REVIEW ARTICLE

The Evolution of Human Computation: Past, Present and Future

Man-Ching Yuen[1]

[1] Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China

Abstract: Human computation is a technique that utilizes human abilities to perform computation tasks that are difficult for computers to process. Since the concept of human computation was introduced in 2003, many applications have been developed in an ad hoc manner and a large number of published works have focused on different aspects of human computation, such as computational techniques and performance analysis. To understand how to apply human computation to solve different problems today, it is important to understand how human computation evolves and why it happens. In light of the achievements of human computation so far, we consider what the future may hold for this relatively new field.

Keywords: human computation; human information processing; human factors

1. Introduction

As a result of significant advances in computing technology, computers can now solve many problems that previously proved difficult for computer programs. However, many tasks that are trivial for humans continue to challenge even the most sophisticated computer programs. We classify the tasks into three categories:

Annotation-Although computers can store and process digital files in various data formats, e.g., text, images, sound clips, videos and web pages, they still have trouble with basic audio or visual tasks like reading distorted text or locating a simple object in an image. By contrast, humans can understand and analyze everyday images easily, such as identifying objects in an image and indicating where they are located in the image[1]. Hence, annotation tasks that humans can perform easily are difficult for computer programs.

Possessing the knowledge about the real world-Basic facts about the world that the majority of humans accept as truths, such as "water is liquid," are called commonsense knowledge. It is impossible for computer programs to have such knowledge without manual input of the relevant facts. However, incorporating commonsense knowledge into computer programs is a tedious process [1]. Humans can describe physical locations in detail by supplying photographic content, descriptive content, "feelings" and emotional words [2], but computer programs are obviously unable to perform such tasks.

Indicating human preferences-Since human preferences are subjective, computer programs cannot predict them accurately. A group of people may have different preferences for the same thing. Eliciting human preferences for large datasets and creating rankings based on those preferences has many practical applications in community-based websites. For example, in the case of image searches, knowing which images are more appealing would allow a search engine to display those images first; and in the field of computer vision, such data could be used to train algorithms that assess the quality of images automatically [3].

1.1 Definition of human computation

Human computation comprises computations carried out by humans, where the computation is the mapping process

Copyright © 2018 Man-Ching Yuen
doi: 10.18063/phci.v1i2.805
This is an open-access article distributed under the terms of the Creative Commons Attribution Unported License (http://creativecommons.org/licenses/by-nc/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

of some input representations to some output representations. Human computation systems are systems that humans interact with to perform human computations. Quinn and Bederson [4] used the definition in von Ahn's 2005 dissertation entitled "Human Computation" [5], which defines human computation as "a paradigm for utilizing human processing power to solve problems that computers cannot yet solve"[5].

2. In the past

2.1 Chronology of works on human computation

In this section, we discuss the research works that influenced later projects in the field of human computation. Table I provides a brief chronology of the works on human computation.

Humans were the first "computers" in the world. Before computers were capable, human beings were recruited as "human computers" to perform mathematical computations for the division of labor and mass production. The first example of human computation is the human computers of the 18th century. Later, "automatic computers" were developed.

In the 20th century, humans were needed again for "AI-complete" tasks [11]. A number of early projects tried to solve many difficult AI problems by using the power of computers and their users around the world. The main characteristic of the projects was that humans performed tasks for financial gain. One notable project, called CYC, was initiated by Douglas B. Lenat in 1995. CYC was an expert system with a domain that covered everyday objects and actions. Humans were paid to input information [12]. However, payment for information was a significant cost in building up the system. Moreover, the shortage of human contributions led to encoding viewpoints or statements that required subsequent modifications [13].

To reduce the cost of human computation applications, it was suggested that users should be asked to perform tasks voluntarily. The first project based on this idea was called the Open Mind Initiative [14-17], a world-wide collaborative effort initiated by the MIT Media Lab in 1999. The objective was to collect commonsense knowledge from people to develop intelligent software. Over eighty hundred thousand commonsense data entries were collected from some 20,000 contributors between 1999 and 2004. A potential drawback of asking unpaid volunteers to perform tasks is that the system is too reliant on the willingness of the participants to donate their time. In our view, in order to motivate contributors, these systems need to provide some social benefit to contributors. Nevertheless, many later projects still rely on unpaid volunteers to contribute information to their systems [7; 18-20]. On the other hand, a key advantage of the mass collaboration approach is that volunteers can make contributions with little or no training. Moreover, the true error rate of data collected through mass collaboration is upper bounded [21-22].

