Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2011 Proceedings - All Submissions

8-7-2011

Evaluating Design Solutions Using Crowds

Jin Bao
Stevens Institute of Technology, jbao@stevens.edu

Yasuaki Sakamoto
Stevens Institute of Technology, ysakamot@stevens.edu

Jeffrey V. Nickerson
Stevens Institute of Technology, jnickerson@stevens.edu

Follow this and additional works at: http://aisel.aisnet.org/amcis2011_submissions

Recommended Citation
Bao, Jin; Sakamoto, Yasuaki; and Nickerson, Jeffrey V., "Evaluating Design Solutions Using Crowds" (2011). *AMCIS 2011 Proceedings - All Submissions.* Paper 446.
http://aisel.aisnet.org/amcis2011_submissions/446

This material is brought to you by AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2011 Proceedings - All Submissions by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Evaluating Design Solutions Using Crowds

Jin Bao
Stevens Institute of Technology
jbao@stevens.edu

Yasuaki Sakamoto
Stevens Institute of Technology
ysakamot@stevens.edu

Jeffrey V. Nickerson
Stevens Institute of Technology
jnickerson@stevens.edu

ABSTRACT

Crowds can be used to generate and evaluate design solutions. To increase a crowdsourcing system's effectiveness, we propose and compare two evaluation methods, one using five-point Likert scale rating and the other prediction voting. Our results indicate that although the two evaluation methods correlate, they have different goals: whereas prediction voting focuses evaluators on identifying the very best solutions, the rating focuses evaluators on the entire range of solutions. Thus, prediction voting is appropriate when there are many poor quality solutions that need to be filtered out, and rating is suited when all ideas are reasonable and distinctions need to be made across all solutions. The crowd prefers participating in prediction voting. The results have pragmatic implications, suggesting that evaluation methods should be assigned in relation to the distribution of quality present at each stage of crowdsourcing.

Keywords

Crowdsourcing, creativity, human-based genetic algorithms, evaluation, Mechanical Turk.

INTRODUCTION

Crowds can be used to both generate and evaluate design. There are many ways to instruct the crowd: which ways are best suited for particular types of problems? The question is important, because the use of the crowd to design solutions is becoming popular. The Internet makes possible a different kind of infrastructure to utilize the wisdom of crowds (Surowiecki 2004): researchers have successfully conducted large scale experiments using crowds, in which humans perform tasks akin to computers performing processing (von Ahn and Dabbish 2008; Kittur, Chi and Suh 2008; Little, Chiton, Goldman and Miller 2010; Raykar, Yu, Zhao, Valadez, Florin, Bogoni and Moy, 2010; Quinn and Bederson, 2011). The term crowdsourcing (Howe 2006) has been coined to describe this activity, which allows for the collection of solutions from large crowds with diverse background across time and space (Wagner and Back 2008).

This work can be seen in many ways as extensions of long-standing concerns with coordination in the information systems research (Crowston 1997; Montoya-Weiss, Massey and Song 2001; Dennis and Williams 2003; Paul, Haseman and Ramanurthy 2004; McKnight, Choudhury and Kacmar 2004; Malone, Laubacher and Dellarocas 2010). Both the generators of the output and the evaluators are working toward a common goal, and in this sense we can characterize them as forming a loosely-coupled virtual team, albeit a team that engages in a minimal form of collaboration. The lack of two-way communication between participants may be an asset: it might, perhaps, ameliorate some of the problems of group activity, such as production blocking (cf. Gallupe et al., 1992). Thus, crowdsourcing allows for a new kind of organization structure that has different characteristics from teams that have been investigated in the past: it is loosely coupled, like electronically mediated virtual teams, while at the same time allowing for fast assembly and massive scale (Nickerson and Sakamoto 2010; Nickerson, Sakamoto and Yu 2011).

