Note: This is an author pre-submission version of a chapter from P. Michelucci (ed.), Handbook of Human Computation, DOI 10.1007/978-1-4614-8806-4_9, © Springer Science+Business Media New York 2013, published version available at: http://link.springer.com/chapter/10.1007/978-1-4614-8806-4_9

Human Computation: A Manifesto

Pietro Michelucci[1]

[1]ThinkSplash LLC

Abstract Today humans face many challenges as a species, including some that pose grave risks. Technology has been a significant contributor to these risks, but it may also lead to solutions. In the first part of this chapter, we consider how Human Computation (HC), the study of humans as computational elements in a purposeful system, has already been helpful in solving problems. We further consider why HC may be instrumental for mitigating future risks. In the second part of this chapter, we examine the maturity of human computation as both a practice and a discipline. This analysis informs a proposal for technical maturation as well as a formal definition of the field and its distinguishing qualities, all in service of accelerating research and ensuring responsible use of any resultant capabilities. Though the ideas in this chapter may be informed by engagement with the HC community, this manifesto represents a personal perspective.

Introduction

To live in the space of hope is to exist in an uncertain future. We have become complacent in our circumstantial despair and now avert our eyes from the mounting challenges posed by the explosion of innovation in this digital age. Indeed, it is easier to believe that "powers that be" or even technology itself will deliver us beneficently from extinction. But should we accept on faith that all sovereign nations and rogue states have employed provably foolproof safeguards against unintended nuclear missile launches because if even the slightest chance of failure existed, the consequence would be so grave as to compel such safeguards? And what of Thomas Friedman's (1999) democratization of technology? The widespread availability of increasingly potent capabilities has empowered individuals and small groups with state-level capabilities. How does a people safeguard against ubiquitous omnipotence?

In the remainder of this introduction, we consider the growing potential for threats due to existing and emergent technologies, examine proposed strategies for managing them, and consider how Human Computation (HC), the study of humans as computational elements in a purposeful system, may be instrumental for mitigating future such risks. Following the introduction, we examine the maturity of human computation as both a practice and a discipline. This analysis informs a proposal for technical maturation as well as a formal definition of the field and its distinguishing qualities, all in service of accelerating research and ensuring responsible use of resultant capabilities. Though the ideas in this chapter may be informed by engagement with the HC community, this manifesto represents a personal perspective.

The democratization of power

Consider that a sophisticated terrorist group could employ a single person with a "suitcase nuke" (See Figure 1; Woolf, 2010; Horrock, 2001) to devastate the center of a metropolitan area (Bunn and Maslin, 2011). The single greatest barrier to constructing such a weapon of terror is the acquisition of weapons-grade fissile material (Horrock, 2001), such as highly enriched uranium (HEU). As it turns out, during the period from 1993 to 2007 the International Atomic Energy Agency reported 18 incidents of HEU trafficking (see Sanfilippo et al., this volume), some of which involved seizures of kilogram scale quantities (IAEA, 2007).

Figure 1: A backpack for the US-manufactured Mk-54, a man-portable tactical nuclear weapon.[1]

Consider the polymerase chain reaction (PCR). This is a technique for high volume replication of DNA, the molecule that encodes the genetic programming for all known life, including most viruses. PCR requires a series of carefully calibrated temperature changes over a period of time. Such a process is enabled by a device called a "thermocycler", which is basically a high-precision, programmable oven. If you would like to try it in the comfort of your home, you can purchase a kit (Figure 2) for $599.00 at Amazon.com. For safety, please replicate only harmless DNA.

Figure 2: A low-cost thermocycler

Tiger by the tail

Nuclear weapons management and genetically engineered pandemic viruses are among a growing list of *known* risks. What of the *unknown* risks? In their discussion of cumulative culture as a collective memory for preserving and advancing technology, Paul Smaldino and Peter Richerson (this volume) aptly observe that no single human being today knows how to build a modern computer from scratch. This calls attention to our reliance on both communities of distributed knowledge and the infrastructure that supports the propagation of such knowledge,

[1] Photo Source: http://en.wikipedia.org/wiki/Suitcase_nuke; licensed under Creative Commons attribution CC BY-SA 3.0

and hence our vulnerability to a breakdown of either. The situation is far worse. The insularity of expert knowledge has become such that even within a narrow field of study, the rate of advancement is so great that it is nigh impossible for researchers to maintain broad awareness of the intradisciplinary consequences of their work, not to mention the combinatorial explosion of disruptive possibilities that arises when new technologies are combined across fields. Thus, even with the most conservative policies in place, we could not presently appreciate the deep and thorough implications of our technological pursuits. Technology today is a tiger held by the tail.

