Human Computation: Charting The Growth Of A Burgeoning Field

Alexander J. Quinn, Benjamin B. Bederson
University of Maryland
Department of Computer Science
Human-Computer Interaction Lab
College Park, MD 20742 USA
{aq, bederson}@cs.umd.edu

ABSTRACT

The rapid growth of human computation within research and industry has produced many novel ideas aimed at organizing web users to do great things. However, the growth is not adequately supported by an overarching framework with which to understand each new system in the context of the old. We give a human computation classification system that can help identify parallels between different systems and reveal "holes" in the existing work as opportunities for new research. Since human computation is often confused with "crowdsourcing" and other terms, we explore the precise position of human computation with respect to other related topics.

Author Keywords

Human computation, crowdsourcing, taxonomy.

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION

Since the birth of artificial intelligence research in the 1950s, computer scientists have been trying to emulate human-like capabilities, such as language, visual processing, and reasoning. Alan Turing wrote in 1950:

> *"The idea behind digital computers may be explained by saying that these machines are intended to carry out any operations which could be done by a human computer."* [59]

His article stands as enduring evidence that the roles of human computation and machine computation have been intertwined since the earliest days. Even the idea of humans and computers in complementary roles was imagined by Licklider in 1960. His sketch on man-machine symbiosis detailed how each would serve the role the other could not [36]. Only recently have we have begun to *combine* machine capabilities with the natural talents of the billions [74] of Internet users around the world.

In 2005, a doctoral thesis on human computation was completed [61]. Four years later, the first annual Workshop on Human Computation was held in Paris with participants representing a wide range of disciplines. This diversity is important because finding appropriate and effective ways of facilitating online human help in the computational process will require new algorithms and solutions to tough policy and ethical issues, as well as the same understanding of users that we apply in other areas of HCI. Today, this burgeoning subfield draws researchers from areas as diverse as artificial intelligence [34,56], business [40,54,27,73], cryptography [69], digital art [17,28], genetic algorithms [30,53], and human-computer interaction [1,3,10,37].

As the area has grown, it has become increasingly necessary to establish a consistent set of terms and distinctions. These are essential to understand the place of each project in this growing field of research. A consistent vocabulary is also needed by graduate students and researchers new to the area, who are trying to synthesize the existing body of work in order to find open problems and opportunities for growth.

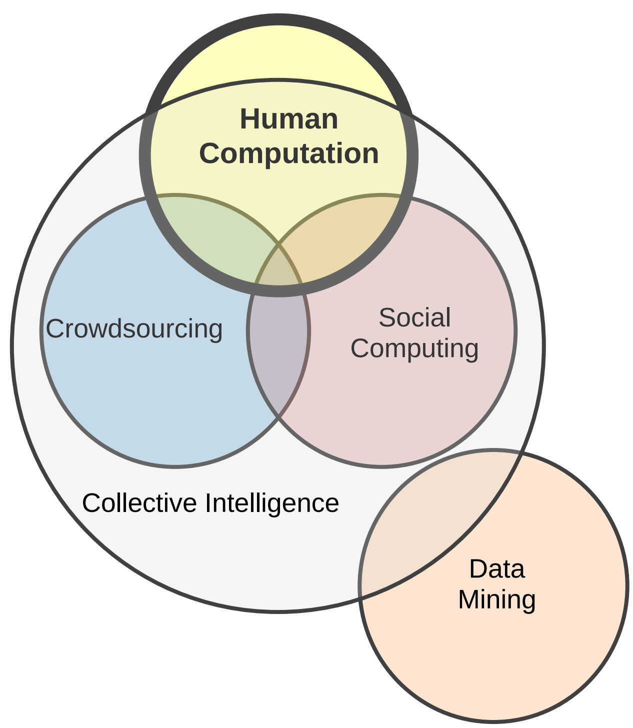

Figure 1: Human computation is a means of solving *computational* problems. Such problems are found only occasionally in crowdsourcing and social computing.

In this paper, we will present a classification system for human computation systems that highlights the distinctions and similarities among various projects. The goal is to reveal the structure of the design space, which we hope will make it easier for researchers to discover unexplored or underexplored areas of opportunity.

The key contributions can be summarized as follows:

- Human computation is defined concretely and positioned in the context of related techniques and ideas.
- We give a set of dimensions that can be used to classify and compare existing human computation systems.
- We show how to apply the system to find open opportunities for future research in human computation.

DEFINITION OF HUMAN COMPUTATION

There are a wide variety of interesting ways that people work with computers and ways they work with each other through computers. For this paper, we focus on *human computation*. Human computation is related to, but not synonymous with terms such as *collective intelligence*, *crowdsourcing*, and *social computing*. These terms are sometimes confused with one another and all are important to understanding the landscape in which human computation is situated. Therefore, before discussing our human computation taxonomy itself, we will define these terms and their relationship to human computation. The definitions and distinctions are important because they dictate the kind of classification that makes sense. Since we claim no overarching authority over the definitions, we will defer to primary sources as wherever possible.

