used by Samsung for semantics for speech recognition commands on the 2014 models of Samsung Smart TVs. Many companies are also using our tools on a prototype basis. Companies include Bank of America, British Telecom, Dentsu, Microsoft, Nihon Unisys, Proctor & Gamble, Samsung, and others.

A topic of particular interest to many companies at the moment is Opinion Analysis, since companies are intensely interested in monitoring blogs, forums, Twitter, etc. for customer opinion. Commonsense NLP can sometimes discover a non-obvious consensus among users, or other insights that go beyond the simple word-counting that constitutes the only other alternative. We've packaged up a ready-to-use opinion analysis and visualization platform, Luminoso [Speer, Havasi, Treadway, Lieberman 10] that is enjoying popularity among sponsors.

Thanks!

Finally, I'd like to take this opportunity to thank my colleagues at the Media Lab, and elsewhere at MIT, other universities, and our sponsors, for their continuous support, friendship, and enlightenment throughout the years.