In 2003, Luis von Ahn *et al*. pioneered the concept of human computation to distinguish humans from computers. They developed a computer program called CAPTCHA, which was an automated test that was easy for humans, but difficult for current computer programs. It has many practical security applications, such as preventing dictionary attacks requesting free email services. For instance, Yahoo! uses a CAPTCHA program to prevent bots from registering for free email accounts. It asks users to read a distorted word because current computer programs cannot read distorted text as well as humans [7]. After CAPTCHA was launched, the concept of human computation started to attract the attention of researchers. Luis von Ahn *et al*. were aware of two drawbacks in systems that rely on paid or unpaid humans to contribute information. The first drawback is that the systems are difficult to scale up. As mentioned earlier, hiring humans to perform tasks requires financial resources; and, in the long term, it is difficult to motivate unpaid volunteers to contribute information. The second drawback is that there is no mechanism to guarantee the information provided is correct. Humans can make mistakes or even input incorrect information deliberately. In principle, humans can be paid to verify the accuracy of completed tasks, but payment is not a permanent solution.

To collect accurate information input by humans and build a scalable system that can obtain information from manual input, Luis von Ahn and his colleagues developed the ESP game [8], which was the first human computation game. They started collecting labels for images on theWeb in 2004. Human computation games, also called "Games

with a Purpose (GWAP)", are interactive systems in a game format that collect players' manual input [8]. People around the world spend billions of hours playing computer games each year [9]. Human computation games provide an effective way to channel the human effort spent playing games to solve problems that are easy for humans, but difficult for computers [23]. People play human computation games because they are fun. An added advantage is that accurate information can be collected because the data verification mechanism is built into the game rules. The ESP game was subsequently licensed by Google as the Google Image Labeler (Google Image Labeler website: http://images.google.com/imagelabeler/) in 2006. As of July 2008, 200,000 players had contributed more than 50 million labels for images [6]. The success of the ESP game has motivated researchers to develop other human computation games with various objectives, such as annotating audio clips [24-25] and collecting human preferences [3].

In addition, by outsourcing a task that is traditionally performed by an employee to Internet users, a company can reduce production costs and make more efficient use of labor and resources. The first crowdsourcing system, Amazon Mechanical Turk, was launched in 2005. A crowdsourcing operation involves requesters and workers. A requester submits a task request; a worker selects and completes the task; and the requester only pays the worker if the task is completed successfully. In addition, the requester pays Amazon a certain percentage of the amounts paid to workers for successfully completed HITs.

The paradigm of human computation has been employed in several real-world scenarios, such as image analysis, speech recognition and natural language processing. However, there was no basic framework for the design of systems until Gentry et al. proposed a framework for secure distributed human computation in 2005[26]. Gentry et al. [26] published the first work on the performance analysis of human computation systems and the algorithms used. In the performance analysis, standard probability theory was used to evaluate the security and robustness of systems to malicious attacks; and for the algorithms, a suite of design principles was derived based on standard decision-theory concepts.

Year	Event	Project	Objective
The 18th century	Humans were the first "computers" who performed mathematical computations before computers were capable	Human beings were recruited as "human computers"	Mathematical computations
1995	The first human computation project that asked humans to perform tasks for pay	CYC[12]	Cognizing commonsense knowledge
1999	The first human computation project that asked humans to perform tasks voluntarily	Open Mind Initiative[14-17]	Cognizing commonsense knowledge
2003	Luis von Ahn proposed his first human computation project	CAPTCHA[7]	Distinguishing humans from computers in a security measure
2004	Luis von Ahn proposed the first GWAP (Games with a Purpose)	ESP game[8]	Annotating images
2005	The first crowdsourcing system, Amazon Mechanical Turk, was launched publicly	Amazon Mechanical Turk	Providing a platform for outsourcing tasks to the crowd
2005	The first work to propose a framework, conduct performance studies and design algorithms for use in human computation tasks that focus on security issues	Secure Distributed Human Computation[26]	Developing the design principles of human computation applications
2006	Google launched the Google Image Labeler, which was inspired by the ESP game	Google Image Labeler	Annotating images