Human Computation

The term *human computation* was used as early as 1838 [71] in philosophy and psychology literature, as well as more recently in the context of computer science theory [59]. However, we are most concerned with its modern usage. Based on historical trends in its use in computer science literature (Figure 2) as well as our examination of citations between papers, it appears that the

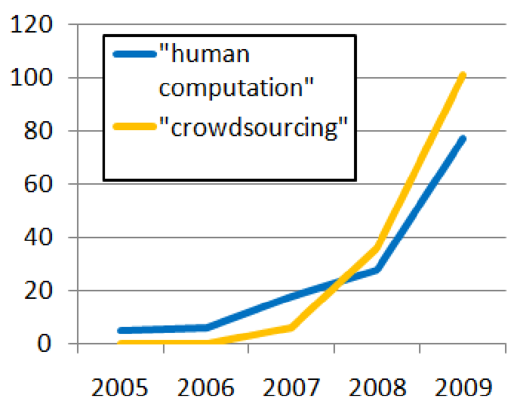

Figure 2: Growth in the use of the terms "human computation" and "crowdsourcing" within the computer science literature was measured as the number of search results in the ACM Guide To The Literature for these two terms in single year intervals.

modern usage was inspired by von Ahn's 2005 dissertation titled "Human Computation" [61] and the work leading up to it [60,63,65,69]. The thesis defines the term as:

> *"…a paradigm for utilizing human processing power to solve problems that computers cannot yet solve."*

This seems compatible with definitions given elsewhere by von Ahn (co-author on the first on) and others (the rest):

> *"…the idea of using human effort to perform tasks that computers cannot yet perform, usually in an enjoyable manner."* [31]

> *"…a new research area that studies the process of channeling the vast internet population to perform tasks or provide data towards solving difficult problems that no known efficient computer algorithms can yet solve."* [9]

> *"…a technique that makes use of human abilities for computation to solve problems."* [76]

> *"…a technique that makes use of human abilities for computation to solve problems."* [7]

> *"…a technique to let humans solve tasks, which cannot be solved by computers."* [52]

> *"…systems of computers and large numbers of humans that work together in order to solve problems that could not be solved by either computers or humans alone"* [49]

> *"…a new area of research that studies how to build systems, such as simple casual games, to collect annotations from human users."* [33]

Most other papers using the term do not define it explicitly. From these definitions and the work described in literature that self-identifies with the term, we derive a few criteria for what constitutes human computation:

- The problems fit the general paradigm of computation. and thus might someday be solvable by computers.
- Participation is solicited proactively for the purpose of performing a computation. This excludes methods such as data mining where human-generated artifacts are analyzed retrospectively.

COMPARISON WITH RELATED IDEAS

The definition and criteria do not include all technologies by which humans collaborate by way of computers, even though there may be intersections with related topics. For example, human computation does not encompass online activity where the initiative and flow of activity are directed primarily by human inspiration, as opposed to a predetermined plan designed to solve a computational problem. Thus, online discussions and creative activities are excluded by the definition.

We argue that editing Wikipedia articles is also excluded, though the distinction is subtle. An encyclopedia purist might argue that an online encyclopedia should contain no creative content and could be interpreted as a very advanced search engine or information retrieval system that gathers existing knowledge and formulates it as prose. Such is the goal of Wikipedia's "neutral point of view" policy [72]. If realized fully and perfectly, perhaps Wikipedia (and similar projects) might reasonably be considered an example of human computation. However, Wikipedia was designed not to fill the place of a machine, but as a collaborative writing project in the place of professional encyclopedia writers of yore. The current form of Wikipedia is rife with discussion and controversy among authors and editors, who each bring different perspectives, which may yield a rich and interesting result, but also inevitably bring personal interpretation and bias to the project. When classifying an artifact, we consider not what it aspires to be, but what it is in its present state.

Perhaps most notably, the very choice of which articles to create is left in the hands of the authors, the people who would be counted as part of the computational machinery if Wikipedia editing were considered computation. A computer with free will to choose its tasks would cease to be a computer. Therefore, Wikipedia authors cannot be regarded as merely performing a computation.

Crowdsourcing

The term *crowdsourcing*, first coined in a Wired magazine article by Jeff Howe [22] and the subject of his book [23], is derived from *outsourcing*. Howe's web site offers the following definition of the term, which frames it as a replacement for roles normally filled by regular workers:

> *"Crowdsourcing is the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call."* [24]

It also lists this secondary "soundbyte" version:

> *"The application of Open Source principles to fields outside of software."*

The key distinction between human computation and crowdsourcing is that human computation replaces computers with humans, whereas crowdsourcing replaces employees and other traditional human workers with members of the public at large. There may be some areas of overlap, especially for work such as translation, where computers and humans already had established roles performing the same type of task. However, the center of gravity of the two terms is different.

The overlap shown in Figure 1 represents applications that could be equally considered replacements for traditional human roles or computer roles. For example, translation is now widely understood as a task to be done by machines when speed and cost are the priority, or by professional translators when quality is the priority. Thus, applications such as monolingual translation [2], which aim to provide a middle ground would be considered as members of both.

Social Computing

Technologies such as blogs, wikis, and online communities are often categorized as examples of social computing, a term that is sometimes used in the context of human computation because the human activities are mediated and supported by computational resources. Various definitions are given in the literature:

> *"… applications and services that facilitate collective action and social interaction online with rich exchange of multimedia information and evolution of aggregate knowledge have come to dominate the Web."* [47]

> *"… the interplay between persons' social behaviors and their interactions with computing technologies"* [16]

The scope of social computing is broad, but always places humans in a social role, and not usually a computational one. The key distinction with human computation is that social computing studies or facilitates relatively natural human behavior that happens to be mediated by technology, whereas participation in a human computation is directed by the computational process itself.