With such enormous potential, it is important to systematically study the building blocks of larger systems, and in particular how the crowd can best evaluate the collaborative output of many hundreds of people. Here, we compare two evaluation methods, Likert scale rating and prediction voting, that can be used in a large-scale crowdsourcing system. In the Likert scale rating method, participants are asked to evaluate the creativity of solutions on a five-point Likert scale, and the average score for each solution is recorded as the fitness value of that specific solution. In the prediction voting method, participants are asked to predict whether each solution is the winner of the creativity award or not, and the number of votes for each solution is recorded. The two evaluation methods we compare in the present work combine features of voting, consensus, averaging, and prediction markets that are commonly used methods in crowd evaluation (Malone, Laubacher, and Dellarocas 2010).

We evaluated the evaluation methods in the context of a system we built before (Nickerson and Sakamoto 2010; Nickerson et al. 2011). In this paper, we first describe the design of our crowdsourcing system, and the way the two alternative evaluation

methods fit in. Then we present the methodological framework of our study, followed by the evaluation results. Finally we discuss these results, and suggest ways the research can be extended.

BACKGROUND

Creativity

In this study, we examine creativity in crowds. Researchers more or less agree that a creative product is one that excels on two dimensions: the exact description of the dimensions varies slightly, for example, novel and useful (Mumford, 2003), or original and valuable (Amabile 1996). In much creativity research, originality and practicality are the terms used (Runco and Pritzker, 1999; Ward et al., 1997), and we will continue the tradition.

Two processes underlie the creation of original and practical solutions: generation, and exploration (Finke, Ward and Smith, 1992). For example, the Geneplore model (Finke, Ward and Smith, 1992) suggests that generating a diverse set of candidate solutions would be useful for creating something original. During exploration, solutions can be combined, in which poor features are left out, and good features are synthesized, increasing the practicality of the combined solutions. In order to combined ideas, there needs a selection phase, in which the two ideas to be combined are selected.

Generation and exploration are akin to divergent thinking and convergent thinking (Guilford, 1967; Osborn, 1957).

In our crowdsourcing system, we create different roles of crowds that collectively complete the generation process and selection process. Specifically, we implement a system based on genetic algorithms. Genetic algorithms (GA) are search procedures based on the mechanics of natural selection and genetics; they use machines to generate and evaluate solutions to perform intelligent crossover by applying selection operators to existing solutions (Goldberg 1989). Genetic algorithms that use human judgment to evaluate solutions are known as interactive genetic algorithms: this paper focuses on the interactive component, in which the crowd evaluates and steers the evolution.

Fitness Function

A fitness function is an objective function that evaluated the optimality of each solution in a genetic algorithm. Fitness is important to determine: in genetic algorithms, the fitter solutions are more likely to breed. In interactive genetic algorithms, human judgment stands in as a fitness function (Bentley and Corne 2002; Banerjee, Quiroz and Louis, 2008). In our crowdsourcing system humans are also utilized to evaluate solutions. However there are several differences between the fitness functions in interactive genetic algorithms and in our system. First of all, the most significant problem in human evaluation in interactive genetic algorithms is the fatigue problem, caused by overload (Takagi, 1998). The fatigue problem, however, has been greatly reduced in our crowdsourcing system. One of reasons is that our crowdsourcing system enables the evaluators to choose the number of evaluations they will perform: when they get fatigued they stop. Another reason is that the overall staffing structure for evaluation is different. In interactive genetic algorithms evaluation tasks are assigned to participants through the hiring of a small number of contractors. In contrast, in our crowdsourcing system evaluation tasks are undertaken by hundreds of participants online. The last and the most important difference is that whereas a machine generates the solutions in need of evaluation in IGA, in our system human beings generate the solutions. Because of these differences, more work on testing fitness functions is called for.

A Crowdsourcing System

Our current crowdsourcing system is an online human-based genetic algorithm system (cf. Kosorukoff, 2001). It is an infrastructure as shown in Figure 1: a way to organize the flow of information between agents performing different functions that earlier were internal functions of genetic algorithms.