Table 1. A Brief Chronology of Works on Human Computation

3. At present

3.1. The significance of human computation

Before the concept of human computation was introduced, traditional approaches for solving problems that are difficult for computers but trivial for humans focused on improving artificial intelligence algorithms [6]. However, in many cases, the algorithms could not solve the problems in a satisfactory manner. The concept of human computation was developed by L. von Ahn in 2003. von Ahn and his colleagues used the concept of human computation to develop a test called CAPTCHA[7] which differentiates between humans and computers. In 2004, von Ahn and Dabbish developed the first human computation game, the ESP game [8] for collecting labels for images. The introduction of the ESP game induces the rise of human computation. The use of human computation for problem solving is becoming more popular today because of the following two factors:

Improving the performance of search engines-It is easy to search for information on the WorldWideWeb by inputting keywords to search engines. Alexa (http://www.Alexa.com) lists Google (http://www.google.com/) as the Internet's most visited website (http://www.alexa.com/siteinfo/google.com). According to NetMarketShare (https://netmarketshare.com/search-engine-market-share.aspx), Google's worldwide popularity peaked at 71.98% in August, 2018. The market share of other search engines was as follows: Baidu (http://www.baidu.com/) (14.04%), Bing (http://www.bing.com) (7.76%) and Yahoo! (http://www.yahoo.com/) (4.44%). By August 2018, Google ran thousands of servers worldwide, processing millions of search requests every day; and about one petabyte of data were generated by users every hour. Google continues to conduct millions of searches every day. The company's success - its competitive advantage - lies in its ability to find meaning in massive sets of data. The larger the data sets, the greater the number of potential meanings that can be derived and the better the search results. It is necessary to construct a comprehensive database for a search engine, but computers still have trouble labeling images, music and videos. By contrast, humans can understand and analyze such content easily. Therefore, human computation can help to evaluate and improve the searching results of search engines.

Bringing human intelligence to computers - To create truly intelligent systems, it is necessary to collect knowledge from humans [9]. An example is reCAPTCHA, which is based on the concept of CAPTCHA. CAPTCHAs are security measures that are widely used on the World Wide Web to prevent automated programs from abusing online services. They do so by asking humans to perform tasks, such as deciphering distorted characters, which computers cannot perform in a satisfactory manner. Everyday, users around the world type more than 100 million CAPTCHAs, each of which takes a few seconds to type. In aggregate, this amounts to hundreds of thousands of hours per day. The reCAPTCHA system uses such human efforts to help digitize old printed materials. Users are asked to decipher scanned words from old materials that computerized optical character recognition (OCR) systems cannot recognize, for example, issues of the New York Times published between 1851 and 1980. reCAPTCHA has been deployed on more than 40,000 websites and it has transcribed over 440 million words. The system achieved an accuracy rate of 99.1% at the word level (216 errors out of 24,080 words) in September 2008, whereas the accuracy of standard OCR was only 83.5% (3,976 errors) [10]. Because of the ubiquity of the Internet, the number of users and web applications continues to increase rapidly. Between 2000 and 2009, the number of Internet users grew by 380%, so the demand for new applications is increasing as a consequence. A large number of previous works in human computation have been proposed [30-31].

3.2. Limitation of human computation

In this section, we discuss the limitations of human computation systems that we should consider. First, human computation requires humans to participate in computation tasks. However, how to motivate humans is a major challenge in the design of human computation systems. Workers have to be motivated in the right way. For example, at Amazon Mechanical Turk, workers are willing to contribute their efforts because they can work on tasks to earn money anywhere, anytime. The workers are simply motivated by the monetary reward. It is impossible to ask workers to perform explicit computation tasks without pay at Amazon Mechanical Turk. In addition to the right motivation method,

the degree of motivation is important. Consider a time-consuming translation task on Amazon Mechanical Turk. It is difficult to find a worker to accept the task if the pay is too low, but some workers are if the pay is high enough. Moreover, Heyman and Ariely have shown that for tasks that require mental skills, increased payments might cause degradation in performance [27].