An Aristotelian Oath

At the turn of the millennium, Bill Joy, the co-founder of Sun Microsystems, wrote a sobering Wired article (Joy, 2000) in which he sought to rouse the rest of the world to the looming dangers of unchecked technological advancement, particularly in the areas of genetics, nanotechnology, and robotics (abbreviated "GNR"), with a focused concern about self-replication in all three domains. His answer to the existential threat has been one of relinquishment - that is, advocating that we simply give up certain perilous technological pursuits, and verify compliance by embracing a "strong code of ethical conduct".

Measures and countermeasures

Joy's efforts to begin this conversation in earnest led to a panel discussion event at the Washington National Cathedral called "Are We Becoming an Endangered Species? Technology and Ethics in the Twenty First Century". In this discussion, Raymond Kurzweil (2001) countered Joy's noble, though perhaps unrealistic vision of consensual relinquishment by suggesting that we proceed gingerly: "…the only viable and responsible path is to set a careful course that can realize the benefits while managing the risks." In supporting this risk-management view, Kurzweil appealed to the observation that new technological threats do not arise in a vacuum, and that there is a commensurate coevolution of technological means to control them. He took computer viruses as a case study, observing that digital disease has remained in check due to the ebb and flow of measures and countermeasures (e.g., anti-virus software). From this, Kurzweil surmises that giving 15 billion dollars to NIH and NSF to spend on countermeasures to address new technology threats would go a long way toward keeping Joy's GNR risks in balance. And perhaps it would, though it has not been tried 12 years later.

Irreversible disruption

Kurzweil's view on countermeasures, however, underplays the role of ecology, because it ignores that both technological risks and controls exist within a context that critically influences outcomes. Technology, humanity, and the planetary environment in which they coexist form a closed dynamical system. Such complex systems exist in an equilibrium state. As such, they exhibit sensitivity to bifurcation (e.g., Silvert, 2002). That is, when a perturbation occurs that causes tolerances to be exceeded, there is a destabilizing and potentially irreversible effect. Though such a system will likely settle into a new equilibrium state, it may be qualitatively different than the former one. For example, there is a level of ionizing radiation above which most organisms cannot survive. Consider a new, runaway technology that irradiates the biosphere. If fatal levels of radiation are absorbed before a protective technology can be developed, then humanity faces extinction. The key point here is that as technologies extend their impact to a global scale, they are more likely to disrupt homeostatic factors in irreversible ways. In the microcosm of Kurzweil's computer viruses, there has always been the option to "cheat", that is to transcend the virtual domain within which these viruses are transmitted by physically disconnecting computers from each other. Such a cheat does not exist in the physical world[2].

Can machines save us?

How then do we mitigate the existential risk of irreversible disruption? Perhaps we leverage the inimitable power of the very emergent technologies we fear. For example, could we possibly build a machine that is smart enough to save us from our own undoing? And to what consequence? Raymond Kurzweil is well known for popularizing John von Neumann's notion of a technological singularity (Kurzweil, 2006), the point in time at which machine-based intelligence will exceed human intelligence. This is often misconstrued to represent the demise of the humanity. In fact, "singularity" is a term borrowed from cosmology to refer metaphorically to a black hole's event horizon, beyond which nothing is knowable. The implication is that we cannot predict what life would be like after such an event. Most theories anticipating the near-term occurrence of a singularity are predicated on the belief that computer processing speed, in terms of calculations per second, is somehow tantamount to intelligence. In this view, a simple extrapolation of Moore's Law, which predicts a doubling of computational speed every 1.5 years, suggests that home computers will exceed the intelligence of humans by

---

[2] In personal correspondence, Michael Witbrock has aptly observed that this is not strictly true; that if we were willing to transcend our planetary context, space colonization might afford a similarly cheat.

the year 2020. Kurzweil, however, acknowledges that processing speed is not enough - that to manifest increases in processing speed as superior intelligence, it will be necessary to build machine-based systems that emulate a precise physical model of the human brain. But is such a model truly within reach?

The elusive singularity

Our understanding of the human brain has increased dramatically over the past decade. We are developing a more detailed understanding of the role of glial cells as an adjunctive communication network to neurons and the existence of stigmergic hormonal processes used to communicate locally in the brain (see Larson-Prior, this volume). We, thus, increasingly view the brain as a complex system of intertwined systems. We are also now, for better or worse, able to use brain scans (fMRI) to measure consumer preference, detect lies, and recognize increasingly complex thought patterns. But recognizing patterns in the brain tells us no more about how those patterns formed than recognizing an animal species tells us about its complex ontogeny. Indeed, these advancements suggest, perhaps more than anything, that there is more to learn about the brain than we previously realized before it could be replicated in-silico (or the synthetic substrate du jour).

