Supporting Knowledge of Design Risks in Collective Intelligence Efforts

SPENCER E. CARLSON, DANIEL G. REES LEWIS, LEESHA V. MALIAKAL, MATTHEW W. EASTERDAY, and ELIZABETH M. GERBER, Northwestern University

1. INTRODUCTION

Collective intelligence promises to tackle major societal challenges, from climate change to poverty, by using computation to combine human intelligence in better ways [Malone 2018; Malone and Bernstein 2015], yet our knowledge of supporting planning in collective intelligence is incomplete.

Existing collective intelligence efforts vary in the complexity [1] and ill-structuredness[2] of the tasks that human contributors complete, and the domain expertise of humans who plan those tasks. In some systems, humans with domain expertise structure and decompose a complex, ill-structured task into smaller tasks that others can complete without special planning expertise [Benkler et al. 2015; Valentine et al. 2017]. In other systems, complex tasks are automatically decomposed into well-defined tasks [Dow et al. 2012; Kittur et al. 2011; Noronha et al. 2011]. But a third, understudied kind of collective intelligence—what we call *novice-based collective innovation*—requires domain novices to plan highly complex, ill-structured tasks [Easterday et al. 2018]. This kind of collective intelligence is critically important for solving societal problems, which are highly complex and ill-structured [Jonassen and Hung 2015], yet many of the people available to plan and work on those problems (such as undergraduate activists) are domain novices. Unfortunately, novice-based collective innovation has proven elusive, with many efforts producing vast, low-quality results [Salehi and Bernstein 2018]. To improve novice-based collective innovation, how might systems must support novices to plan effectively?

1.1 Iteration and Design Risks

Iteration—the design process of seeking new knowledge to update the problem definition and solution idea [Adams and Atman 1999; Adams et al. 2003]—provides a way for collective intelligence systems to refine solutions to complex, ill-structured problems. Expert designers iterate by (a) reflecting on their solution, (b) identifying risks—gaps in their knowledge that could lead their solution to fail—and (c) planning to seek new knowledge to improve their solution [Adams and Atman 1999; Adams et al. 2003; Schön 1983; Guindon 1990; Kolodner and Wills 1996]. For example, expert policy designers might seek knowledge about politicians' positions to mitigate the risk of designing a politically toxic climate policy.

Novices, lacking experience, are unaware of all of the risks that could lead their solution to fail [Adams et al. 2003; Crismond and Adams 2012]. This makes it difficult for novices to plan effectively by seeking out the knowledge needed to test risky assumptions and iterate toward a successful solution. Fortunately, it is possible to help novices iterate effectively by giving them templates that scaffold the iteration planning process, and a checklist to help them identify design risks [Rees Lewis et al. 2018].

The challenge for collective intelligence systems is to surface the right design risks. Because design risks are often domain-specific (e.g., the risk of designing a politically toxic climate policy is not relevant when designing open source software), it is impossible to identify a single set of risks that all collective

[1] Size and difficulty [Jonassen and Hung 2015]

[2] The degree to which necessary goals and subtasks must be discovered before they can be completed [Jonassen and Hung 2015]

