Toward Automating Crowd RE

Pradeep K. Murukannaiah*, Nirav Ajmeri†, and Munindar P. Singh†
* Department of Software Engineering, Rochester Institute of Technology, Rochester, NY 14623-5608, USA
† Department of Computer Science, North Carolina State University, Raleigh, NC 27695-8206, USA
pkmvse@rit.edu, najmeri@ncsu.edu, m.singh@ieee.org

***Abstract*—Crowd RE is an emerging avenue for engaging the general public or the so called *crowd* in variety of requirements engineering tasks. Crowd RE scales RE by involving, potentially, millions of users. Although humans are at the center of Crowd RE, automated techniques are necessary (1) to derive useful insights from large amounts of raw data the crowd can produce; and (2) to drive the Crowd RE process, itself, by facilitating novel workflows combining crowd and machine intelligence.**

To facilitate automated techniques for Crowd RE, first, we showcase a crowd-acquired dataset, consisting of requirements and their ratings on multiple dimensions for the smart homes application domain. Our dataset is unique in that it contains not only requirements, but also the characteristics of the crowd workers who produced those requirements including their demographics, personality traits, and creative potential. Understanding the crowd characteristics is essential to developing effective Crowd RE processes. Second, we outline key challenges involved in automating Crowd RE and describe, how our dataset can serve as a foundation for developing such automated techniques.

***Keywords*-Crowd RE; personality; creativity; smarthome; dataset; data challenges**

I. Introduction

User participation is essential in a variety of requirements engineering (RE) tasks, including requirements identification, prioritization, conflict analysis, and negotiation. Crowd RE seeks to facilitate large scale user participation in RE, e.g., to discover creative requirements [1], extract requirements from textual sources on the Web [2], and receive feedback on applications and thereby insights on how to evolve requirements [3]. Crowd RE makes RE efficient and yields requirements representing the needs of a wide variety of stakeholders.

Humans are the centerpiece of Crowd RE. However, human effort, even if it is that of the crowd, is expensive. We envision Crowd RE to be a process that combines the works of humans and automated techniques, and helps reinforce each other. That is, humans perform tasks that automated techniques are not yet capable of, e.g., tasks requiring creativity [4], and automated techniques perform tasks that are nontrivial or time consuming for humans, e.g., clustering thousands of requirements.

Automated techniques can facilitate Crowd RE in two ways. First, large scale user participation implies that Crowd RE produces large amounts of data. The crowd-produced data is likely to be unstructured, typically in natural language, and noisy. The raw data, in itself, would be of little value. However, manually deriving useful insights from the raw data can be nontrivial. Consider, for example, application reviews [3] as the crowd-produced data, which includes insights valuable to RE such as users' sentiment about existing features and new feature requests hidden within some reviews. However, deriving such insights, manually, is both time consuming and error prone. Thus, we need automated techniques to process the crowd-produced data and derive insights valuable to RE.

Second, human effort is valuable and we want to expend it only when required. Automated techniques can help reduce the human effort required in a Crowd RE task. Consider, for example, the idea generation task [1], where crowd users come up with requirements for an application. Since crowd users often work independently, it is likely that many users produce similar ideas in the idea generation task. Imagine that an automated technique clusters ideas as they come in, shows the clusters to the users, and asks them to produce ideas distinct from those in the current clusters. Doing so can reduce the crowd effort required to generate a rich variety of ideas.

Training and testing automated techniques require data. Thus, a key challenge in developing automated techniques for Crowd RE is finding suitable datasets. The specific type of data required depends on the problem. On the one hand, a variety of crowd-produced data is available on the Web, e.g., application reviews [3], tweets about software applications [5], and product discussions on social forums [6]. However, curating useful datasets from Web sources is time consuming since data collection APIs may not always be available and when available, they are typically rate limited. On the other hand, one may explicitly solicit human-intensive work on crowdsourcing platforms, e.g., [7], [8]. Whereas the latter option provides more flexibility on the specific type of data to collect, it requires incentivizing crowd workers, typically via monetary benefits. As it is nontrivial to build these datasets, making them public can be quite valuable to facilitating automated techniques for Crowd RE.

Our contribution in this paper is two fold.

Dataset: We showcase the *smarthome requirements* dataset [9] curated by soliciting requirements from the crowd for smarthome applications. We involved 609 Amazon Mechanical Turk (MTurk) users to curate this dataset.