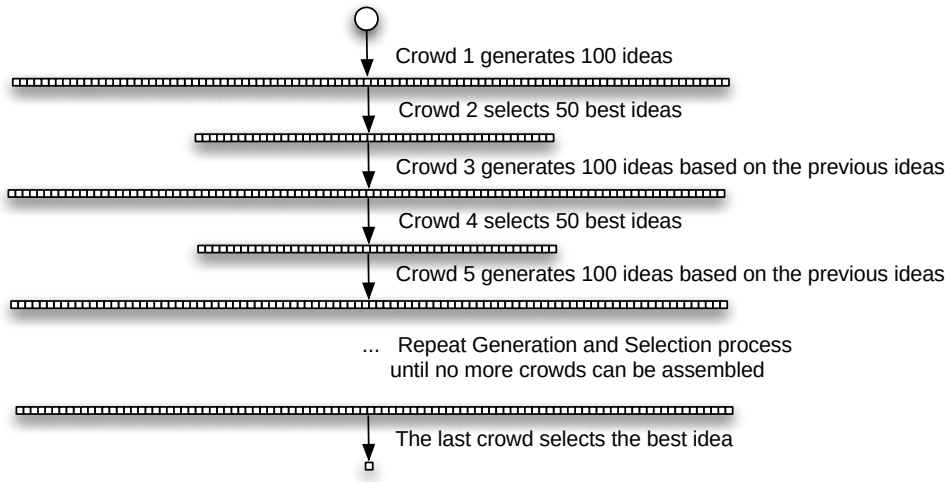

Figure 1. A Crowdsourcing System, showing multiple crowds performing different steps through many generations.

Once we have the new fitness function defined, the crowdsourcing system proceeds to request humans to generate a population of solutions, and then improve it through repetitive application of reproduction (in this paper we test with a combination strategy) and selection operators. In our previous studies, five groups of crowds collectively generated, evaluated, and combined solutions (Nickerson et al. 2011). For example, our crowdsourcing system generated solutions for the oil spill problem in Mexico Gulf. We use ideas generated in these past studies to test two evaluation methods. The crowdsourcing system used in the current work is explained below.

Generation

Initially many individual solutions are generated to form an initial population. For example, in many experiments, 100 solutions, a typical starting population size, are generated (cf. Deb 1999).

Selection

During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected based on the fitness function, where fitter solutions are typically more likely to be selected. In our system, we utilize tournament selection (Fogel 2006; Goldberg 1989). The tournament selection rates the fitness of each solution and preferentially selects the best solutions. While evaluating solutions, the human evaluators are not aware of each other's designs or choices, only of the two designs presented, eliminating the possibility that the participants are swayed by vocal group members (e.g. Asch 1951).

Combination

The next step is to generate a second-generation population of solutions from those selected through a genetic operator that performs a form of crossover we refer to as *combination*. Combination has contributed to the success of genetic algorithms in optimizing many tasks (e.g., Fogel, 2006; Goldberg, 1989). For each new solution to be produced, a pair of parent solutions is selected for breeding from the pool selected previously. By producing a child solution through combination, a new solution is created which typically shares many of the characteristics of its parents. New parents are selected for each new child, and the process continues until a new population of solutions is generated. A novel feature of our crowdsourcing system is that humans perform this combination step.

Termination

The generational process is repeated until a termination condition is reached. There are several common terminating conditions. In our current experiment, we cap the number of crowds.

Evaluation

We test two methods of evaluating solutions: Likert scale rating, and prediction voting.

Likert Scale Rating: Participants are presented with the instruction: "Recently we collected 180 ideas for solving oil spill problems. We need your help to evaluate how original (novel and surprising) this idea is" followed by a five-point Likert scale as indicated in Figure 2. The fitness value each solution gets from one evaluation is a discrete value from -2 to 2. The solution considered as "not at all original/practical" is rated as -2 and the one considered as "very highly original/practical" is rated as 2. Multiple participants evaluate each solution. The fitness value of this specific solution is the average of these evaluation scores.