But even if we could embed a functional model of the human brain in a computer and run it a thousand or even a million times faster than biological brain, wouldn't it still think only as well as a human? In other words, wouldn't the complexity of its thought processes be the same and wouldn't its capacity for knowledge remain unchanged? Furthermore, there is no evidence to suggest that merely adding artificial neurons to such an artificial brain would make it smarter or that we would have any idea how to usefully connect it to other artificial brains to produce superhuman intelligence. This is not to say that machine-based intelligence will never exceed, in some manner, human intelligence, but rather that the key enabler of such advancement will likely not be processing speed or even brain replication, but rather a deep and sophisticated understanding of how intelligence manifests within a complex network such as the brain. Only then should we start to worry about machines saving us.

A singularity with humans-in-the-loop

But perhaps now is the beginning of "then". Human computation represents the prospect of a different kind of technological singularity, and perhaps one that is more imminently attainable. Indeed, the opportunity exists today to sidestep the issue of replicating human intelligence in machines and turn our attention more directly to the study of methods by which unprecedented cognitive capabilities could be achieved through a carefully conceived combination of biologic human intelligence. In other words, we already have computational agents that are as smart as humans - they're called "humans". Let us then investigate in earnest how we might support the interaction of these agents within a technology-mediated infrastructure toward a degree of cognitive sophistication heretofore unseen. Indeed, there is preliminary evidence (see "Organismic Computing" chapter, this volume) to suggest that, under the right circumstances, large groups of people can exhibit greater synergy than smaller groups. If we can identify and implement such circumstances in sustainable and purposeful ways, then perhaps we can induce a phase transition in humanity - a fundamental change in its collective capability without loss of individuality[3].

When technology is a solution

Bill Joy (2006) once said in a Ted Talk, "You can't solve a problem with the management of technology with more technology." To this we might add: "…unless it is problem-solving technology." Even if one does not fully embrace a speculative future with collective superhuman intelligence, there are many practical examples today of human computation technology being used to solve problems, some of which were potentially caused by technology in the first place. For example, Patrick Meier (this volume) reports on the use of Ushahidi, a crowd-powered crisis management system, to mitigate the damage caused by Hurricane Sandy and Typhoon Pablo. The frequency of intense storm systems such as these is believed to be increasing as a result of climate change (Knutson, 2010), which itself has been linked to democratization of external combustion engine technology and the resultant carbon dioxide emissions (Solomon, et al., 2009). Human computation is also being used to improve outcomes in infectious disease (see Wicks and Little, this volume), for which technology has also been implicated as a cause (Brieman, 1996). Indeed, Haym Hirsh's essay "Human Computation in the Wild" (this volume) is rife with examples of crowd-enabled systems solving problems, even in the pre-digital age. As our understanding of human computation becomes more sophisticated and we gain experience in its application, it is not unreasonable to expect that it will become more prevalent in our arsenal of coping strategies.

[3] This notion of a phase transition in humanity derives from the canonical notion of a physical phase transition, in which there is a change from one state to another without a change in composition.

What worked for Linux

It is noteworthy that the "top-down" solutions to looming risks proposed by Bill Joy and Ray Kurzweil [4]originated from the Washington National Cathedral, though perhaps more figuratively than literally. Eric S. Raymond (1997) penned a catalytic essay called "The Cathedral and the Bazaar", about open source software (OSS) development. In this essay, Mr. Raymond extolled the virtues of bottom-up software development in which hundreds of disorganized software developers around the world ("the bazaar") volunteered small bits of time in piecemeal fashion, in what resulted ultimately in Linux[5], arguably the most popular operating system in the world. What was most notable to Mr. Raymond was that such a distributed effort with so much left to chance could succeed so brilliantly where the traditional, top-down ("cathedral") model of software development had failed. Ironically, Pavlic and Pratt (this volume) have identified many parallels between human behavior in OSS and adaptive eusocial behavior in ants that endows them with emergent collective capabilities. Thus, it is a thesis of this manifesto that human computation (as a general class of organized distributed behavior) is the metaphorical bazaar to Joy's and Kurzweil's cathedral, and as such, may more robustly and adaptively address the existential risks of tomorrow and the practical issues of today.

A plan for conscientious progress

It is one thing to speak evangelistically of progress and quite another to realize it. The remainder of this chapter serves as a proposal for the conscientious advancement of human computation as both a practice and a science. The next section briefly outlines practical considerations for advancing the state of the art. This is followed by an analysis of human computation as a formal discipline, which forces some stakes into the ground. Finally, recognizing the inevitability of growth in this new field, we consider ways to improve the likelihood that the technology is developed and used responsibly.

---

[4] Bill Joy and Ray Kurzweil are widely respected as technical luminaries of our times. It is only on the shoulders of these prescient giants that a context for advancing human computation is formulated herein.

[5] Linux underlies the Android operating system.