Data Mining

Data mining can be defined as:

> *"the application of specific algorithms for extracting patterns from data."* [18]

Data mining arises in discussions of human computation because some applications make use of things humans have done or created in the past. Take for example Google's PageRank web indexing algorithm [46]. PageRank uses the links between pages to estimate the relevance of their content to search queries. Some might argue that PageRank is an example of human computation because it depends on processing done by humans to perform a job that is of a computational nature. However, the term data mining does not generally encompass the process by which the data was initially created. The human's role is entirely excluded. Therefore, the PageRank algorithm analyzing links in the web, or any other machine retrospectively analyzing artifacts of past human activity, cannot be considered as human computation. In contrast, if the design of the system included the initial human participation, it would not be considered data mining. It would be human computation. This is why Figure 1 shows no overlap of data mining with human computation, crowdsourcing, or social computing. All of them explicitly include the role of the human.

The separation between data mining and human computation also makes sense practically, because many common themes and challenges in the area of human computation (i.e. resistance to cheating, motivation, etc.) do not apply to data mining, and thus would not make sense in a taxonomy that included it.

Collective Intelligence

Encompassing all of the four categories defined so far – human computation, crowdsourcing, social computing, and data mining – is the overarching idea that large groups of loosely organized people can accomplish great things working together. Malone's overview focuses primarily on crowdsourcing examples, but gives a much broader definition for the term itself:

> *"... groups of individuals doing things collectively that seem intelligent."* [40]

In particular, it explicitly includes the PageRank algorithm, as well as virtually any group collaboration, even including "*families, companies, countries, and armies.*" This is roughly consistent with other overviews of the topic [35,41] although specific emphasis may vary. Therefore, as Figure 1 illustrates, collective intelligence is considered a superset of social computing and crowdsourcing, because both are defined in terms of social behavior. Since some data mining applications do not involve humans (i.e. climate data), data mining crosses the circle.

The key distinctions between collective intelligence and human computation are the same as with crowd sourcing, with the additional distinction that collective intelligence applies only when a group is involved, whereas in some rare instances, human computation can apply even to a few (or even one) individuals doing a single task in isolation. This is why part of human computation protrudes outside collective intelligence.

An example of an application that is human computation but not collective intelligence is VizWiz, a mobile application that enables a blind person to take a picture of a situation in life and ask a question about it to workers in the cloud (i.e. "Which of these doors is the men's room?") [4]. This task is already a goal of computer vision research. To cope with the possibility of bad answers (i.e., mistakes, fraud, sabotage, etc.), the system solicits redundant answers from a handful of workers. However, even though the participants are plural, this can hardly be considered an example of collective intelligence since they have no interaction with each other or the blind user.

CLASSIFICATION DIMENSIONS

The common denominator among most human computation systems is that they rely on humans to provide units of work which are aggregated to form an answer to the request. However, there are a wide variety of possible orientations and algorithms that can (or could) be utilized.

The classification system we are presenting is based on six of the most salient distinguishing factors, which are summarized in Figure 3. For each of these dimensions, we provide a few notable values corresponding to existing systems or notable ideas from the literature. Part of the job of researchers and technologists working to advance human computation will be to explore new possible values to address unmet needs, such as better control over speed and quality, efficient use of workers' time, and positive working relationships with the humans involved.

Motivation

One of the challenges in any human computation system is finding a way to motivate people to participate. This is eased somewhat by the fact that most human computation systems rely on networks of unconnected people connected to computers in their homes. Therefore, they need not go anywhere or do anything too far out of their ordinary lives. However, since the computations frequently involve small unit tasks that do not directly benefit the contributors, they will only participate if they have a motivation – a reason why doing the tasks is more beneficial to them than not doing them. Unlike a traditional job, which almost always pays with money, human computation workers may be motivated by a number of factors. We start with the most basic external motivation: pay.

- **Pay**

  Financial rewards are probably the easiest way to recruit workers, but as soon as money is involved, people have more incentive to cheat the system to increase their overall rate of pay. Also, because participants are usually anonymous, they may be more likely to do something dishonest than if they were working in person.

  Mechanical Turk [42] is an online market for small tasks (computational or not) that uses monetary payment. Developers can write programs that automatically submit tasks (called "HITs") to be advertised on the site. The tasks are completed by a network of workers, usually directly through the Mechanical Turk web site. Typically, workers receive a few pennies per task, although the prices and task complexity are driven by the open market nature of the site.

  Another example that uses financial motivation is ChaCha [6], a search service that uses humans to interpret search queries and select the most relevant results. LiveOps [39] is a company that employs workers online to handle phone calls for businesses, as a sort of distributed call-center. The workers follow scripts, which makes the job analogous to an automated telephone system. An older example was the Cyphermint PayCash identification system [48,19], a system for ATMs that used humans to authenticate the user's identity.

  A more recent development is to pay with virtual goods (i.e. FarmVille seeds) for use in popular online games.

- **Altruism**

  Do good. It may sound easy to trust in people's desire to help, but it can only work if participants actually think the problem being solved is interesting and important. Effectively, this requires that the overarching goal be something that most potential workers value.