Not at all original ○○○○○ Very highly original

Figure 2. The five-point Likert scale rating.

Prediction Voting: Participants are presented with the instruction: "Recently we collected 180 ideas for solving oil spill problems. One idea that was most novel and surprising received an originality award" followed by a prediction vote as shown in Figure 3. The value each solution gets from one evaluation is a binary value, where 0 means "not the winner" and 1 means "the winner". Multiple participants rate each solution, and the fitness of each solution is defined as the sum of values for this specific solution.

This idea is ○ the winner ○ NOT the winner.

Figure 3. The prediction voting.

Hypotheses

Goals guide people's actions (Love 2005). Different goals lead to different behaviors. Although rating and voting are both about evaluating the quality of solutions, the evaluators' goals differ in the two tasks; thus, we hypothesize that rating and voting will focus participants on different aspects of evaluation. In particular, the goal of the prediction voting is to identify the award winner. Consequently, subjects will try to select only the best quality idea as a winner, and their responses will be biased toward non-winners for the majority of the ideas that are not the best. In contrast, the goal of the rating task is to place each idea on a scale. As a result, the rating task focuses evaluators on the entire range of solutions. In this case, there should be a few solutions with extreme values, and many ideas around the middle, resulting in a normal distribution.

If this is the case, we can match the task to the expected distribution of the quality of solutions. For example, in the early stages of crowdsourcing, there may be many solutions, the majority of which are of low fitness. Then, applying the prediction voting method to select the fit solutions will help the system by filtering out the majority of low fit solutions. In the later stages when the system matures, there may be few low fit solutions. Then, we do not want to filter out many solutions using the prediction voting method. Instead, we want all solutions to have a chance to produce offspring. To do so, the Lickert scale rating method is appropriate.

METHOD

Sample

In previous studies using our crowdsourcing system, we asked crowds to generate 468 solutions to the oil spill problem in the Mexico Gulf (Nickerson et al. 2011). Another 311 solutions were contributed to the Principal Investigator Association's website, principalinvestigators.org, by professors, scientists and engineers who can be considered experts. For the purpose of evaluation, we randomly sampled 180 solutions out of the 779 total solutions. There were no duplicate solutions in our sample pool.

Design and Procedure

The 180 solutions were evaluated by utilizing the two previously described evaluation methods. Ten different participants rated each solution. Each participant had a choice of rating one or many solutions, but only with one randomly assigned evaluation method.

Once we had the evaluation scores, we simulated two sets of tournament selections based on the fitness values obtained from the two evaluations. In each simulation of tournament selection, two solutions were randomly selected and the fitter one was

chosen. If two ties were selected, another two would be re-selected until the selected were not tied. Each of the two simulations was run 10,000 times. The different patterns of the selected populations resulting from the two evaluation methods were compared.

Population

One hundred and forty-six subjects (87 males) participated in the Likert scale rating evaluation for a nominal stipend. Their ages ranged from 14 to 61 with an average age of 29. On average they spent 21 seconds to complete one rating task. Each participant was required to take a demographic survey.

Another 101 subjects (51 males) participated in the prediction voting evaluation for a nominal stipend. Their ages ranged from 16 to 67 with an average age of 33. On average they spent 15 seconds to complete one voting task.

RESULTS

Correlation Between Likert Scale Rating and Prediction Voting

We first examined the correlation between the fitness values of the 180 solutions based on Likert scale rating and prediction voting. There were significantly positive correlations both in originality ($r = 0.46$, $p < .001$) and practicality ($r = 0.51$, $p < .001$). Excluding 84 solutions rated as 0 in the prediction voting resulted in even stronger correlation: $r = 0.82$, $p < .001$ for originality, and $r = 0.80$, $p < .001$ for practicality. These results show that the solutions evaluated highly in one method tend to be also highly evaluated in the other method.