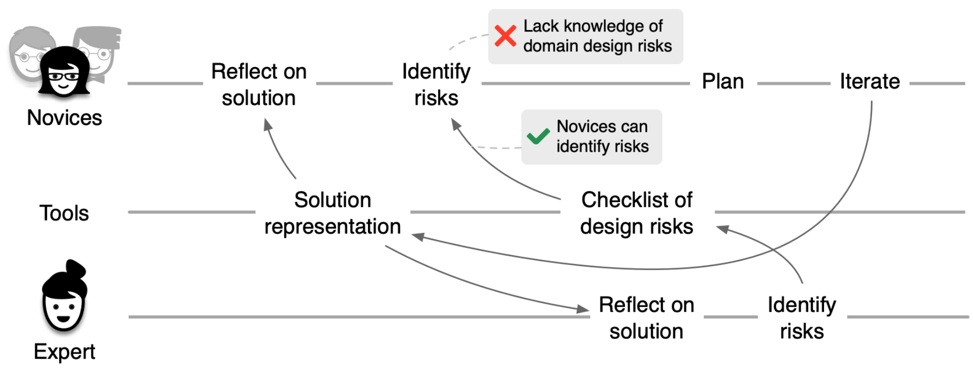

Fig. 1. Collective intelligence systems that rely primarily on contributions from domain novices can address innovation problems by helping novices plan—specifically, by supporting their knowledge of domain-specific design risks, which can be surfaced through structured critique from relatively few domain experts.

intelligence systems must support. Instead, collective intelligence requires a method for surfacing the specific risks that are relevant in different projects so that the novices in those projects can iterate intelligently. We explore how this may be possible using *risk-surfacing systems* that support structured critique between domain experts and domain novices as a way of surfacing the experts' tacit knowledge of design risks (Fig. 1).

2. METHODS

We prototyped a risk-surfacing system in a face-to-face setting in order to understand the fundamental processes that a collective intelligence system must ultimately support, whether online, face-to-face, or both. We tested the prototype process in an extracurricular innovation club where 21 undergraduate novices designed solutions to societal challenges while experts (6 design faculty and graduate students, including the researchers) critiqued the projects to surface tacit knowledge of design risks. The design challenges were: improving airport accessibility for travelers with autism, reducing wheelchair damage from air travel, supporting people with dementia, increasing first responder support for youth mental health, and reducing teen depression. Experts focused on surfacing design risks related to domain expertise that was required for all challenges, such as user-centered design, social impact, and working with project partners.

Each week, the novices used a template to represent their solution idea, identified risks, planned to learn new knowledge, and iterated on their solution representation according to what they learned. Over 6 1-week iterations, one expert observed each novice team's planning discussion and solution representation to identify risks. Experts added these risks to a checklist (Table I) that novices used in the following weeks to plan more effectively. Novices worked on projects for 35+ hrs/week while experts critiqued for only 2hrs/week.

3. FINDINGS AND CONCLUSIONS

We found this approach successfully surfaced 49 design risks that the novices struggled to recognize, but the experts recognized as critical for developing intelligent design solutions (Table I). While the specific risks surfaced in this study have limited generalizability, the *approach* we tested may be useful for surfacing domain-specific design risks across a far wider range of innovation projects.

This study highlights an opportunity for novice-based collective intelligence efforts to tackle increasingly complex, ill-structured tasks—a necessary condition for mobilizing broad sectors of society to

Table I. Experts surfaced their tacit knowledge of 49 design risks by critiquing novices' solution representations.