  Altruism was the primary motivation driving an effort in early 2007 to find the computer scientist Jim Gray when he went missing during a sailing trip. Thousands of online volunteers combed through over 560,000 satellite images [21] hoping to determine Gray's location. Sadly

Dimension	Values	Example
Motivation	Pay	Mechanical Turk [42]
	Altruism	helpfindjim.com (Jim Gray) [21]
	Enjoyment	ESP Game [60]
	Reputation	Volunteer translators at childrenslibrary.org
	Implicit work	reCAPTCHA [68]
Quality control	Output agreement	ESP Game
	Input agreement	Tag-a-tune [31]
	Economic models	(see [19])
	Defensive task design	(see [5])
	Redundancy	reCAPTCHA
	Statistical filtering	(see [8,27])
	Multilevel review	Soylent [3]
	Automatic check	fold.it (protein folding game) [14]
	Reputation system	Mechanical Turk
Aggregation	Collection	reCAPTCHA
	Wisdom of crowds	Kasparov-World chess game []
	Search	helpfindjim.com (Jim Gray)
	Iterative improvement	(see [38])
	Human-genetic algorithm	(see [29])
	None	VizWiz [4]
Human skill	Visual recognition	ESP Game
	Language understanding	Soylent
	Basic human communication	ChaCha [6]
Process order	Computer ➔ Worker ➔ Requester	reCAPTCHA
	Worker ➔ Requester ➔ Computer	ESP Game
	Computer ➔ Worker ➔ Requester ➔ Computer	FACTory [58]
	Requester ➔ Worker	VizWiz
Task-Request Cardinality	One-to-one	ChaCha
	Many-to-many	ESP Game
	Many-to-one	helpfindjim.com (Jim Gray)
	Few-to-one	VizWiz

Figure 3: Overview of our classification of Human Computation characteristics.

the effort was not successful, but the heroic efforts of these volunteers nevertheless illustrated that people will expend significant time and effort for the right cause.

- **Enjoyment**
The abundance of time-consuming, entertainment activities on the Internet attests that even simple forms of entertainment have value to many web users. By making a task entertaining, as a game or any other enjoyable activity, it is possible to engage humans to do tasks.

Games With A Purpose is a popular strategy where you create a game that requires the player to perform some computation in order to get points or succeed. People choose to play because they enjoy it. If the game is fun, they may play for a very long time and supply a lot of computational power. However, it can be difficult to turn many computational tasks into a game that is actually fun to play [62]. Also, when creating a game to serve a human computation purpose, it is important to prove that the game is correct and will yield high quality results, much like designing an algorithm [64]. Many examples have been developed for a wide variety of applications, including Curator [70], ESP Game [60], FACTory [58], foldit [14], Peekaboom [67], Search War [32], Tag-A-Tune [31], and Verbosity [66].

- **Reputation**
Where the problem is associated with an organization with some prestige, human workers may be motivated by the chance to receive public recognition for their efforts. This strategy has been imploded by the International Children's Digital Library to recruit volunteer translators.

- **Implicit work**
It is sometimes possible to make the computation a natural part of another activity users were already doing as part of their lives. Implicit work (also called dual-

purpose work [49]) is the strategy of formulating the computational problem in a way that takes advantage of something web users already do. In doing so, the users' pre-existing motivation to do the activity helps the computation, as well. However, examples are few because it is very difficult to find a match.

ReCAPTCHA [50,68] is a human computation system for transcribing old books and newspapers for which OCR is not very effective. It takes advantage of the pre-existing need for CAPTCHAs, the distorted images of text that are used by websites to prevent access by automated programs. When a user goes to a website (i.e. a webmail service), instead of seeing computer generated distorted text, they see an image of a word from an old book or newspaper, for which the OCR software could not identify the content. By typing the letters in the course of visiting the website, the user provides computational power to help with the transcription effort.

Quality control

Even if the users are motivated to participate, they may try to cheat or sabotage the system. Workers may also be acting in good faith, but misunderstand the directions or make mistakes due to personal bias or lack of experience with the subject matter. Ipeirotis used expectation maximization to estimate the quality, and also infer characteristics of the type of error [27].

The main quality control mechanisms we observed are:

- **Output agreement [64]**
  Epitomized by the ESP game (a game for labeling images), two or more contributors work independently and simultaneously in different locations and the answer is not accepted unless they agree on the same answer.
- **Input agreement [64]**
  This is almost the converse of output agreement. Two humans working independently and simultaneously in different locations are given inputs that may or may not be the same. They are asked to describe the inputs to one another and then try to decide whether they are looking at the same input or different inputs. If both participants agree, then the description is deemed to be correct.

  Input agreement was introduced by Law with the Tag-a-Tune game [31], which collects descriptions of music clips. The players each listen to a sound file and type descriptions. If both players agree that the others descriptions seem to be describing the same clip, then the descriptions are deemed to be relevant.
- **Economic models**
  When money is used as a motivating factor, it may be possible to build in probability-based economic models that guarantee that on the average case, a user does no better than breaking even if they cheat [19].
- **Defensive task design**
  More practically, a range of solutions have been developed recently to improve the accuracy of paid systems like Amazon Mechanical Turk [42]. One approach is to design the tasks so it is no easier to cheat than to do the task.
- **Redundancy**
  By spending more money or finding more contributors, you can have each task done by multiple workers, and use a voting scheme to identify good answers. An added bonus is that once schemes are put in place to identify poor human performers, all of their work can be removed. In our experience using Mechanical Turk, a very significant proportion of bad work is done by a small number of human workers.
- **Ground truth seeding**
  Another approach used by CrowdFlower and many users of Mechanical Turk is to start with a small number of tasks for which ground truth is already known or provided by a trusted person. This allows the system to identify workers who are deliberately submitting useless answers, or perhaps just confused by the instructions.
- **Statistical filtering**
  Either filter or aggregate the data in such a way as to remove the effect of irrelevant contributions. For example, Chen discounts any results that do not fit the expected distribution [8].
- **Multilevel review**
  A first set of participants do the work, and then a second set of participants review and rate the quality of their work. This is similar to output agreement, except that for multilevel review, the work need not be done synchronously.