Evaluation Results

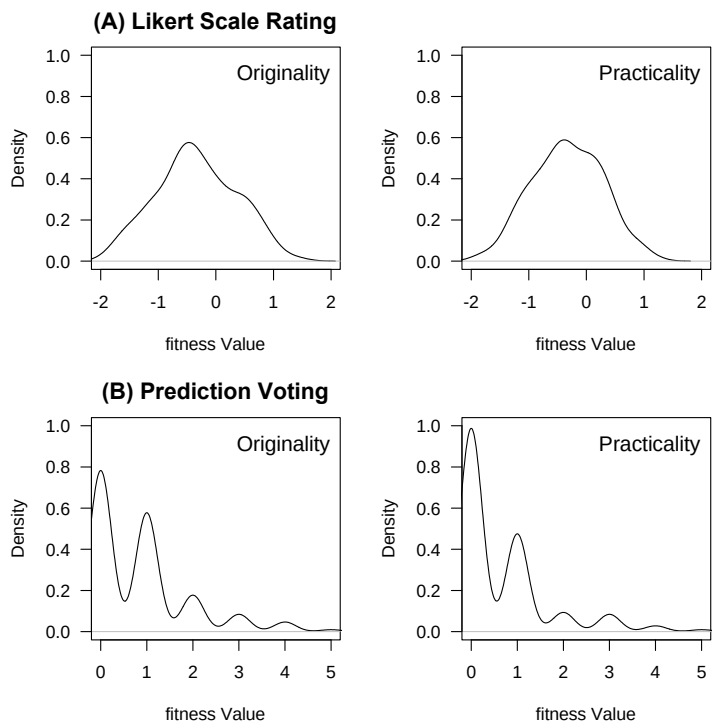

Figure 4. Density of each level of fitness value in the two evaluations (the horizontal axis represented the fitness value. The vertical axis represented the density).

Likert Scale Rating

Figure 4(A) shows the distribution of the fitness values with regard to originality and practicality based on Likert scale rating. The fitness values ranged from -1.9 to 1.5. The median value obtained in both originality and practicality dimensions were less than 0. As Figure 4(A) shows, there were many solutions with median-quality, and at the two extremes, there appear to

be more poor quality solutions than good quality solutions. Using the Likert scale rating, participants tend to consider the highest value less likely to be reached; that is, they are more willing to rate the worst solutions as -2 rather than rating the best solutions as 2.

Participants had distinguished the difference between solutions at a detailed level. The 180 solutions were grouped into 47 different quality levels of solutions between the values -2 to 2.

Prediction Voting

As shown in Figure 4(B), the distribution of fitness values based on prediction voting resulted in different patterns from that based on Likert scale ratings, despite the high positive correlation between the two methods. The prediction voting evaluation leads to many ideas being considered a poor fit, which is advantageous when there are many poor solutions that need to be removed. When solutions evaluated by prediction voting received from no votes to five votes, we found that more than 50% of solutions did not get any votes at all, less than 10% of solutions received more than two votes, and more than one third of solutions were voted only once or twice. Thus, in prediction voting, people tended to vote for the very good solutions and ignore the poor and mediocre solutions. A comparison of Figure 4(A) and 4(B) shows that whereas prediction voting results in many poor solutions and a few good solutions, Likert scale rating leads to many average solutions with a few poor solutions and even fewer good solutions.

In contrast to the results from the Likert scale rating, there were many ties in prediction voting, especially for solutions with no votes; moreover, by definition there are only five levels of solution quality. Within each of the five levels of solutions, this evaluation cannot distinguish quality, so the ratings are quantized.