Technical Maturity for Progress

We need repeatable methods. Due to the logistical complexity of human participation in human computation systems, we cannot simply employ extant software engineering methods to accomplish anything more than simple crowdsourcing. This is beginning to change (see Morishima, this volume), but in order to progress at a reasonable pace, putting more effort into HC research and less into HC engineering, we need a basic technical maturity. As it is, each novel manifestation of human computation requires a ground-up development effort. Thus, we will need the HC equivalent of a printing press in order for research to move beyond a geologic rate. The following is a representative list of technical desiderata that would be expected to enable a more mature HC practice.

Infrastructure

HC needs a technological state space, a persistent memory for HC systems that does not rely upon the fallible memory of humans. It would further benefit from an "always-on", generalized load-balancing architecture that is robust to the asynchronous and unpredictable availability of human agents. HC also needs service-oriented protocols that permit function calls to these asynchronous humans. Crowd Agents and related methods (see Lasecki & Bigham, this volume) constitute a significant advancement in this direction.

Programming Language

HC needs a development platform that includes an HC programming language, or at least new HC extensions to existing languages. It needs middleware with common classes of crowdsourcing algorithms, implementations of design patterns (see Greene's introduction to the Techniques and Modalities, this volume), and an associated API; and each platform should have associated open source software development projects to create and curate interface elements suited to human participation in HC systems. Ultimately, function calls should require only a specification of the information processing requirements of the human task (e.g., the input, expected output, processing time requirements, etc.); execution should be handled by platform-specific runtime modules that self-adapt to the interface affordances of the execution platform.

Integrated Development Environment (IDE)

HC needs a single integrated environment for development, debugging, performance testing, and execution. Generic, adaptable, and extensible IDEs exist today. Any one of them could be modified to serve as an interface for HC software development.

A three-phase debugging paradigm would help minimize the expense of utilizing actual human computational resources. In this paradigm, phase one debugging would involve farming out tasks to simulated human agents. This would enable a low-cost evaluation of basic system performance by simulating different degrees of variability in human response time and availability. In the second phase, a combination of machine and human agents could be employed in which the HC behaviors of a small proportion of real human agents would dynamically induce more human-like behavior in the machine agents. This phase would be suitable for testing the ability of the system to properly handle the expected information content returned by humans. The final phase of debugging would employ only human agents to ensure that the system would behave predictably in the context of both system performance and information processing. In this three-phase model, minimal use of human resources during testing would be assured by only advancing to subsequent phases of debugging when previous phases, which involve less human involvement, have passed without errors.

Toward a common framework

This brief exposition is not intended as a formal and precise specification for technical maturity, but rather to be suggestive of the kind of technologies and approaches that would lead to repeatability, modularity, code reuse, and cross-platform execution that is now commonplace in software engineering. A perusal of the Infrastructure and Architecture section (this volume) reveals that some of these pieces are already coming to fruition. Ultimately, it may be the binding of these pieces within a common development framework that gives rise to rapid HC development. However, the degree of community collaboration necessary for such technical coalescence may first require greater conceptual coalescence and maturity as a discipline. Indeed, that is the topic of the next section.

Toward a discipline

In 1995, Donald Liles and his colleagues at the University of Texas in Arlington realized that Enterprise Engineering was beginning to distinguish itself from related fields, and took that as an opportunity to reflect on the significance and characterization of such an occurrence. This exercise in community self-reflection coalesced the views of such paradigmatic thinkers as Thomas Kuhn, Peter Keen, and Gavriel Salvendy into an elegant treatise of disciplinary emergence. According to Liles (1996), a discipline represents a worldview, a community, and a set of practices that generate knowledge, which in turn further informs those practices. As with Enterprise Engineering, the emergence of Human Computation (HC) as a discipline is not an end; it is a process of reorganization to accommodate a distinct and increasingly prevalent new approach.

Liles (2006) proposes a list of six defining characteristics for a discipline, which includes a focus of study, a world view or paradigm that binds the community, a set of reference disciplines from which the new field originated but now distinguishes itself, unique principles or practices, an active research agenda, and societal constructs, such as the deployment of education and promotion of professionalism. Herein we seek to describe the state of Human Computation according to those characteristics in order to better understand where HC stands today as an emerging discipline and to help inform its future course.

Focus of study

According to Liles, disciplines emerge to solve new problems not addressed by existing disciplines. Thus, the focus of study stems from the fundamental question being addressed by the discipline. For Human Computation, in all of its incarnations, the central question distills to:

> "How do we create new capabilities and derive knowledge through human participation in computational systems?"

The pursuit of answers to this question leads to a "body of knowledge, principles, and practices" pertaining to the design and analysis of human computation systems.