Second, the quality of the output aggregated in a human computation system relies heavily on the task design in the system. The design of a task has a major impact on the quality of the output aggregated in human computation systems. In February 2007, Penguin Books and De Montfort University launched a project called "A Million Penguins." Seeded with one sentence, the novel entitled "A Million Penguins" was open to anyone who wanted to help write it in a wiki. Within a five-week period, more than 11,000 edits were made to over 1,000 pages of the wiki. The project leaders concluded that the project did not produce a novel [28]. Rohwer observed that the project lacked sufficient control to produce the necessary integration and structure for the novel, and that led to the failure of the project [29]. However, it is possible for a human computation system to aggregate a literary novel in another way. Given the title and the first sentence of a novel, a system can call for submissions from a community. Members of the community can submit novels they have written, and all members can vote their favorite novels. The novel with the highest number of votes is the output aggregated in the system. In this case, each novel is written by one person, so no post-processing is needed to produce a successful novel. This example demonstrates that it is necessary to take great care in designing tasks in a human computation system so that the results are useful and valid.

Third, the output aggregated in a human computation system may not meet the expectations of all people. Some human computation tasks aim to generate creative ideas, but the output aggregated in the system may not satisfy the expectations of all people. Consider the t-shirt designs collected in Threadless. Threadless can never fully claim that a t-shirt design that wins in a competition is wanted by all people. In fact, graphic logos and images are perceived differently in different cultures, religions and groups. Having experiences on designing clothing items, professional designers can create great t-shirt designs that match the target style. Therefore, the quality of the output designed by creative professionals may be better than the quality of the output aggregated in a human computation system. However, a company may prefer individual workers in the crowd to employees because the costs are lower. Compared to the profits a company makes on the sale of its crowd-made products, the amount paid in prize money to the winning designers is quite small. In this case, the company may bypass legislation regarding workers' rights and wages, so new employment regulation is needed to support this situation. Finally, some human computation tasks require specific human skills. Everyone has specific skills and cannot work on all kinds of human computation tasks. Although humans can develop a certain number of skills over time, they cannot master all skills at once. Besides, different individuals have different levels of the same skill. At Amazon Mechanical Turk, some tasks are not chosen by workers because the tasks require some specific skills, e.g., translating documents into Spanish.

4. In the future

In the future, more intelligent applications will be developed with the aid of the data collected from human computation systems. The following are some possible applications of human computation on the Internet:

We can collect commonsense knowledge to train computer programs so that they are more intelligent. The objective is to enable the programs to provide accurate answers to basic questions automatically.

A knowledge-based translation system is another application that needs human knowledge to complete the translation process accurately and concisely.

When people search the Internet for information about a geographical location and the surrounding buildings; for example, they may be looking for a good restaurant. Such information must be collected by humans.

5. Conclusion

To provide an overview of human computation, we have compiled a brief chronology of the major research works that influenced the development of the field of human computation. We have described the significance of human

computation at the present. As there are some limitations of human computation systems that should be considered, we have discussed the problems that have to be addressed. The article provides a better understanding of human computation systems and should facilitate future research and the development of applications. We believe that future research will focus on how to enhance human computer interaction so that more diverse problems can be solved at a lower cost by applying the concept of human computation.