Simulation Results

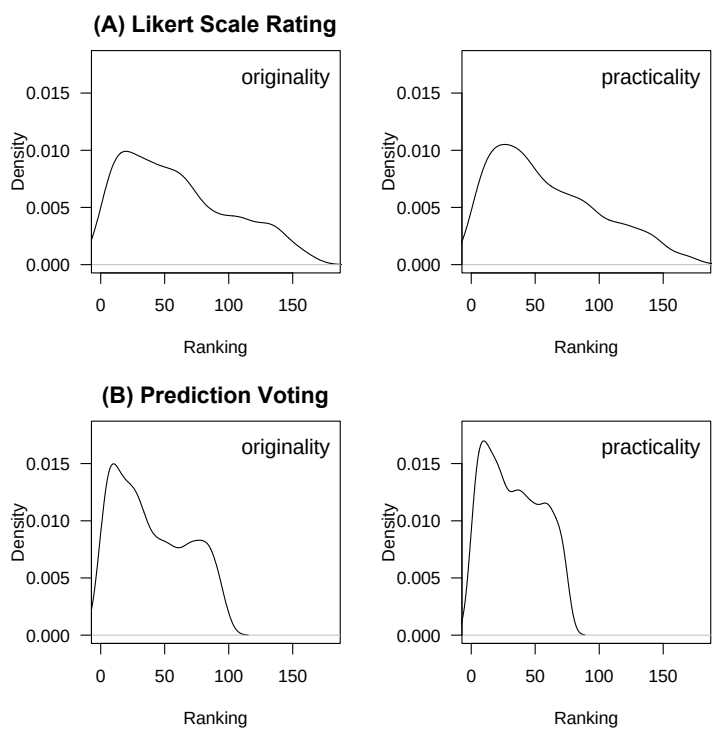

Figure 5. Density of each ranking level in the two simulations (the horizontal axis here was the ranking of all 180 solutions from 1 to 180 with the fittest values on the left. The vertical axis was the density of each ranking level).

Likert Scale Rating

As indicated in Figure 5(A), based on this fitness value from the Likert scale rating evaluation, the simulation of tournament selection became asymmetric. The mass of the distribution was concentrated on the left of the figure. However a proportion

of low fitness values on the right were also selected. The pattern in the originality dimension was the same as with the practicality dimension.

Prediction Voting

As indicated in Figure 5(B), the group of solutions that did not get any votes had been removed by the tournament selection. The selection population was concentrated on the high-ranking solutions. Adjacent to the small peak on the right side of the figure is the group of lower-vote solutions. Those are the solutions that only got voted for once or twice.

Motivation

We posted the two evaluation conditions at the same time. We noticed a difference in the number of tasks participants took on: participants in the prediction voting evaluation were willing to do more. On average each participant in the prediction voting took six more evaluations than the participants asked to do the Likert scale assessments. Moreover, the participants spent six seconds less in the prediction voting than in the Likert scale rating. The prediction voting appeared much easier for participants: they just had to evaluate whether a solution was the winner or not, and did not have to place each solution on a scale. In aggregate, the Likert scale rating evaluation experiment took 26 more hours to complete than the prediction voting. One possible explanation might be that the Likert scale rating took longer to solicit workers because it was less fun to perform.

DISCUSSION

In the current work, we compared two methods of evaluating solutions in a crowdsourcing system. In the prediction voting method, the evaluators' goal was to indentify the winning solution that received an originality or practicality award. In the Likert scale rating method, the goal was to place each solution on a scale. Both methods measured the quality of the solutions: in fact, the correlation between the two evaluations showed that the two evaluation methods were measuring solutions in a similar way. However, the different goals of the tasks led subjects to emphasize different aspects of solution quality, suggesting that each might be appropriate for different situations.

The prediction voting method was designed to pick out the very best solutions. Half the solutions did not get any votes. Our simulation results showed that these sub-best solutions would be eliminated in tournament selection, and thus would not be combined into a new solution. The elimination of many ideas will be useful if early in the crowdsourcing there are many poor quality solutions. However, in later stages when the crowdsourcing system matures and there are few poor quality solutions, the prediction voting method will unnecessarily eliminate good quality solutions. More broadly, there is a tradeoff: this technique will eliminate poor quality solutions and reward the best solutions, but at the cost of losing many of the sub-best but still good quality solutions.