Data challenges: We identify key opportunities to develop automated Crowd RE techniques based on our dataset. These opportunities include both techniques to process the user stories produced by crowd workers and techniques to make the process of acquiring the data more efficient.

Section II summarizes our dataset, including distributions of key variables, and Section III outlines the data challenges.

II. Dataset

We collected *smarthome requirements* dataset [9] in two phases. In the first phase, we asked crowd workers to describe their requirements of a smart home. In the second phase, we asked other crowd workers to rate the requirements produced by workers in the first phase on the dimensions of clarity, novelty, and usefulness of the requirements. For each crowd worker, we collected information about their demographics, personality traits, and creative potentials. A subset of the dataset is used to understand the influences of crowd workers' personality traits and creative potential on the novelty and usefulness of the requirements the workers produce [1].

Figure 1 shows a model of the dataset. It includes requirements, their ratings, and the characteristics of the crowd workers collected via a presurvey on demographics, personality survey, creativity survey, and a postsurvey.

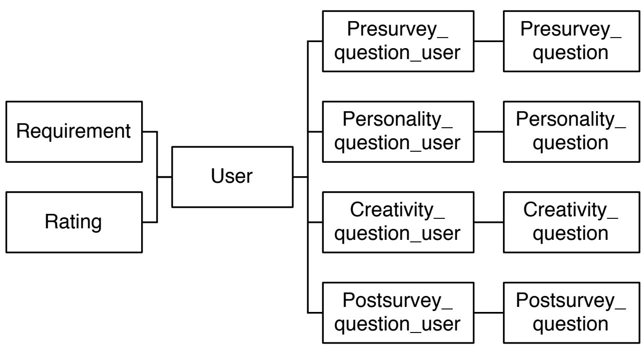

Fig. 1. Model showing requirements, ratings, and user characteristics.

A. Requirements

In the first phase, we acquired requirements via a sequential work structure [10], where we cognitively stimulated the workers to produce creative requirements by exposing them to requirements produced by other workers. The work structure consisted of three stages and involved 300 crowd workers.

In the first stage, we showed each of 50 workers two example smarthome requirements from us and asked the worker to produce at least 10 distinct smart home requirements. In the second stage, we showed each of 128 workers 10 requirements selected from the first stage and asked the worker to produce at least 10 new requirements. Similarly, in the third stage, we showed each of 122 workers 10 requirements selected from from the second stage and asked to produce at least 10 new requirements. In each stage, we encouraged workers to produce more creative requirements than those shown to them.

The workers produced requirements in the user story format as shown in Figure 2. Each worker chose one of *Entertainment*, *Energy*, *Health*, *Safety*, and *Other* as the application domain for each requirement the worker produced. The worker also added a comma separated list of tags describing the requirement. Table I summarizes the number of requirements we acquired for each application domain.

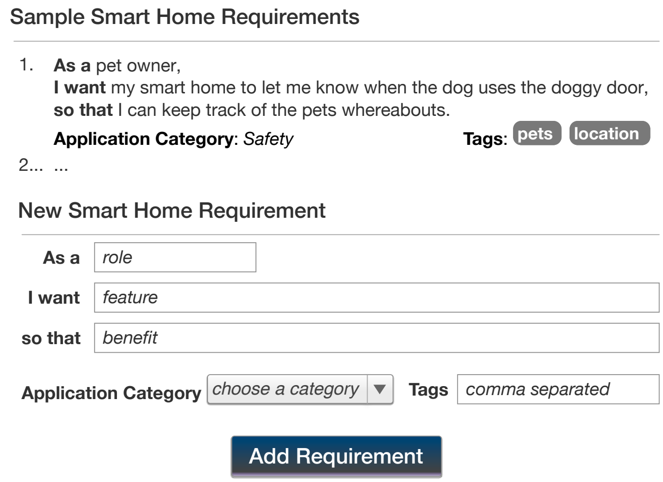

Fig. 2. A screen mockup [1] showing the user story format in which crowd workers produced smarthome requirements.

TABLE I
Number of requirements for each application domain, and the top five tags (unprocessed) within each domain.

