CROWD COMPUTING AND HUMAN COMPUTATION ALGORITHMS

Robert C. Miller

MIT CSAIL
32 Vassar St
Cambridge, MA 02139 USA
rcm@mit.edu

ABSTRACT
Crowd computing harnesses the power of people out in the web to do tasks that are hard for individual users or computers to do alone. Like cloud computing, crowd computing offers elastic, on-demand human resources that can drive new applications and new ways of thinking about technology. Crowd computing depends on appropriate design choices for end-user, crowd, and the software that they are using to communicate and coordinate.

This talk will describe several of our explorations into aspects of crowd computing system design:

Human computation algorithms (Little et al. 2010) coordinate small contributions from many people to solve a challenging problem (such as recognizing terrible handwriting, shown below). This algorithmic approach to crowd coordination is the framework for the systems that follow.

Soylent (Bernstein et al. 2010) is a word processor plugin that crowdsources text editing tasks, such as proofreading and cutting (shown in the image below). Soylent uses a human computation algorithm called Find-Fix-Verify for quality control, getting diverse edits from a crowd while still producing good-quality results.

VizWiz (Bigham et al. 2010) is a smartphone app that helps blind people see using a crowd’s eyes, showing how crowds can be used in systems that require interactive response times.

Adrenaline (Bernstein et al. 2011) is a camera shutter driven by crowd perception, finding the best still frame from a video in seconds. This system shows that a synchronous crowd can be coordinated to produce a good-quality answer quickly.

Caesar (Tang 2011) is a system for code reviewing by students, teachers, and alumni of a software engineering course, which explores design questions of task size and task routing in a crowd with varying skill levels.
Overall, crowd computing raises new challenges at the intersection of systems and human-computer interaction, including minimizing latency, improving quality of work, and providing the right incentives to the crowd. We are now in a position where "Wizard of Oz" is no longer just a prototyping technique -- thanks to crowd computing, Wizard of Oz systems can be useful and deployable.

ACKNOWLEDGEMENTS

This work was supported in part by Adobe, by the Quanta T-Party project, and by the National Science Foundation (award number SoCS-1111044). Any opinions, findings, conclusions or recommendations expressed herein are those of the authors and do not necessarily reflect the views of the sponsoring organizations.

REFERENCES

Bernstein, M.S., Little, G., Miller, R.C., Hartmann, B., Ackerman, M.S., Karger, D.R., Crowell, D., & Panovich, K. (2010), Soylent: a word processor with a crowd inside, in 'Proc. UIST '10'.