Project area	Design risks	Rationale for risk to the project
Partner	1. No contact with a real person at a partner organization 2. No identified partner need 3. Need isn't reasonable (it conflicts with data and/or common knowledge) 4. No supporting evidence (including both content and source)	If designers do not understand the partner's needs, there is a risk of designing a solution that the partner does not want.
User access plan	5. No plan 6. Plan won't achieve access to intended user 7. Plan probably won't work (based on common knowledge and any evidence) 8. No supporting evidence (including both content and source)	If designers do not have a plan to access users, there is a risk they will be unable to check whether they understand the users' needs and are making progress toward a solution the users want.
Demoing plan	9. No plan 10. Plan doesn't involve demoing to intended partner 11. Plan doesn't involve demoing every 1-2 weeks 12. Plan probably won't work (based on common knowledge and any evidence) 13. No supporting evidence (including content and source)	If designers do not have a plan for demoing, there is a risk they will be unable to check whether they are making progress toward a solution the partner wants.
Desired social impact	14. No defined desired impact 15. Desired impact is not a social impact 16. No identified challenges preventing desired impact 17. Challenges aren't believable (given data and common knowledge) 18. No objective way to measure impact 19. No baseline measurement or goal 20. No deadline for reaching the goal 21. No supporting evidence (including content and source)	If designers do not understand the desired social impact of their project, there is a risk they will misconstrue the root causes of the social problem and design an ineffective solution and there is a risk they will be unable to judge whether their solution made an impact.
User	22. No identified user 23. User doesn't matter to the partner 24. No identified user need (including a job, pain, and gain) 25. User need is not well-supported by data and common knowledge 26. No supporting evidence (including content and source)	If designers cannot articulate a user need that is supported by evidence, there is a risk they will misconstrue the root cause(s) of that need and design ineffective solutions.
Root causes	27. No defined causal chains that link the obstacles to satisfying the user's need, partner's need, and desired impact each back to a fixable root cause 28. Missing obstacles to satisfying user need, partner need, and/or desired impact 29. Causal chains are not credible relationships between individual variables 30. Causes not reasonable given common knowledge and available data 31. No supporting evidence (including content and source)	If designers have not identified the fixable root causes of a problem, there is a risk that their solutions will be ineffective and rejected by users or the partner.
Value proposition	32. No value proposition (including solution, features, and how it serves user need) 33. Value proposition doesn't address a fixable root cause and user need 34. No argument for how solution's features will overcome root cause to address need 35. Not specific enough to guide building and testing prototypes 36. Not specific enough that testing it would yield decisive results 37. No/weak evidence that the solution is desirable 38. No/weak evidence that the solution is effective 39. Evidence doesn't specify both content and source of data	If designers cannot explain and provide evidence of how their solution will solve the user's problem, there is a risk that it will not.
Existing solutions	40. No identified existing solutions (or very strong argument that none exist) 41. No reasonable argument (based on data and common sense) that they are inferior 42. Existing solutions not actually relevant to the intended user and need 43. No supporting evidence (including content and source)	If the solution is inferior to existing solutions, there is a risk that the user or partner will not adopt it.
Implementation strategy	44. No defined implementation strategy (including defining the resources needed) 45. Implementation strategy will probably implement the solution in a way that does not work for the partner or does not achieve desired impact 46. No credible arguments (based on common knowledge and data) that strategy is feasible 47. No supporting evidence (including content and source)	If designers do not know how they will build and diffuse the solution—or if they lack evidence that their strategy will work—there is a risk of designing something that is never implemented.
Impact	48. No believable argument that desired impact was achieved 49. No supporting evidence (including content and source)	Even if a solution is implemented, there is a risk that it may not make the intended social impact for unforeseen reasons.

solve societal problems through collective intelligence. Specifically, we offer initial evidence that collective intelligence efforts that rely primarily on contributions from domain novices can use structured critique to surface domain experts' tacit knowledge of design risks, which other research suggests can help novices plan iterations in innovation problems [Rees Lewis et al. 2018]. Future research should examine whether and how this kind of structured critique interaction can be supported through collaborative technologies to surface relevant design risks and enable intelligent iteration in novice-based collective innovation.

REFERENCES

Robin S. Adams and Cynthia J. Atman. 1999. Cognitive processes in iterative design behavior. In *Proceedings of the 29th ASEE/IEEE Frontiers in Education Conference*. San Juan, Puerto Rico, 11A6/13–11A6/18. DOI:http://dx.doi.org/10.1109/FIE.1999.839114

Robin S. Adams, Jennifer Turns, and Cynthia J. Atman. 2003. Educating effective engineering designers: the role of reflective practice. *Design Studies* 24, 3 (May 2003), 275–294. DOI:http://dx.doi.org/10.1016/S0142-694X(02)00056-X

Yochai Benkler, Aaron Shaw, and Benjamin Mako Hill. 2015. Peer Production: A Form of Collective Intelligence. In *Handbook of collective intelligence*, Thomas W. Malone and Michael S. Bernstein (Eds.). MIT Press, Cambridge, Mass, 175–204.