  Soylent, a word processor that uses human computation to help with text editing tasks, uses such an elaborate multilevel review strategy to ensure quality [3].
- **Expert review**
  A trusted expert reviews contributions for relevance and apparent accuracy. For example, with Amazon Mechanical Turk, people who post tasks (*requesters*) may approve the quality of work and withhold payment, as needed.
- **Automatic check**
  Some problems are much easier to verify than compute, and this class of problems lends itself to automatic checking. For example, in automated planning, a subfield of AI, it is difficult for a computer to generate a plan that gets from the start state to end state, but given a plan as input, it is easy to test if it is correct. In the foldit game, users use a graphical interface to predict protein structures, an important problem in biochemistry. The game uses the Rosetta energy a property of protein structures, to identify the most useful results [14].

- **Reputation system**
  In some systems, users may be motivated to provide quality answers by a reputation system. A worker who frequently submits bad work can be blocked from accessing future tasks or, more positively, given special access to more desirable tasks.

Aggregation

Part of the process of human computation is to combine all of the contributions to solve the global problem. The means of doing this partly determines what kinds of problems a system or strategy can be applied to. The principle aggregation methods are listed below.

- **Collection**
  A knowledge base of discrete facts, or sometimes a hierarchical ontology, are collected. A contribution may either add a new fact or improve quality by correcting, refuting, confirming existing facts in the knowledge base.

  Knowledge Collection from Volunteer Contributors (KCVC) is a way of using human computation that uses the collection method of aggregation. The goal of KCVC is to advance artificial intelligence research by using humans to build large databases of common sense facts. The idea is that humans, by way of their child development and adult lives, acquire a great deal of common sense knowledge (i.e. "People cannot brush their hair with a table."). Large databases of such knowledge, such as Cyc [34], have collected such information into knowledge bases and ontologies using data mining. Several efforts have also attempted to use volunteer contributors to provide the data, either by using games (e.g. the FACTory [58], Verbosity [66], 1001 Paraphrases [12]) or by volunteer knowledge gathering activities (e.g. Learner [11]). The field of KCVC, summarized in [13], is important to AI research and has dedicated workshops and symposia about the topic.

- **Statistical Processing of Data**
  Notably described by Surowiecki [57], the data can be aggregated with simple mechanisms, in some cases a simple average. This is where the "Wisdom of Crowds" term originally came from. For example, consider a game where hundreds of people try to guess the number of jelly beans in a large jar. It turns out that under normal circumstances, the average of the guesses will be very close to the actual count. In his book by the same title, Surowiecki explains how aggregating the answers from a decentralized, disorganized group of people, thinking independently can yield surprisingly accurate results to questions that would be difficult to answer by one person alone. It works only if the individual errors of each person are uniformly distributed, which in turn requires that the individual judgments be made independently and be aggregated correctly.

  Several online polling web sites and prediction markets harness this concept to not only determine a group opinion, but to predict the future [73] (e.g. Ask500People [1], News Futures [45], and Iowa Electronic Markets [25]). While prediction markets are not examples of human computation, they are one of the most commonly cited examples of Wisdom of Crowds.

- **Search**
  Several projects have used large numbers of volunteers to sift through photographs or videos, searching for some desired scientific phenomenon, person, or object.

  The Space Sciences Laboratory at the University of California at Berkeley previously also used human computation to search for tiny matter from space as part of the Stardust@home project [43]. The particles had been mixed into an aerogel collector from the Stardust spacecraft. Contributors searched through photographs of the aerogel for traces of the particles. This recognition problem was much too difficult for computer vision algorithms or even untrained humans. Therefore, participants had to complete an online training program to learn how to identify the tiny particles before they could contribute. With the this aggregation method, the only contributions that are of value are the one(s) that contain the target (e.g. photo of Jim Gray or trace of a particle).

- **Iterative improvement**
  For some applications, it makes sense to give each worker the answer given by previous workers for the same task. As a test of this method, Little asked workers on Mechanical Turk to try to read some text that had been severely distorted for purposes of the experiment [38]. Initially, the image of distorted text was given to two workers. Next, a third worker (or small group of workers) examined the transcriptions from the first two workers, and chose the best one, which was given to the fourth worker as a starting point. They found that iterative improvement had the potential to yield very high quality results, but quality was less consistent than the control method they compared with.

- **Genetic algorithm**
  It is possible to execute genetic algorithms using human computation. The idea is that humans can contribute solutions to problems and subsequent participants perform other functions such as initialization, mutation, and recombinant crossover [29]. A relevant project is the Free Knowledge Exchange (FKE) [30], which uses human-based genetic algorithms to formulate answers to freeform questions answered by users [12]. Although FKE does not fit our definition of computation, the process could presumably be applied to problems that do.

- **None**
  Some human computation systems do not involve aggregation at all, but simply use humans to perform a large number of small tasks which are independent of one another.