Unique worldview

A discipline manifests a unique perspective from which its constituents view the world. This perspective determines the framework of practice and is sufficiently complex to be divided into sub-disciplines. In HC, the uniqueness of this perspective arises in part from the unusual combination of five assumptions:

- **Behavioral** - Human Computation employs and studies human interaction.
- **Complex** - Human Computation necessarily involves a system of humans, which are themselves complex dynamic systems. It is within the structure of this complexity that new capabilities or intelligence may emerge.

- **Ecological** - Human Computation presumes an ecological perspective because participation is situated. Individual cognition and agency are part of an interactive system within which they exhibit reciprocal influence with other agents, both machine and human, as well as the environment.
- **Purposeful** - Human Computation is purposeful at the agent level, system level, or both, whether the goals are imposed overtly or manifest simply as a tendency toward some equilibrium state.
- **Engineered** - Human Computation is the product of engineering, whether information processing architecture, mechanism design, or simply a technosocial infrastructure that gives rise to new patterns of behavior. The engineer may be a person, a system, or even a process, such as evolution.

These five assumptions give rise to a multitude of sub-disciplines that derive from existing parent disciplines but constitute sub-disciplines by their *specific and unique application to human computation*. Among these are:

- *Theory of Computation* – the formal analysis and performance characterization of algorithmic behavior that involves human computational elements (see Crouser, et al., this volume, for a ground-breaking foray into this sub-discipline)
- *Computer Engineering* – the development of a scalable and reliable computational infrastructure to support computation that combines machine and human processing elements (see chapter on "Crowd Agents", Lasecki & Bigham, this volume)
- *Distributed Computing* (particularly multi-agent architecture) – the theory and design of multi-agent computing systems in which some agents are humans (see Castelli, et al., Durfee, this volume)
- *Software Engineering* – a systemic approach to the design, development and testing of software that runs on human computational infrastructure (see Morishima's HC development platform, this volume)
- *Human-Computer Interaction* – the study, planning, and design of human interaction specific to the provision of information processing support to an HC system (see Reeves, this volume)
- *Artificial Intelligence* – the design of HC systems that exhibit intelligence (see Heylighen, this volume)
- *Machine Learning* – the design of machine-based algorithms that incorporate humans as either a source of learning bias or as dynamic resources for augmenting machine capabilities (e.g., human-based genetic algorithms – see Nickerson, this volume)
- *Cybernetics* – the control systems analysis of the constraints and possibilities of closed-loop human computation systems (see Nechansky, this volume)
- *Motivation Theory* – the theory of human participation behavior in HC systems and engineering incentive structures that maximize participation volume and quality (Mason, Ghosh, Reed et al., all this volume), and creating systems that

themselves exhibit goal-directed behavior (see "Organismic Computing," this volume)

- *Evolutionary Biology* – the study of evolution as an algorithmic approach to human computation (see Nickerson, this volume)
- *Cognitive Science* – the study and architecture of HC systems that think and the analysis of their thought processes (see Blumberg's chapter "Patterns of Connection"); also the comparative analysis of information processing capabilities between machines and humans (see Crouser et al., this volume)
- *Entomology* – the study of eusocial insect behavior as both an explanatory and generative model of superorganismic behavior in humans (see Moses, Pavlic & Pratt, both this volume)
- *Organizational Science* – that study of organizational workflow models as candidate HC architectures (see Brambilla & Fraternali)
- *Social Informatics* – the analysis of human social behavior in HC systems via quantitative modeling (see Lerman's introduction to the Analysis section in this volume)
- *Knowledge Engineering* – the encoding and locus of knowledge in HC systems (see Gil, Witbrock, both this volume)
- *Cultural Anthropology* – the use of culture as a model of transcendent state space for collective knowledge (see Smaldino & Richerson, this volume) and cultural evolution as a model for collective problem-solving (see Gabora, this volume)
- *Psychopathology* – the study of mental illness applied to the classification, diagnosis, and treatment of behavioral pathology in societies and superorganisms (see Blumberg & Michelucci, this volume)
- *Social Psychology* – the role of group dynamics and social cognition in collective intelligence and group efficacy (see Woolley & Hashmi, this volume)
- *Information Theory* – the ability to characterize the transformation of information (see Gershenson, this volume) by humans to inform the design and understanding of HC systems
- *Epistemology* – the interplay and representation of belief and truth in human-based computation (see Nechansky, this volume)
- *Cognitive Neuroscience* – the use of biological models of cognition (e.g., brains) to inform the design of distributed thinking systems in which networked nodes are human (see Larson-Prior, this volume)

The existence of such numerous and diverse sub-disciplines suggests that the underlying worldview is sufficiently substantive to support a discipline (Keen, 1980). However, the most telltale sign of Human Computation's disciplinary maturity may be the active research referenced within these sub-disciplines.