Domain	Req. Count	Tags (with Counts)
Energy	626	energy (116), lights (41), electricity (28), water (25)
Entertainment	471	tv (65), music (57), entertainment (41), movies (16), cooking (10)
Health	593	health (67), food (58), pets (22), sleep (20), clean (17)
Safety	892	safety (184), security (61), alarm (39), children (34), doors (29)
Other	384	food (25), cooking (15), kitchen (15), water (11), cleaning (11)

B. Ratings

In the second phase, we acquired ratings for the requirements acquired in the first phase. Each of the 309 workers in the second phase, distinct from the workers in the first phase, rated up to 30 requirements, on a Likert scale of 1 (very low) to 5 (very high), for *clarity*, *usefulness*, and *novelty*. These rating metrics were described to the workers as follows:

Clarity A clear requirement is unambiguous and provides an appropriate level of detail.

Usefulness A useful requirement leads to products that provide value or utility to their users.

Novelty A novel requirement is something that a user finds original and unexpected, i.e., something that is not commonplace, mundane, or conventional.

We collected a total of 8115 ratings for 2966 requirements. Not all requirements received an equal number of ratings because of the randomness we added to the rating process. Overall, 91% requirements were rated at least by two workers. Specifically, as Figure 3 shows, 13.4% of requirements were rated by four or more workers, 55.8% requirements by three, 21.8% requirements by two workers, and 6.2% by one worker. About 3% of requirements remain unrated.

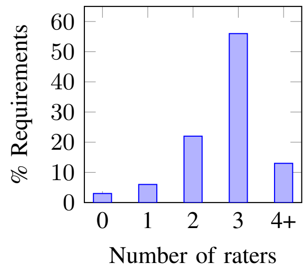

Fig. 3. Distribution of number of ratings acquired for requirements.

Figure 4 shows the distributions of clarity, usefulness, and novelty ratings.

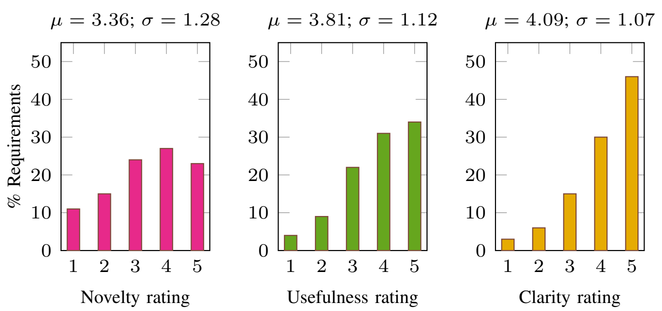

Fig. 4. Distributions of novelty, usefulness, and clarity ratings measured on a Likert scale of 1 (very low) to 5 (very high).

C. Worker Characteristics

Before we acquired requirements and ratings from the workers, we asked the workers to complete presurveys about demographics, personality, and creativity; the workers also completed a post survey about their experience about the task.

Presurvey on demographics: The presurvey includes questions on gender, age, education, work experience, familiarity with computer science, and familiarity with smart homes and devices. Table II summarizes the workers' demographics.

TABLE II
DEMOGRAPHICS OF OUR STUDY WORKERS.

Gender	Male: 52.9%, Female: 46.4%, Other: 0.7%
Age	18–24: 14.1%, 25–34: 52.5%, 35–44: 23.2%, 45–54: 6.2%, 55 or older: 4%
Education	Graduate degree: 14.6%, Bachelor's degree: 42.5%, Some college but no degree: 29.6%, High school: 13%, Less than high school: 0.3%
Work experience with a technology company	5+ years: 5.9%, 1–5 years: 14.1%, <1 year: 7.5%, No experience: 72.5%
Familiarity with computer science, IT and computer networks	Very low: 4.8%, Low: 16.4%, Medium: 45.1%, High: 24.8%, Very high: 8.9%
Familiarity with smart homes	Very low: 7.9%, Low: 23.5%, Medium: 44.2%, High: 20.2%, Very high: 4.2%
Use smart homes device at home	Yes: 33.3%, No: 53.5%, Not sure: 13.2%

Personality and creativity surveys: After collecting demographics, we measured workers' personality traits and creative potential. We employed the Mini-IPIP (International Personality Item Pool) [11] scale to measure a worker's Big Five personality traits of Extraversion (E), Agreeableness (A), Conscientiousness (C), Neuroticism (N), and Openness to experience (O). The Mini-IPIP scales consist of 20 items (11 negative items)—four items for each Big Five trait. Each worker answers 20 items on a Likert scale of 1 (strongly disagree) to 5 (strongly agree). Score of each Big Five trait is computed as the mean of the positive and reverse-scored negative items corresponding to the trait.