Human Skill

Depending on the application, human computation may leverage a variety of skills that are innate to nearly all humans or, in some cases (i.e. translating from Chinese), special knowledge or abilities held by only some. When designing a solution that uses human computation, it is helpful to be very specific about what skill is being used, in order to factor out other aspects of the problem that could just as easily be done by a computer. For example, to improve an image search engine, one could imagine employing an extremely large number of humans to search through images exhaustively to find ones that fit a query. Obviously, it is far more efficient to have them simply associate text descriptions with images in a database. Connecting words with the contents of images is the key part of the problem that humans do not do well. Conversely, computers are better at searching through large amounts of information. Since human computation is focused on computation-oriented tasks, we are primarily interested in skills that computers could do but humans can do better.

Process Order

In any human computation system, there are three roles: the worker, computer, and requester. The requester is the end user who benefits from the computation (i.e. someone using an image search engine to find something). A subtle distinction among human computation systems is the order in which these three roles are performed. We consider the computer to be active only when it is playing an active role in solving the problem, as opposed to simply aggregating results or acting as an information channel. Many permutations are possible. Here are a few we have observed in the literature:

- **Computer ➔ Worker ➔ Requester (CWR)**
  With reCAPTCHA [68], a *computer* first makes an attempt to recognize the text in a scanned book or newspaper using OCR. Then, words which could not be confidently recognized are presented to web users (*workers*) for help. Their transcriptions become part of the transcription of the book or newspaper for use by users (*requesters*) reading or listening to it.

- **Worker ➔ Requester ➔ Computer (WRC)**
  Players (*workers*) of image labeling games [20,60,etc.] provide labels for images, which are then aggregated by a computer to remove labels believed it to be faulty. When a web user (*requester*) visits the image search site and enters a query, the *computer* searches the database of labels to find matches.

- **Computer➔Worker➔Requester➔Computer (CWRC)**
  The Cyc system (*computer*) has inferred a large number of common sense facts by analyzing text. To improve the quality of the facts in the database, they use FACTory [58], a game with a purpose. Facts from the database are presented to players of the game (*workers*) who can confirm or correct them, thus improving the quality of the Cyc database. When an end-user (*requester*) of the Cyc system performs a query that requires AI functionality, the system (*computer)* works again to make inferences based on the information in that database.

- **Requester ➔ Worker (RW)**
  Amazon Mechanical Turk allows a *requester* to submit tasks, such as audio transcription or text dictation, for which no active computational component is necessary. Workers do the tasks for a small fee. In such small cases, quality can be confirmed by spot checking.

Task-request cardinality

When an end-user uses a service powered by human computation, there may be many human workers involved in producing the result, especially if a lot of aggregation is required. In other cases, just one or a handful of workers may suffice. This depends on the structure of the problem, and thus is a key dimension in classifying human computation systems, as well as analyzing the financial or time requirements of any given system.

- **One-to-one**
  With ChaCha's web search a single human worker would handle the entire search. (Recently, ChaCha changed to a multi-stage process for their SMS-based mobile search service.)

- **Many-to-many**
  Image search engines use tagging done by many humans to annotate each image in the search index, which is then used to process any number of search requests. Without receiving several annotations for each of a very large number of images, it would be impossible to return results for any single query.

- **Many-to-one**
  In the above-mentioned search for Jim Gray over 12,000 [55] volunteers worked together to find just one image out of over 560,000.

- **Few-to-one**
  VizWiz [4], the mobile application for blind users, uses a handful of workers on Mechanical Turk to supply redundant answers to each query.

OPENINGS FOR GROWTH

Up to this point, we have described a classification system that can be used to understand human computation systems in the broader context of what exists. We are now ready to use the dimensions to presuppose some possible areas for future exploration. The examples mentioned so far occupy specific points in the design space, but by combining the dimensions in other ways, it is possible to imagine new kinds of systems.

It should be noted that recognizing that a configuration that has not yet been tried does not imply that it would be useful. However, understanding the design space may be helpful for researchers looking for solutions to specific domain problems.

Consider new dimension pairs

For researchers looking for new avenues within human computation, a starting point would be to pick two dimensions and list all possible combinations of values. For example, considering motivation and aggregation shows that input agreement has not been applied with paid labor or any motivation other than enjoyment. Similarly, combining cardinality with motivation reveals that there are no known examples of one-to-one human computation motivated by altruism. One could imagine a service similar to VizWiz, but staffed by trusted volunteers.

Invent new values for dimensions

Another way to use a classification system is to consider if there are other possible values that could be used for a given dimension. For example, one might look for human skills that have not been well-explored, yet. The foldit protein folding game made use of humans' spatial manipulation abilities, a skill that still has not been well explored in human computation research.

Classify new work and consider variations

When encountering new work, it may be helpful to think of the applicable values for each dimension of the system. Doing this may help identify ways in which the novel aspect of the system could be combined with other ideas. For example, when encountering the VizWiz mobile application, one might note that it uses pay for motivation, few-to-one cardinality, and a Requester➔Worker (RW) process order. Changing the process order to (CRW) might yield an interesting application that uses the mobile device's CPU to do meaningful processing before the request is ever entered into the phone. Perhaps it could use situational awareness to suggest possible requests.

CONCLUSIONS

Our motivation in developing this classification system was to stimulate new directions of research in human computation. Many researchers and practitioners are excited about this new area within HCI, but there can be a tendency to focus on just one style of human computation, potentially missing more suitable solutions to the problem.

Beyond new algorithms and designs, there is also a pressing need to address issues related to ethics and labor standards. It is possible that as technology obviates the jobs of some unskilled workers, some future human computation systems could become a viable employment option for such workers. Getting to that point will require much more progress in all aspects of human computation research.