To be clear, this list of sub-disciplines does not implicate Human Computation as a transdisciplinary field. The intended direction of applicability is from each of the parent disciplines to HC – not the reverse. That is not to neglect the potential applicability of HC to these or other disciplines, but that is not the relationship being conveyed here.

Reference disciplines

Though new disciplines may emerge to solve problems not addressed by existing disciplines, they critically rely upon the knowledge, methods, and tools of the primary disciplines from which they borrow – their "reference disciplines". As indicated above, numerous disciplines contribute to Human Computation; however, only five of these (see Figure 3), seem truly foundational to HC. In the absence of these reference disciplines the pursuit of human computation would seem untenable.

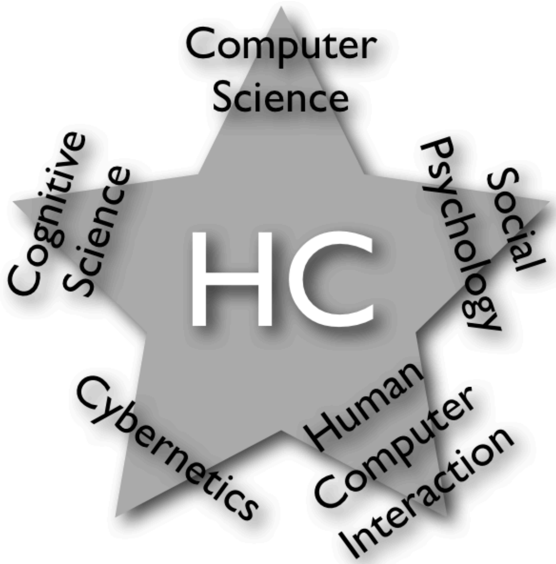

Figure 3: Five reference disciplines of human computation

While the existence of these reference disciplines enables the pursuit of HC, it is their formal acknowledgement that supports the broad acceptance of Human Computation in the scientific community by anchoring its conceptual framework in established bodies of work.

Principles and practices

Human Computation borrows, perhaps most directly, from Software Engineering in terms of theory, abstraction, design, and implementation. While the unique characteristics of HC will likely cause these principles and practices to evolve in new directions, this sub-discipline of the Computer Science reference discipline, serves as a reasonable starting point.

Research agenda

An active research agenda with diverse lines of inquiry is fundamental to a thriving discipline. As evidenced by the rich and diverse body of work conveyed in this volume, HC seems to meet this criterion. However, it constitutes such an expansive and fertile space of research that designating sub-agendas easily becomes an arbitrary exercise in framing the dimensionality of the problem space. Nonetheless, consideration of the foundational assumptions (see above) of HC helps cluster active research into sensible sub-agendas. Taking this approach, we end up with these key research areas within Human Computation:

- Participation – incentivizing participation and modeling interactions in HC
- Application – architecting purposeful HC systems
- Efficacy – engineering circumstances conducive to synergy
- Security – creating HC systems robust to surreptitious participation
- Platform – creating tools and infrastructure to support HC development
- Analysis – the study of HC system behavior

Subdividing the research space in this way helps us locate our own research efforts among related work, and engage within interested sub-communities.

Education and professionalism

The emergence of a discipline is reflected as much by the community structures in place to support sharing and learning as by its conceptual distinctiveness and technical maturity. Though in the near term this handbook may serve as a catalyst for new research, in the longer term it will persist as a community knowledge base of general principles, key ideas, and emerging research. Its diverse interdisciplinary authorship will serve to draw out latent HC community members from related disciplines. Other community structures such as a forthcoming interdisciplinary journal of human computation and a new professional society within IEEE will serve as a home to those expatriates.

Though HC has a rich history of workshops (e.g., HComp, SocialCom, CI, So-Human, etc.), they have been historically hosted by conferences from reference disciplines, and populated primarily by participants originating from those disciplines. However, coinciding with the publication of this first edition handbook, the First AAAI Conference on Human Computation and Crowdsourcing will be held in November of 2013.

In my personal experience, using the term "human computation" in public produces blank stares and confusion. Though not part of Liles' exposition on education, it might be worth considering that general acceptance of a discipline requires not just the formal education of scientists and practitioners, but also dissemination to the general public of a broad-based popular understanding about what human computation does. Proactive public engagement on this would serve to reduce potential misunderstandings about human computation, whether at the definitional level (e.g., "is this about humans using computers?") or the implementation level (e.g., "is this dehumanizing?").

