Human Computation and Crowdsourcing: Works in Progress Abstracts:
An Adjunct to the Proceedings of the Third AAAI Conference on Human Computation and Crowdsourcing

# Using Expertise for Crowd-Sourcing

**David Merritt, Mark S. Ackerman, Mark W. Newman,
Pei-Yao Hung, Jacob Mandel, Erica Ackerman**
University of Michigan, Ann Arbor, Michigan

**Abstract**

In this paper, we examine whether the use of expertise ratings can help crowd-sourcing systems. We show, using simulations, that a crowd-sourcing system based in social navigation works better when users' expertise levels are taken into account.

## Introduction

We wanted to understand how differentiating levels of user expertise might benefit crowd-sourcing systems. Expertise has already been used in crowd-sourcing with considerable benefit. For example, ManyEyes (Viegas et al. 2007) uses a crowd to understand data visualizations with people calling out what they know and understand, and Lasecki et al. (2013) assume a wide range of everyday expertise in the crowd that visually impaired users can tap into. However, there have been, to our knowledge, no systematic examinations of the use of expertise.

In our current work, we use simulations to examine the benefits and issues in differentiating levels of expertise in one scenario: a crowd-sourcing system based around social navigation, or how digital traces of what many people have done before might best be used (Wexelblat and Maes 1999).

To answer this question, we present results from large-scale simulations. These simulations were effective for examining the basic utility of expertise at a scale appropriate for crowd-sourcing systems.

## Escalier

We used, for our examination, a heavily abstracted crowd-sourcing system, one that creates and uses a map of *possible* (i.e., usable) ordered vectors of tuples of (property, value, type). For simplicity, we refer to these vectors as configurations. One might think of these as system configurations, security settings, or activity traces (where time is implicit or could be a fourth parameter in the tuple). The system's goal is to aid in the discovery of the map of *valid* (i.e., stable) configurations. In the abstract, this is an instance of social navigation in social computing, where users use one another's digital traces to guide further action. In our scenario, the crowd-sourced system, which we will call Escalier, allows many users to co-construct a map of valid configurations by guiding their explorations of single points or subspaces in the configuration space.

In our scenario, users report their configurations based on an objective function. This map, constructed from many users' reports, is probabilistic because any given user report could be incorrect, and so we have only a likelihood estimate that the given configuration meets the criterion.

To find valid configurations, we assume a constant stream of user reports to Escalier. A user report consists of submitting the configuration, and expressing whether it meets an objective function. This user report may be explicit or inferred. For example, the user can explicitly report whether she believes her system is secure or not. The user report can be automatic, such as when the system itself reports whether it has crashed. We want to explore how a social computing system like Escalier would be affected if the expertise levels of the users are known.

## Simulations

We chose to examine the use of expertise using simulations.

To do this, we divided all possible configurations C into two subsets: a subset of configurations $C_V$ that meet an objective function and a subset of configurations, $C_{IV,}$ that do not. We will term these configurations as valid or invalid, respectively, for clarity in the discussion. We assume that users or their agents must report whether a configuration meets this objective function, that these reports occur over time, and these reports can have a probability of being erroneous. Not all user reports will be accurate; users, or their user agents, may not accurately assess whether the configuration is usable.

Because of space, we cannot describe the models and runs of the simulations in detail. While we modeled the space of configurations in a number of ways, here, we present simulation data where valid configurations are clustered around canonical configurations, or configurations

known in advance to be valid. In this model, a configuration is a 50-parameter vector, where each parameter can have 10 values. The size of the simulation space C is therefore $10^{50}$ configurations. We ran Monte Carlo simulations, where each Monte Carlo run laid out between 1,020 and 20,300 configurations. The rest of the space consisted of invalid configurations. We needed a relatively limited model of users because we were primarily interested in their search behavior in the aggregate. We allowed simulated users to have expertise, by employing a more broad and diverse search strategy than do novices. Based on White et al. (White, Dumais, and Teevan 2009), we modeled three characteristics of user behavior: First, users have varying levels of expertise, and higher levels of expertise are more rare than lower expertise. Thus we use five expertise levels ranging from novice to expert. A given user's expertise level is determined according to a Pareto distribution where novices are the majority and experts are rare. Second, users' levels of expertise correlate with the accuracy of their mental model of the problem space. Users with higher levels of expertise are more accurate in their reporting than novice users are. Third, we assume users with higher levels of expertise will perform a more thorough search of the configuration space.