References

1. von Ahn, L. Games with a purpose. Computer. 2006; 39: 92–94.
2. Casey, S., Kirman, B., Rowland, D. The gopher game: A social, mobile, locative game with user generated content and peer review. In ACE '07: Proceedings of the International Conference on Advances in Computer Entertainment Technology. ACM, 2007: 9–16.
3. Hacker, S., von Ahn, L. Matchin: Eliciting user preferences with an online game. In CHI '09: Proceedings of the 27th International Conference on Human Factors in Computing Systems. ACM, 2009: 1207–1216.
4. Quinn, A. J., Bederson, B. B. Human computation: a survey and taxonomy of a growing field. In CHI '11: Proceedings of the 2011 Annual Conference on Human Factors in Computing Systems. ACM, 2011: 1403–1412.
5. von Ahn, L. Human computation. Ph.D. thesis, Carnegie Mellon University. 2005.
6. von Ahn, L., Dabbish, L. Designing games with a purpose. Communications of the ACM, 2008; 51(8): 58–67.
7. von Ahn, L., Blum, M., Hopper, N. J., *et al*.. CAPTCHA: Using Hard AI Problems for Security. In EUROCRYPT '03: Proceedings of the 22nd International Conference on Theory and Applications of Cryptographic Techniques. Springer, 2003: 294–311.
8. von Ahn, L., Dabbish, L. Labeling images with a computer game. In CHI '04: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2004: 319–326.
9. von Ahn, L., Ginosar, S., Kedia, M., *et al*. Improving accessibility of the web with a computer game. In CHI '06: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2006: 79–82.
10. von Ahn, L., Maurer, B., Mcmillen, C., *et al*.. reCAPTCHA: Human-Based Character Recognition via Web Security Measures. Science. 2008: 321, 5895, 1465–1468.
11. Grier, D. A. When computers were human. Princeton University Press. 2005.
12. Lenat, D. B. CYC: A large-scale investment in knowledge infrastructure. Communications of the ACM. 1995; 38(11): 33–38.
13. Friedland, N. S., Allen, *et al*.. Project halo: Towards a digital aristotle. AI Magazine. 2004; 25(4): 29–48.
14 . Stock, D., C.P., L. Open mind animals: Insuring the quality of data openly contributed over the world wide web. In Proceedings of AAAI Workshop on Learning from Imbalanced Data Sets. AAAI Press, 2000; 4–9.
15. Stork, D. G. The open mind initiative. IEEE Expert Systems and Their Applications. 1999; 14(3): 19–20.
16. Stork, D. G. Open data collection for training intelligent software in the open mind initiative. In EIS '00: Proceedings of the Engineering Intelligent Systems Symposium. 2000a: 1–7.
17. Stork, D. G. Using open data collection for intelligent software. Computer. 2000b; 33(10): 104–106.
18. Liu, H. and Singh, P. Conceptnet: A practical commonsense reasoning toolkit. BT Technology Journal. 2004; 22: 211–226.
19. Havasi, C., Speer, R., and Alonso, J. Conceptnet 3: A flexible, multilingual semantic network for common sense knowledge. In RANLP '07: Proceedings of Recent Advances in Natural Language Processing. 2007.
20. Chung, H. Globalmind - bridging the gap between different cultures and languages with common-sense computing. M.S. thesis, Massachusetts Institute of Technology. 2006.
21. Lam, C. P. and Stork, D. G. Evaluating classifiers by means of test data with noisy labels. In IJCAI'03: Proceedings of the 18th international joint conference on Artificial intelligence. Morgan Kaufmann Publishers Inc., 2003: 513–518.
22. Richardson, M., Domingos, P. Building large knowledge bases by mass collaboration. In K-CAP '03: Proceedings of the 2nd international conference on Knowledge capture. ACM, 2003: 129–137.
23. Dasdan, A., Drome, C., Kolay, S., *et al*.. Thumbs-up: a game for playing to rank search results. In HCOMP '09: Proceedings of the ACM SIGKDD Workshop on Human Computation. ACM, 2009: 36–37.
24. Law, E., Von Ahn, L. Input-agreement: a new mechanism for collecting data using human computation games. In CHI '09: Proceedings of the 27th international conference on Human factors in computing systems. ACM, 2009: 1197– 1206.
25. Barrington, L., O'malley, D., Turnbull, D., *et al*.. User-centered design of a social game to tag music. In HCOMP '09: Proceedings of the ACM SIGKDD Workshop on Human Computation. ACM, 2009: 7–10.
26. Gentry, C., Ramzan, Z., Stubblebine, S. Secure distributed human computation. In EC '05: Proceedings of the 6th ACM conference on Electronic commerce. ACM, 2005: 155–164.
27. Heyman, J., Ariely, D. Effort for Payment. A Tale of Two Markets. Psychological Science. 2004; 15(11): 787–793.
28. Mason, B., Thomas, S. A million penguins research report. Tech. rep., Institute of Creative Technologies, De

Montfort University, Leicester, UK. 2008.
29. Rohwer, P. A note on human computation limits. In HCOMP ’10: Proceedings of the ACM SIGKDD Workshop on Human Computation. ACM, 2010: 38–40.
30. Yuen, M.-C., Chen, L.-J., King, I. A survey of human computation systems. In CSE ’09: Proceedings of IEEE International Conference on Computational Science and Engineering. IEEE Computer Society, 2009: 723–728.
31. Yuen, M.-C., King, I., Leung, K.-S. A survey of crowdsourcing systems. In SocialCom ’11: Proceedings of The Third IEEE International Conference on Social Computing. IEEE Computer Society, 2011: 766–773.