Finally, the first two sections addressed distinctions among definitions in some detail. While it is important as a community to agree on concrete definitions, it is also equally important to consider what is left out. Perhaps future incarnations will be more social in nature, while remaining primarily dedicated to performing computations.

ACKNOWLEDGMENTS

We gratefully acknowledge the advice and support of Chang Hu, Pete Kelleher, Oded Nov, Ben Shneiderman, Alexander Sorokin, Yan Xu, and many others.

REFERENCES

1. Ask 500 People http://www.ask500people.com
2. Bederson, B.B., Hu, C., & Resnik, P. 2010. Translation by Iterative Collab between Monolingual Users, GI'10
3. Bernstein, M., Miller, R.C., Little, G., Ackerman, M., Hartmann, B., Karger, D.R., Panovich, K. Soylent: A Word Processor with a Crowd Inside. In Proc UIST'10.
4. Bigham, J. P., Jayant, C., Ji, H., Little, G., Miller, A., Miller, R. C., Miller, R., Tatrowicz, A., White, B., White, S., Yeh, T. 2010. VizWiz: Nearly Real-time Answers to Visual Questions. In Proc UIST 2010.
5. Callison-Burch, C., Dredze, M. Creating Speech and Language Data W/ Amazon's Mechanical Turk. *Wshp Creating Speech And Language Data* NAACL-HLT'10.
6. ChaCha search engine. http://www.chacha.com.
7. Chan, K. T., King, I., and Yuen, M. 2009. Mathematical Modeling of Social Games. In Proc CSE'09
8. Chen, K., Wu, C., Chang, Y., and Lei, C. A crowdsourceable QoE evaluation framework for multimedia content. In *Proc Multimedia* (2009).
9. Chandrasekar, R., Chi, E., Chickering, M., Ipeirotis, P. G., Mason, W., Provost, F., Tam, J., von Ahn, L. 2010. Front matter. In *Proc of the KDD-HCOMP'10*.
10. Chi, E. H. 2009. Augmented social cognition: using social web technology to enhance the ability of groups to remember, think, and reason. In SIGMOD '09
11. Chklovski, T. Learner: a system for acquiring commonsense knowledge by analogy. Proc K-CAP '03
12. Chklovski, T. Collecting paraphrase corpora from volunteer contributors. In Proc K-CAP'05.
13. Chklovski, Timothy & Gil, Yolanda. Towards Managing Knowledge Collection from Volunteer Contributors, Proc of AAAI, KCVC05 (2005)
14. Cooper,S., Khatib,F., Treuille,A., Barbero,J., Lee,J., Beenen,M., Leaver-Fay,A., Baker,D., Popovic,Z. Predicting protein structures with a multiplayer online game. In *Nature*, 466(7307):756–760, August 2010.
15. CrowdFlower http://www.crowdflower.com
16. Dryer, D.C., Eisbach, C., Ark, W.S. 1999. At what cost pervasive? a social computing view of mobile computing systems. In *IBM Systems Journal*, Dec 1999.
17. Edmunds, P. SwarmSketch. http://swarmsketch.com/.
18. Fayyad, U., Piatetsky-Shapiro, P Smyth, P. Knowledge Discovery and Data Mining: Towards a Unifying Framework. 1999. In *Proc.KDD*, Aug. 1999.
19. Gentry, C., Ramzan, Z., and Stubblebine, S. Secure distributed human computation. In Proc EC '05.
20. Gonçalves, D., Jesus, R., and Correia, N. 2008. A gesture based game for image tagging. In CHI EA'08
21. Help Find Jim (web site). http://www.helpfindjim.com
22. Howe, Jeff, The Rise of Crowdsourcing. *Wired. Jun'06.*