David P. Crismond and Robin S. Adams. 2012. The Informed Design Teaching and Learning Matrix. *Journal of Engineering Education* 101, 4 (Oct. 2012), 738–797. DOI:http://dx.doi.org/10.1002/j.2168-9830.2012.tb01127.x

Steven Dow, Anand Kulkarni, Scott Klemmer, and Björn Hartmann. 2012. Shepherding the crowd yields better work. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work (CSCW '12)*. Association for Computing Machinery, Seattle, Washington, USA, 1013–1022. DOI:http://dx.doi.org/10.1145/2145204.2145355

Matthew W. Easterday, Elizabeth M. Gerber, and Daniel G. Rees Lewis. 2018. Social Innovation Networks: A New Approach to Social Design Education and Impact. *Design Issues* 34, 2 (April 2018), 64–76. DOI:http://dx.doi.org/10.1162/DESI_a_00486

Raymonde Guindon. 1990. Designing the Design Process: Exploiting Opportunistic Thoughts. *Human–Computer Interaction* 5, 2-3 (June 1990), 305–344. DOI:http://dx.doi.org/10.1080/07370024.1990.9667157

David H. Jonassen and Woei Hung. 2015. All problems are not equal: Implications for problem-based learning. In *Essential Readings in Problem-Based Learning*, Andrew Walker, Heather Leary, Cindy Hmelo-Silver, and Peggy A. Ertmer (Eds.). Purdue University Press, 7–41.

Aniket Kittur, Boris Smus, Susheel Khamkar, and Robert E. Kraut. 2011. CrowdForge: crowdsourcing complex work. In *Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11)*. Association for Computing Machinery, Santa Barbara, California, USA, 43–52. DOI:http://dx.doi.org/10.1145/2047196.2047202

Janet L. Kolodner and Linda M. Wills. 1996. Powers of observation in creative design. *Design Studies* 17, 4 (Oct. 1996), 385–416. DOI:http://dx.doi.org/10.1016/S0142-694X(96)00021-X

Thomas W. Malone. 2018. *Superminds: the surprising power of people and computers thinking together*. Little, Brown and Co, New York, NY.

Thomas W. Malone and Michael S. Bernstein (Eds.). 2015. *Handbook of collective intelligence*. MIT Press, Cambridge, Massachusetts.

Jon Noronha, Eric Hysen, Haoqi Zhang, and Krzysztof Z. Gajos. 2011. Platemate: crowdsourcing nutritional analysis from food photographs. In *Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11)*. Association for Computing Machinery, Santa Barbara, California, USA, 1–12. DOI:http://dx.doi.org/10.1145/2047196.2047198

Daniel G Rees Lewis, Jamie S Gorson, Leesha V Maliakal, Spencer E Carlson, Elizabeth M Gerber, Christopher K Riesbeck, and Matthew W Easterday. 2018. Planning to Iterate: Supporting Iterative Practices for Real-world Ill-structured Problem-solving. In *Rethinking learning in the digital age: Making the Learning Sciences count*, Vol. 1. International Society of Learning Sciences, London, UK, 9–16.

Niloufar Salehi and Michael S. Bernstein. 2018. Hive: Collective Design Through Network Rotation. *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW (Nov. 2018), 1–26. DOI:http://dx.doi.org/10.1145/3274420

Donald A. Schön. 1983. *The reflective practitioner: how professionals think in action*. Basic Books, New York.

Melissa A. Valentine, Daniela Retelny, Alexandra To, Negar Rahmati, Tulsee Doshi, and Michael S. Bernstein. 2017. Flash Organizations: Crowdsourcing Complex Work by Structuring Crowds As Organizations. In *Proceedings of CHI 2017*. ACM Press, 3523–3537. DOI:http://dx.doi.org/10.1145/3025453.3025811