To assess workers' creative potential, we employed the Creative Personality Scale (CPS) [12]. The CPS is a 30-item adjective list, consisting of 18 positively scored (e.g., capable, unconventional, and snobbish) and 12 negatively scored (e.g., conservative, honest, and narrow interests) items. Crowd workers answered whether each of those items described them on a Likert scale of 1 (strongly disagree) to 5 (strongly agree). A CPS of a worker is computed as the mean of the positive and reverse-scored negative items.

Figure 5 shows the distributions of personality and creative potential of crowd workers in our dataset.

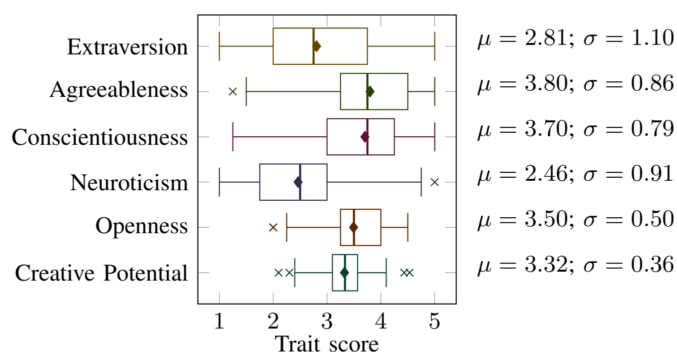

Fig. 5. Distributions of personality traits and creative potential.

Postsurvey: The postsurvey measures the workers' experience of completing the task, for example, how workers perceived the task difficulty (Figure 6 shows the distribution).

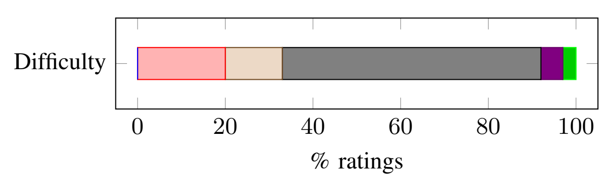

Fig. 6. Distribution of task difficulty on a five-point Likert scale, ranging from easy (on the left) to difficult (on the right), as perceived by the workers.

III. DATA CHALLENGES

Although the *smarthome requirements* dataset was originally collected for understanding human factors in Crowd RE, it is a rich data source. Below, we identify key data challenges (DCs) our dataset may help tackle.

A. Predicting Novelty and Usefulness

In the second phase of data collection, we acquired creativity (novelty and usefulness) ratings for the requirements produced by crowd workers. Automating this process can save significant effort and resources during the Crowd RE process.

DC1. How can we predict the novelty and usefulness of a crowd-acquired requirement?

We conjecture that novel requirements are likely to be divergent from other requirements whereas useful requirements are likely to be those proposed by many users. Thus, understanding semantic similarity between requirements may be key to predicting novelty and usefulness.

Tackling this nontrivial challenge would be valuable not just to Crowd RE, but to creativity research, in general.

B. Summarizing Requirements

In the first phase, crowd workers produced 2966 requirements. Inspecting all requirements to identify unique requirements can be tedious.

DC2. How can we summarize crowd-acquired requirements?

Since crowd workers who produced these requirements work independently, many requirements are likely to be similar. Thus, clustering requirements, e.g., as in [13], can be a simple, but an effective means of summarizing requirements. Further, one may also consider extracting goal models, e.g., as in [14], for summarizing requirements.

C. Prioritizing Requirements

Given multiple stakeholders and a large set of requirements, selecting a subset of those requirements is important when deciding what requirements to implement in a product [15].

DC3. How can we prioritize among the crowd-acquired requirements?

From our dataset, one can build models to prioritize requirements based on factors such as novelty, usefulness, and popularity (needs to be inferred), or a combinations of these.

D. Recognizing Conflicts and Context

The set of crowd-acquired requirements may not be internally consistent—it may contain conflicting requirements [16].

DC4. Given a candidate requirement from the set of all crowd-acquired requirements, how can we identify requirements that conflict with the candidate requirement?

Recognizing events and entities in a requirement, and relationships between entities across requirements may help infer conflicts. However, conflict recognition can be nontrivial and may require understanding the *context* [17] in which a requirement is made, which in itself can be challenging.

E. Identifying Expert Workers

As a crowd worker performs multiple task of a certain type, the worker gains expertise on that task.

DC5. How can we identify a set of expert workers suitable for a given Crowd RE task?