23. Howe, Jeff. *Crowdsourcing: Why the Power of the Crowd is Driving the Future of Business*. Crown.
24. Howe, Jeff. Crowdsourcing: A Definition http://crowdsourcing.typepad.com
25. Iowa Electronic Markets http://www.biz.uiowa.edu/iem
26. Ipeirotis, P. G., Chandrasekar, R., Bennett, P. Report on the human computation workshop. KDD-HCOMP'10.
27. Ipeirotis, P. G., Provost, F., and Wang, J. 2010. Quality management on Amazon Mechanical Turk. HCOMP'10.
28. Koblin, A. Sheep Market. http://thesheepmarket.com.
29. Kosorukoff A. Human based genetic algorithm. IlliGAL report # 2001004. 2001, U of Illinois.
30. Kosorukoff, A. & Goldberg, D. E. Genetic algs for social innov & creativity. IlliGAL rpt #2001005, U of IL
31. Law, E., von Ahn, L. 2009. Input-agreement: a new mchnsm for collecting data using hcomp games.CHI'09.
32. Law, E., von Ahn, L., Mitchell, T. Search War: A Game for Improving Web Search. In Proc KDD-HCOMP'09.
33. Law, E., West, K., Mandel, M., Bay, M., Downie, J. S. Evaluation of algorithms using games : The case of music tagging. In Proc. wISMIR 2009.
34. Lenat, D. B. 1995. CYC: a large-scale investment in knowledge infrastructure. CACM Nov. 1995
35. Levy, P. Collective intelligence: Mankind's emerging world in cyberspace. Perseus, Cambridge, MA, 1997.
36. Licklider, J.C.R. 1960. Man-computer symbiosis. *IRE Transact. Hum. Fact. Engineer*. i, 1 (Mar. 1960), 4-11.
37. Lieberman, H., Smith, D., Teeters, A. Common Consensus. Wshp Common Sense Intell Intrfcs, IUI'07
38. Little, G., Chilton, L. B., Goldman, M., and Miller, R. C. 2010. Exploring iterative and parallel human computation processes. *In Proc KDD-HCOMP'10*
39. LiveOps http://www.liveops.com
40. Malone, T.W., Laubacher,R., Dellarocas, C. 2009. Harnessing crowds: mapping the genome of collective Intellgnc, CCI Working Paper, 2009-001, MIT Ctr 4 CI.
41. Mataric, M.J. Designing emergent behaviors: From local interactions to coll. intell. From Anmls To Anmls 2 '93.
42. Mechanical Turk. http://mturk.com.
43. Mendez,B.M., Craig,N, Westphal,A. J. Stardust@home: Enlisting Studnts & the Public in the Srch for Interstellar Dust. Am. Astro. Soc. Mtg 207 Vol. 37 p.1265. 12/2005
44. Nalimov, E. V., Wirth, C.,Haworth, G. M. KQQKQQ & the Kasparov-World Game. ICGA J, 22 (4). pp.195-212.
45. News Futures http://www.newsfutures.com
46. Page, L., Brin, S., Motwani, R., Winograd, T. The PageRank Citation Ranking. STNFRD INFOLAB. '99.
47. Parameswaran, M. Whinston, A.B. Social Computing: An Overview. In CAIS, vol 19, 2007, p. 762.
48. Peha, J. M., Khamitov, I. M. 2003. PayCash: a secure efficient Internet payment system. In Proc ICEC'03.
49. Quinn, A., Bederson, B.B. A Taxonomy of Distributed Human Comp. Tech Rpt HCIL-2009-23, U of MD.
50. reCAPTCHA. http://recaptcha.net.
51. Rohwer, P. 2010. A note on human computation limits. In *Proc KDD-HCOMP'10.*
52. Schall, D., Truong, H., and Dustdar, S. 2008. The Human-Provided Svcs Frmwk. *Proc CECANDEEE '08*.
53. Secretan, J., Beato, N., D Ambrosio, D. B., Rodriguez, A., Campbell, A., Stanley, K. O. Picbreeder: evolving pictures collaboratively online. In *Proc CHI'08*.
54. Sheng, V., Provost, F., and Ipeirotis, P. Get Another Label? Improving Data Quality & Data Mining. (2008).
55. Silberman, S. Inside the High Tech Hunt for a Missing Silicon Valley Legend. From *Wired Mag*. Jul 24'07. http:// wired.com/techbiz/people/magazine/15-08/ff_jimgray
56. Singh, P., Lin, T., Mueller, E. T., Lim, G., Perkins, T., and Zhu, W. L. 2002. Open Mind Common Sense: Knowledge Acquisition from the General Public. In Comp. Science, vol. 2519. Springer-Verlag, 1223-1237.
57. Surowiecki, J. The Wisdom of Crowds. Anchor Books.
58. The FACTory. http://game.cyc.com.
59. Turing A.M. 1950. Computing Machinery and Intelligence. In *Mind*, vol. 59, no. 236, pp. 433--460.
60. von Ahn, L. and Dabbish, L. Labeling images with a computer game. In Proc CHI'04.
61. von Ahn, L. *Human Computation*. Doctoral Thesis. UMI Order Number: AAI3205378., CMU (2005).
62. von Ahn, L. Games with a Purpose. Computer. Jun '06
63. von Ahn, L., Blum, M., and Langford, J. 2004. Telling humans and computers apart automatically. CACM '07.
64. von Ahn, L., Dabbish, L. General Techniques for Designing Games with a Purpose. CACM Aug'08.
65. von Ahn,L, Graham,M,I, Dabbish,L, Kitchin,D, Blum, L. Solving Hard AI Problems With Comp. Games. CMU Technical Report CMU-CS-02-191, Nov. 2002.
66. von Ahn, L., Kedia, M, and Blum, M. Verbosity: a game for collecting common-sense facts. In Proc CHI'06
67. von Ahn, L., Liu, R., and Blum, M. 2006. Peekaboom: a game for locating objects in images. In Proc CHI'06.
68. von Ahn, L., Maurer, B., McMillen, C., Abraham, D., Blum, M. 2008. ReCAPTCHA: human-based character recognition via web security measures. Science, vol. 321, pp. 1465-1468, September 12, 2008.
69. von Ahn,L., Blum,M, Hopper,N., and Langford,J. Captcha:Using hard AI problems for security. Euro'03.
70. Walsh, G. Golbeck, J. Curator: a game with a purpose for collection recommendation. In Proc CHI '10.
71. Wayland, F. *The Limitations Of Human Responsibility*. Applewood Books, Bedford MA. 1838
72. Wikipedia. http://www.wikipedia.org.
73. Wolfers, J., Zitzewitz, E. Prediction markets. In *J. of Economic Perspectives*, 18(2):107--126, 2004.
74. World Bank. World Development Indicators. http://data.worldbank.org/indicator/IT.NET.USER
75. Yang, Y., Zhu, B. B., Guo, R., Yang, L., Li, S., Yu, N. A comprehensive human computation framework: with application to image labeling. In *Proc. Multimedia*
76. Yuen, M., Chen, L., and King, I. 2009. A Survey of Human Computation Systems. In Proc CSE 2009