more opportunities to assess areas of the assistant that have conflicting tests. Conversely, tests with a high level of agreement could have a very low testing priority, thereby redirecting user effort toward the areas it will be most beneficial.

Finally, our focus has been on *anonymous* mini-crowds, but there are situations where members of a mini-crowd may be quite familiar with one another. A promising area for future work involves building upon computer-supported collaborative work (CSCW) research to support small groups of collaborating end users assessing a shared assistant, such as with family homes or small workgroups.

VI. Conclusion

This paper provides the first empirical evaluation of mini-crowdsourcing the assessment of intelligent assistants. As these assistants take on more critical tasks, assessing when to rely on them will become increasingly important. Our results show that using an asynchronous mini-crowd to assess these assistants confers benefits to end users, but not without costs. This paper has empirically investigated the trade-offs to better understand the "price" of these benefits.

Larger mini-crowds, as expected, found more of an assistant's errors, tested more of its logic, and introduced enough redundancy to reduce crowd mistakes, as compared with smaller mini-crowds. However, results we did *not* expect were:

- Bigger was not always better: the mini-crowd of 6 was *worse* about introducing false negatives than the mini-crowd of 11.
- Diminishing returns: even in metrics where larger mini-crowds outperformed smaller crowds, the benefit of increasing the crowd size quickly dropped, while the cost scaled linearly.
- No loafing: contrary to the phenomena of social loafing, participants working with large mini-crowds did not overly rely upon the crowd.
- Tool-supported strategies versus mini-crowds: participants using the WYSIWYT/ML-supported "priority" strategy found as many errors as participants working with larger mini-crowds.

Overall, our results are encouragingly positive about a future in which shared testing is paired with shared debugging, to support small ecosystems of end users to quickly and effectively assess intelligent assistants that support important aspects of their work and lives.

Acknowledgments

We thank our participants, Shubhomoy Das, Travis Moore, Shalini Shamasunder, and Katie Shaw. This work was supported in part by NSF 0803487.

References

[1] Ambati, V., Vogel, S., Carbonell, J. Active learning and crowd-sourcing for machine translation. *Proc. LREC* (2010), 2169-2174.

[2] Beizer, B. *Software Testing Techniques*. International Thomson Computer Press (1990).

[3] Chang, C. and Lin, C. LIBSVM: A library for support vector machines. http://www.csie.ntu.edu.tw/~cjlin/libsvm (2001).

[4] Glass, A., McGuinness, D. Wolverton, M. Toward establishing trust in adaptive agents. *Proc. IUI*, ACM (2008), 227-236.

[5] Grady, C. and Lease, M. Crowdsourcing document relevance assessment with Mechanical Turk. *Proc. NAACL HLT Wkshp. Creating Speech and Language Data with Amazon's Mechanical Turk* (2010), 172-179.

[6] Hart, S. and Staveland, L. Development of a NASA-TLX (Task load index): Results of empirical and theoretical research, Hancock, P. and Meshkati, N. (Eds.), *Human Mental Workload* (1988), 139-183.

[7] Heer, J. and Bostock, M. Crowdsourcing graphical perception: Using Mechanical Turk to assess visualization design, *Proc. CHI*, ACM (2010), 203-212.

[8] Kittur, A. Chi, E., and Su, B. Crowdsourcing user studies with Mechanical Turk, *Proc. CHI*, ACM (2008), 453-456.

[9] Kniesel, G. and Rho, T. Newsgroup data set. http://www.ai.mit.edu/jrennie/20newsgroups (2005).

[10] Kulesza, T., Wong, W., Stumpf, S., Perona, S., White, R., Burnett, M., Oberst, I., and Ko, A. Fixing the program my computer learned: Barriers for end users, challenges for the machine. *Proc. IUI*, ACM (2009), 187-196.

[11] Kulesza, T., Stumpf, S., Burnett, M., Wong, W., Riche, Y., Moore, T., Oberst, I., Shinsel, A., McIntosh, K. Explanatory debugging: Supporting end-user debugging of machine-learned programs. *Proc. VL/HCC*, IEEE (2010), 41-48.

[12] Kulesza, T., Burnett, M., Stumpf, S., Wong, W., Das, S., Groce, A., Shinsel, A., Bice, F., and McIntosh, K. Where are my intelligent assistant's mistakes? A systematic testing approach, *Proc. IS-EUD (LNCS 6654)*, 171-186.

[13] Latané, B., Williams, K., and Harkins, S. Many hands make light the work: The causes and consequences of social loafing. *J. Personality and Social Psychology*, 37, 6 (1979), 822-832.

[14] Lim, B., Dey, A. and Avrahami, D. Why and why not explanations improve the intelligibility of context-aware intelligent systems. *Proc. CHI*, ACM (2009), 2119-2128.

[15] Lim, B. and Dey, A. Toolkit to support intelligibility in context-aware applications. *Proc. Int. Conf. Ubiquitous Computing*, ACM (2010), 13-22.

[16] Riungu, L., Taipale, O., and Smolander, K. Research issues for software testing in the cloud, *Int. Conf. Cloud Computing Technology and Science*, IEEE (2010), 557-564.

[17] Ruthruff, J., Prabhakararao, S., Reichwein, J., Cook, C., Creswick, E., and Burnett, M. Interactive, visual fault localization support for end-user programmers. *J. Visual Languages and Computing*, Volume 16 (2005), 3-40.

[18] Rothermel, G., Burnett, M., Li, L., Dupuis, C., and Sheretov, A. A methodology for testing spreadsheets. *ACM Trans. Software Engineering and Methodology 10*, 1 (2001), 110-147.

[19] Talbot, J., Lee, B., Kapoor, A. and Tan, D.S. EnsembleMatrix: Interactive visualization to support machine learning with multiple classifiers. *Proc. CHI*, ACM (2009), 1283-1292.

[20] Von Ahn, L., Maurer, B., McMillen, C., Abraham, D., and Blum, M. reCAPTCHA: Human-based character recognition via web security measures. *Science*, 321 (2008), 1465-1468.