The birth of a discipline

Most, if not all, of the foregoing indicators seem consistent with the present emergence of Human Computation as its own discipline. It is worth mentioning though, that while formal recognition as a discipline may seem beneficial to HC, distancing prematurely from parent disciplines carries its own potential liabilities[6], such as becoming disconnected from communities and related work that have helped sustain HC up until now. Thus, it is critical that we proceed gingerly. Indeed, HC may be best served by preserving strong connections to related disciplines and tempering certain canonical aspects of disciplinary maturation. Perhaps we can borrow from the successes of Cognitive Science, a notable success story among "interdisciplinary disciplines".

This cautionary note notwithstanding, it is still of interest to consider the implications of the apparent disciplinary trajectory of HC based on the above analysis. The primary effect we might expect to result from this is an inflection point in the rate of advancement of HC research and development. Even over the relatively brief 9-month course of this handbook's development, I have borne witness to numerous interdisciplinary "epiphanies" within the book community microcosm. These revelations seemed to result from author exposures to HC-related work across communities that rarely interact. Even if one cannot reasonably extrapolate to the broader community from such anecdotal evidence, it is difficult to ignore the growing interest in this field. So what's next? Perhaps academic programs.

---

[6] I owe special thanks to Mary Catherine Bateson for providing a valued counterpoint to the potential benefits of disciplinary identity, as well as for pointing out the relevance of public education for a new discipline.

A department of human computation?

Today, formal studies in human computation tend to occur as seminar courses within human-centered systems or distributed information systems programs in computer science departments. However, as the HC community coalesces around new research, tools, and community structures, we might expect to witness the emergence of formal programs in HC as we did with Cognitive Science in the late 80s and early 90s. These new programs could be driven by dedicated leaders and faculty who would, through their participation in such programs, begin to identify themselves more formally with the HC discipline, perhaps even referring to themselves as "human computation scientists".

Is there sufficient interest and activity in the field to support a dedicated department of human computation? There's certainly an industry demand for people who are capable of leveraging the power of the crowd. This suggests a commensurate demand for vocational degrees in crowdsourcing. Such a pull from industry may be enough to compel the right visionary dean to back a new department.

Better yet, an interdisciplinary program

On the other hand, perhaps it would make more sense to advance formal education through interdisciplinary degree programs. This would obviate the risk of departmental isolation, which could be fatal for such a conceptually distributed field as HC. Indeed forcing people to choose between an established related reference discipline and a new speculative discipline could reduce the population of both. Furthermore, the barrier to entry for interdisciplinary programs is much lower than for new departments. Among other things, interdisciplinary programs often have minimal requirements for office space and new hires as resources are often shared across existing departments.

The evolution of such programs might begin with concentrations, for which university certificates are awarded. This might be followed by the emergence first of graduate degrees and eventually by undergraduate degrees as the field gradually migrates from specialized to mainstream status. The promotion of such interdisciplinary programs in HC could arise through the efforts of an HC professional society (such as the aforementioned one), by developing academic program requirements for different degree award levels. These core requirements would specify which reference disciplines should have representation among the program faculty, and include guidelines for ensuring a suitable core curriculum as well as recommended course materials. The requirements would be sufficiently flexible to strike a balance between ensuring core competencies in graduates and allowing universities to differentiate their programs. The resultant program specifications could thus be used in turnkey fashion by universities to implement unique programs, taking comfort in the standardization and broad acceptance that would derive from such a community-driven approach. Of further interest to such academic programs might be the conscientious oversight of HC technologies and their development, which is considered next.

A Conscience Committee

Human computation represents a promising means for solving extant and future problems. However, like any new technology it bears its own risks – through vulnerabilities, misuse, and outright abuse. Several contributors to this volume have begun to explore such categories of risk. Dan Thomsen considers human computation systems in the context of cyber security principles to anticipate susceptibility to coordinated attacks as well as vulnerabilities to subtler, though perhaps more insidious, surreptitious participant behaviors. James Caverlee, in his policy chapter on labor standards (this volume; see also Witbrock's introduction to the book section on Infrastructure and Architecture), raises the specter of exploitation and other abuses arising from the commoditization of human computational labor. And from a moral perspective, Juan Pablo Hourcade and Lisa Nathan (this volume) warn against the possibility that human computation could be used as a coercive force. These thought-provoking analyses serve to bootstrap a discussion that will endure for the foreseeable future.

It is impossible to anticipate all of the technical risks and ethical dilemmas that will arise as human computation and spinoff technologies (e.g., artificial generalized intelligence, animal computation, etc.) evolve. Thus, to sustainably address these issues, it is imperative that a body is formed in perpetuity that is geographically and culturally diverse, and composed of human computation cognoscenti, scholars of ethics and morality, and representatives of policy. Such a "conscience committee" would engage regularly in technical risk and ethics analysis, perhaps employing formal methods, such as systemic risk analysis (see Renn, 2004), to ensure a multi-view perspective. The resultant findings would be disseminated to the public via a societal journal and inform new policies, that would be regularly revisited in the context of observed effects and new findings. Crucially, the existence and maintenance of this body would be built into the charter of a human computation society.