In contrast, the Likert scale rating method resulted in a normal distribution of fitness values: there were a few solutions with extreme values, and many ideas around the middle. The rating task focused evaluators on the entire range of solutions. Thus, this technique was not good for discovering the very best out of the good solutions. Similarly, poor quality solutions were mixed with the mediocre using this rating method. Thus, this method is not appropriate for selecting ideas when there are many poor quality ideas to be eliminated. Our simulation results showed that these poor quality ideas would be selected for combination, even though tournament selection biased the selected population toward solutions with high fitness values. Instead, this technique is suitable when most ideas are good quality and should be included in the mating pool for the purposes of diversity. The tradeoff is the following: whereas most solutions can be differentiated from each other using this method, it is not as easy to filter out the extreme solutions, either the best or the worst ones, because the Likert scale task does not have as much resolution at the extremes as the prediction voting task has.

CONCLUSION

Crowdsourcing is becoming a popular way of creating ad hoc organizations. We need to systematically study the crowdsourcing systems in order to improve their outputs. Here we tested two methods for evaluating the designs produced by crowdsourcing systems. Through the application of evaluation, followed by the use of tournament selection, we have shown that Likert scale rating can broadly distinguish between most designs, a technique that might be best in situations in which the quality level of designs is high, with low variance. Prediction voting is much more efficient for filtering the very best solutions and is therefore appropriate for scenarios in which there are large number of poor quality solutions that need to be filtered out, or situations in which the very best solutions need to be determined. With respect to another aspect of evaluation, objective setting, in this study we evaluated creativity on two dimensions: originality and practicality. Future researchers might consider situations in which there are finer grain objectives (for example, practicality might involve issues of cost, time

to market, or reliability) and determine which evaluation methods are appropriate at different stages of such processes. More generally, this paper makes a foray down a little explored path: alternative ways crowds can evaluate creative output so as to further the collective design effort.

ACKNOWLEDGMENTS

This research was supported by the National Science Foundation, grant IIS-0968561.