Our simulations used 10,000-round Monte Carlos to investigate Escalier's usefulness under differing conditions, using the Mersenne Twister random number generator. We ran a 4x4x5x2 factorial simulation experiment where the factors were, in order, the number of users, the number of canonical configurations, the number of valid configurations initialized in the configuration space, and whether or not Escalier was able to consider user expertise. In total, we ran 160 simulations, each with 10,000 rounds and 200,000 simulated users, where 200,000 simulated users were sufficiently large to assess potential utility at scale.

## Simulation Results

In brief, our simulations showed that (1) Even with a relatively small number of users for a social computing system (10,000 users), there are benefits: Escalier helped users discover about one-third more valid configurations (36%) than were originally known. The gain in known valid configurations appears to grow dramatically (1168%) as more users interact with Escalier. (2) The number of false positives is small, and the true positives that are found increase with the number of users.

### Using Expertise

The ability to use a user's expertise seems to have a significant positive effect on helping users find valid configurations more quickly than if expertise were not taken into account. Remember we assume that user reports may be inaccurate, especially if they have low expertise. Figure 1, using 20,000 valid configurations and 300 canonical configurations, shows the comparison between using expertise and not. Without using the expertise rankings, it takes about 160,000 users to discover 3,000 true positives. When expertise assessment is enabled, it takes roughly 90,000 users, which is a 43% decrease in users needed. This boosting effect is especially valuable when there are relatively few users. In another analysis we also found these results were robust against how expertise metrics were calculated.

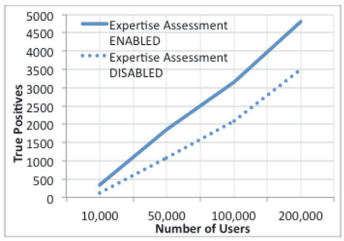


*Figure 1. Effect of expertise on # of valid configurations found. This shows the positive effect of accounting for users' differing expertise levels.*

In summary, the simulations with user expertise being used showed that the use of expertise gives a noticeable boost, even when there are fewer users. Thus it is important to assess users' expertise levels, if possible, for crowd-sourcing systems like the one in our scenario. They also suggest domain experts are between two and three times more effective at finding valid configurations than domain novices and about twice as effective as advanced beginners. Designing similar crowd-sourcing systems to incentivize or solicit input from domain experts is likely to be worthwhile.

All of these results were consistent across all simulations. Sensitivity analyses suggest, even if the model assumptions were changed, performance would not deviate far from the findings presented here.

To conclude, we also found (3) knowing users' expertise levels boosted results, substantially in most cases, and (4) difficult-to-obtain measures of expertise are likely not to be necessary.

## References

Lasecki, W. S., Thiha, P., Zhong, Y., Brady, E., & Bigham, J. P. 2013. Answering visual questions with conversational crowd assistants. *Proc. ASSETS '13, 18*

Viegas, F. B., Wattenberg, M., Van Ham, F., Kriss, J., & McKeon, M. 2007. Manyeyes: a site for visualization at internet scale. *IEEE Visualization and Computer Graphics, 13*(6), 1121-1128.

Wexelblat, A. and Maes, P. 1999. Footprints: History-Rich Tools for Information Foraging. *Proc. CHI '99*, 270-277.

White, R. W., Dumais, S. T. and Teevan, J. 2009. Characterizing the influence of domain expertise on web search behavior. *Proc. WSDM '09.* 132-141.