Conclusion

Regardless of how one envisions the applications or implications of human computation, its increasingly prevalent and complex role in society is indisputable. In this chapter, we have considered the technological plight of our species, the potential risks and rewards of human computation, the maturity of this evolving discipline, and a proposal for its scientific and practical advancement. Each of us contributing to this handbook has, in one form or another, encountered the transformative effects of human computation. Perhaps you have too. This book represents the beginning of a collective effort to shape tomorrow. Please join us in seizing destiny to empower our hope.

Acknowledgments

The author would like to express his enduring gratitude to each of the 116 contributors to this volume, who are collectively catalyzing the emergence of human computation as a discipline. The author also wishes to gratefully acknowledge useful feedback from Mary Catherine Bateson, Kevin Crowston, Kshanti Greene, Antonio Sanfilippo, and Michael Witbrock.

References

Breiman, R. F. (1996). Impact of technology on the emergence of infectious diseases. *Epidemiologic reviews*, *18*(1), 4–9.

Bunn, M., & Maslin, E. P. (2011). All Stocks of Weapons-Usable Nuclear Materials Worldwide Must be Protected Against Global Terrorist Threats. *Journal of Nuclear Materials Management*, *39*(2), 21–27.

Horrock, N. (2001, December 21). FBI focusing on portable nuke threat. *UPI*. Retrieved June 29, 2013, from http://www.upi.com/Top_News/2001/12/21/FBI-focusing-on-portable-nuke-threat/UPI-90071008968550/

Friedman, T. L. (1999). *The Lexus and the olive tree / by Thomas L. Friedman*. New York: Farrar, Straus & Giroux.

IAEA. (2007). IAEA Information System on Illicit Trafficking and Other Unauthorized Activites involving Nuclear and Radioactive Materials. International Atomic Energy Agency. Retrieved from http://www.iaea.org/newscenter/features/radsources/pdf/fact_figures2007.pdf

Joy, B. (2000, April). Why the future doesn't need us. *Wired*. Retrieved from http://www.wired.com/wired/archive/8.04/joy_pr.html

Joy, B. (2006). Bill Joy: What I'm worried about, what I'm excited about. [Video file]. Retrieved from http://www.ted.com/talks/bill_joy_muses_on_what_s_next.html

Keen, P. (1980). MIS Research: reference disciplines and cumulative tradition. In: Proceedings of the First International Conference on Information Systems, Philadelphia, Pennsylvania, pp. 9-18.

Knutson, T. R., McBride, J. L., Chan, J., Emanuel, K., Holland, G., Landsea, C., … Sugi, M. (2010). Tropical cyclones and climate change. *Nature Geoscience*, *3*(3), 157–163. doi:10.1038/ngeo779

Kurzweil, R. (2001). Raymond Kurzweil: Question and Answers. Presented at the Are We Becoming an Endangered Species? Technology and Ethics in the Twenty First Century, Washington National Cathedral. Retrieved from http://www.kurzweilai.net/are-we-becoming-an-endangered-species-technology-and-ethics-in-the-twenty-first-century

Kurzweil, R. (2006). *The singularity is near: when humans transcend biology*. New York: Penguin.

Liles, D. H., D, P., Johnson, M. E., & Meade, L. (1996). The Enterprise Engineering Discipline. In *in Proceedings of the Fifth Annual Industrial Engineering Research Conference*.

Renn, O., & Klinke, A. (2004). Systemic risks: a new challenge for risk management. *EMBO Reports*, *5*(Suppl 1), S41–S46. doi:10.1038/sj.embor.7400227

Sanfilippo, A., Riensche, R., Haack, J., Butner, S. (2013). Psychosocial and Cultural Modeling in Human Computation Systems: A Gamification Approach. In: Michelucci P, editor. The Handbook of Human Computation. New York: Springer.

Silvert, W. (2002). A bifurcation model of speciation due to environmental change. *Proceedings of First Workshop on Information Technologies Application to Problems of Biodiversity and Dynamics of Ecosystems in North Eurasia - WITA'2001*. Retrieved from http://academia.edu/197962/A_bifurcation_model_of_speciation_due_to_environmental_change

Solomon, S., Plattner, G.-K., Knutti, R., & Friedlingstein, P. (2009). Irreversible climate change due to carbon dioxide emissions. *Proceedings of the National Academy of Sciences*. doi:10.1073/pnas.0812721106

Raymond, E. S. (1997). The Cathedral and the Bazaar. Retrieved from http://www.catb.org/esr/writings/homesteading/cathedral-bazaar/

Woolf, A. F. (2010). *Non-Strategic Nuclear Weapons*. DIANE Publishing.