REFERENCES

1. von Ahn, L., and Dabbish, L. (2008). Designing games with a purpose. *Communications of the ACM, 51*(8), 58-67.
2. Amabile, T. (1996). *Creativity in context.* Westview Pr.
3. Asch, S. E. (1951). Effects of group pressure upon the modification and distortion of judgments. In H. Guetzkow (Ed.), *Groups, leadership and men*, Pittsburgh, PA: Carnegie Press.
4. Banerjee, AJ. C. Quiroz, and S. J. Louis. A model of creative design using collaborative interactive genetic algorithms. *In Proceedings of the Third International Conference on Design Computing and Cognition*, DCC08, 2008.
5. Bentley, P. J. and Corne, D. W (eds) Creative Evolutionary Systems, SF: Morgan Kaufmann, 2002.
6. Crowston, K. (1997). A coordination theory approach to organizational process design. *Organization Science*, *8*(2), 157-175.
7. Deb, K., Multi-Objective Optimization Using Evolutionary Algorithms, Wiley, 2009
8. Dennis, A., and Williams, M. (2003). Electronic Brainstorming. *Group creativity: Innovation through collaboration*, 160–178.
9. Finke, R. A., Ward, T. B., and Smith, S. M. (1992). *Creative Cognition: Theory, Research, and Application*. Cambridge, MA: MIT Press.
10. Fogel, D. B. (2006). Evolutionary Computation: Toward a New Philosophy of Machine Intelligence, IEEE Press, Piscataway, NJ. Third Edition.
11. Gallupe, R. B., Dennis, A. R., Cooper, W. H., Valacich, J. S., Bastianutti, L. M. and Nunamaker, J. F. (1992), "Electronic Brainstorming and Group Size," Academy of Management Journal, Vol. 35, No. 2, pp. 350-369.
12. Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Kluwer Academic Publishers, Boston, MA.
13. Guilford, J. P. (1967). *The nature of human intelligence*. McGraw-Hill, New York.
14. Howe, J. (2006). The rise of crowdsourcing, *Wired Magazine*, (14), 1-4.
15. Kittur, A., Chi, E. H., and Suh, B. (2008). Crowdsourcing user studies with Mechanical Turk. In *Proc. CHI 2008*, ACM Press (2008), 453-456.
16. Kosorukoff, A. (2001). Human based genetic algorithm. In *Proc. IEEE Conference on Systems, Man, and Cybernetics*, 3464-3469.
17. Little, G., Chilton, L. B., Goldman, M., and Miller, R. C. (2010). Exploring iterative and parallel human computation processes. In *Proc. of the ACM SIGKDD Workshop on Human Computation*, 68-76.
18. Love, B. C. (2005). Environment and goals jointly direct category acquisition. *Current Directions in Psychological Science, 14*, 195–199.
19. Malone, T. W., Laubacher, R., and Dellarocas, C. (2010). Harnessing crowds: Mapping the genome of collective intelligence, CCI Working Paper 2009-001.
20. McKnight, D. H., Choudhury, V. and Kacmar, C. (2002) Developing and Validating Trust Measures for e-Commerce: An Integrative Typology Information Systems Research, 13, 3, 334–359.
21. Malon, T.W., Laubacher, R., and Dellarocas, C.: Harnessing crowds: Mapping the genome of collective intelligence. MIT Sloand School Working Paper 4732-09, (2010)
22. Montoya-Weiss, M.M., Massey, A. P. and Song, M. (2001). Getting it together: temporal coordination and conflicts management in global virtual teams. *Academy of Management Journal*, 44, 6, 1251-1262.

23. Mumford, M. D. (2003). Where have we been, where are we going? Taking stock in creativity research. *Creativity Research Journal*, *15*, 107-120.
24. Nickerson, J. V., and Sakamoto, Y. (2010). Crowdsourcing creativity: Combining ideas in networks. *Workshop on Information in Networks*. New York, NY.
25. Nickerson, J. V., Sakamoto, Y., and Yu, L. (2011). Structures for creativity: The crowdsourcing of design. CHI Workshop on Crowdsourcing and Human Computation
26. Osborn, A. F. (1957). *Applied Imagination*. New York: Scribner.
27. Paul, S., Haseman, W. D. and Ramamurthy, K. (2004) Collective memory and cognitive-conflict group decision-making: An experimental investigation, *Decision Support Systems*, 36, 3, 261–281.
28. Quinn, A. J., and Bederson, B. B. (2011). Human Computation: A Survey and Taxonomy of a Growing Field. *CHI*.
29. Runco, M. A., and Pritzker, S. R. (1999). *Encyclopedia of Creativity*. San Diego: Academic Press.
30. Surowiecki, J. (2004). The wisdom of crowds: Why the many are smarter than the few and how collective wisdom shapes business, economies, societies, and nations. New York: Random House.
31. Takagi, H. (1998), Interactive Evolutionary Computation: System Optimization Based on Human Subjective Evaluation, *IEEE International Conference on Intelligent Engineering Systems*
32. Wagner, C., and Back, A. (2008). Group wisdom support systems: Aggregating the insights of many through information technology, *Issues in Information Systems*, *9*, 343-350.
33. Ward, T. B., Smith, S. M., and Vaid, J. (1997). *Creative Thought: An Investigation of Conceptual Structures and Processes*. Washington, DC: APA Books.
34. Yu, L., and Nickerson, J. V. (2011). Cooks or cobblers? Crowd creativity through combination, *CHI 2011*, ACM Press.