Research Proposal *Logan Gilmour*

The Internet is a new frontier in human collaboration. One need look no further than Wikipedia to see the potency of crowdsourcing. A crowdsourcing system uses humans to solve a problem devised by the owners of the system [4]. The specific type of crowdsourcing I am interested in researching is human computation. Human computation is ridged in its use of human effort, incorporating it as part of a calculation organized and performed by a computer [6]. The tasks delegated to humans are typically ones that are currently not solved reliably or efficiently enough by A.I. techniques, but are easily solved by humans.

Human Computation can be thought of as "Programming the Global Brain"[1]. However, this resource cannot be naively programmed in the same way that a traditional computer is. Humans require motivation in order to perform tasks. 'Games with a purpose' attempt to provide intrinsic motivation, by presenting the desired human computation task within a game [8]. The translation of work tasks into games has been called 'gamification'. However, many gamification efforts have been derided by those in game studies as 'pointsification' [5]. This refers to the practice of adding a scoring system to a task on the assumption that this will be sufficient to make the task intrinsically motivating. The reality, however, is that blindly adding external evaluation (in the form of points) to a task can actually *harm* that task's intrinsic motivation [5]. I suggest that 'pointsification' is symptomatic of a lack of systematic way to translate a desired computation into an intrinsically motivating task. Adding a scoring system can be accomplished systematically, so this is what is the extent of what is commonly attempted in order to make a task fun.

My proposed approach to the problem of 'pointsification' is the development of a set of design patterns for transforming human computation tasks so that they are intrinsically motivating. In software, design patterns are generalizations of common solutions to architecture problems [9]. These generalizations are reusable when designing software systems, and give software architects a vocabulary with which to discuss larger systems. However, software design patterns are specific to pure-software systems. I plan to identify a set of design patterns for architecting systems of tasks into a game.

There are existing heuristics for the evaluation of the enjoy-ability of video games. Many of these have been identified as having large overlap with Flow theory [7]. This theory suggests that activities that are not too difficult (provoking anxiety) and not too easy (causing boredom) and that provide immediate feedback allow the mind to enter a state of Flow. Flow is intrinsically rewarding, to the extent that people will make significant sacrifices to do activities that produce it [3]. I hope to identify reusable patterns that allow the architect of a human-computation system to increase or decrease difficulty of human computation tasks, and increase feedback. With the ability to tune these variables, it should be possible to make a variety of tasks self motivating.

The ESP game is a good example of self-motivating human computation [8]. In this game, two players are connected randomly across the internet. Their goal is to type the same word, and their only shared context is a single image. It turns out that their guesses are good labels for these images. Labeled images have a variety of uses, and A.I. approaches are not yet sufficiently accurate for many of them. It is unlikely that a human would agree to simply label images for free, but the players of ESP did just that: within the first four months of the game being made available online, 13 630 players created 1 271 451 labels [8]. By taking an easy task (labeling images) making it more difficult (requiring two players to choose the same label), and adding feedback (the two players verify each other), the ESP game makes image labeling fun. I hope to identify general ways to effect such transformations in more complex human computation systems. To evaluate the discovered patterns, I plan to conduct user studies on practical human-computation tasks (such as producing navigation maps for the insides of buildings) before and after 'gamifying' them, comparing the extent to which players reported that they were fun.

If I am successful, it should be significantly easier to create robust human computation systems that use human resources more optimally. Even if the application of the discovered design patterns are insufficient to make a task intrinsically motivating, it could help to make it more desirable (and elicit higher quality work) on payed human computation markets such as Amazon Mechanical Turk, where intrinsic motivation has been shown to positively affect both quantity and quality of work [2]. In cases where tasks need to be more difficult to be engaging, additional cognitive effort could be directed toward solving other problems common to human computation systems, such as cheat-resistance [6].

References

[1] Abraham Bernstein, Mark Klein, and Thomas W. Malone. Programming the global brain. *Commun. ACM*, 55(5):41–43, May 2012.

[2] Dana Chandler and Adam Kapelner. Breaking monotony with meaning: Motivation in crowdsourcing markets. *Journal of Economic Behavior & Organization*, 90(0):123 – 133, 2013.

[3] Mihaly Csikszent. *Flow*. HarperCollins, 1991.

[4] Anhai Doan, Raghu Ramakrishnan, and Alon Y. Halevy. Crowdsourcing systems on the world-wide web. *Commun. ACM*, 54(4):86–96, April 2011.

[5] Scott Nicholson. A user-centered theoretical framework for meaningful gamification. *Proceedings GLS*, 8, 2012.

[6] Alexander J. Quinn and Benjamin B. Bederson. Human computation: a survey and taxonomy of a growing field. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1403–1412, New York, NY, USA, 2011. ACM.

[7] Penelope Sweetser and Peta Wyeth. Gameflow: a model for evaluating player enjoyment in games. *Computers in Entertainment (CIE)*, 3(3):3–3, 2005.

[8] Luis Von Ahn. Human computation. In *Design Automation Conference, 2009. DAC'09. 46th ACM/IEEE*, pages 418–419. IEEE, 2009.

[9] Pree Wolfgang. *Design patterns for object-oriented software development*. Reading, Mass.: Addison-Wesley, 1994.