To tackle this challenge, one may consider building a network of crowd workers, e.g., as in [18], connecting workers with similar characteristics based on the desired outcome. For example, if novel requirements are desired, we can employ worker characteristics that influence novelty (e.g., personality) to construct the network. Then, we can utilize community detection to identify suitable workers.

References

[1] P. K. Murukannaiah, N. Ajmeri, and M. P. Singh, "Acquiring creative requirements from the crowd: Understanding the influences of personality and creative potential in Crowd RE," in *Proceedings of the 24th IEEE International Requirements Engineering Conference*, Beijing, 2016, pp. 176–185.

[2] T. D. Breaux and F. Schaub, "Scaling requirements extraction to the crowd: Experiments with privacy policies," in *Proceedings of the 22nd IEEE International Requirements Engineering Conference*, Karlskrona, Sweden, 2014, pp. 163–172.

[3] M. Gómez, B. Adams, W. Maalej, M. Monperrus, and R. Rouvoy, "App Store 2.0: From crowdsourced information to actionable feedback in mobile ecosystems," *IEEE Software*, vol. 34, no. 2, pp. 81–89, 2017.

[4] N. Maiden, S. Jones, K. Karlsen, R. Neill, K. Zachos, and A. Milne, "Requirements engineering as creative problem solving: A research agenda for idea finding," in *Proceedings of the 18th IEEE International Requirements Engineering Conference*, Sydney, 2010, pp. 57–66.

[5] E. Guzman, R. Alkadhi, and N. Seyff, "A needle in a haystack: What do Twitter users say about software?" in *Proceedings of the 2016 IEEE 24th International Requirements Engineering Conference*, Beijing, 2016, pp. 96–105.

[6] G. M. Kanchev and A. K. Chopra, "Social media through the requirements lens: A case study of Google maps," in *Proceedings of the 1st IEEE International Workshop on Crowd-Based Requirements Engineering*, Ottawa, 2015, pp. 7–12.

[7] "Amazon Mechanical Turk," https://www.mturk.com, Apr. 2017.

[8] "Upwork," https://www.upwork.com/, Apr. 2017.

[9] P. K. Murukannaiah, N. Ajmeri, and M. P. Singh, "The smarthome crowd requirements dataset," https://crowdre.github.io/murukannaiah-smarthome-requirements-dataset/, Apr. 2017.

[10] P. André, R. E. Kraut, and A. Kittur, "Effects of simultaneous and sequential work structures on distributed collaborative interdependent tasks," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Toronto, 2014, pp. 139–148.

[11] M. B. Donnellan, F. L. Oswald, B. M. Baird, and R. E. Lucas, "The Mini-IPIP scales: Tiny-yet-effective measures of the Big Five factors of personality," *Psychological Assessment*, vol. 18, no. 2, pp. 192–203, Jun. 2006.

[12] H. G. Gough, "A creative personality scale for the adjective check list," *Journal of Personality and Social Psychology*, vol. 37, no. 8, pp. 1398–1405, 1979.

[13] K. Chen, W. Zhang, H. Zhao, and H. Mei, "An approach to constructing feature models based on requirements clustering," in *Proceedings of the 13th IEEE International Conference on Requirements Engineering*, Paris, 2005, pp. 31–40.

[14] M. Robeer, G. Lucassen, J. M. E. M. van der Werf, F. Dalpiaz, and S. Brinkkemper, "Automated extraction of conceptual models from user stories via NLP," in *Proceedings of the 24th IEEE International Requirements Engineering Conference*, Beijing, 2016, pp. 196–205.

[15] P. Laurent, J. Cleland-Huang, and C. Duan, "Towards automated requirements triage," in *Proceedings of the 15th IEEE International Requirements Engineering Conference*, New Delhi, 2007, pp. 131–140.

[16] P. K. Murukannaiah, A. K. Kalia, P. R. Telang, and M. P. Singh, "Resolving goal conflicts via argumentation-based analysis of competing hypotheses," in *Proceedings of the 23rd IEEE International Requirements Engineering Conference*, Ottawa, 2015, pp. 156–165.

[17] P. K. Murukannaiah and M. P. Singh, "Xipho: Extending Tropos to engineer context-aware personal agents," in *Proceedings of the 13th International Conference on Autonomous Agents and Multi-Agent Systems*, Paris, 2014, pp. 309–316.

[18] S. L. Lim and A. Finkelstein, "StakeRare: Using social networks and collaborative filtering for large-scale requirements elicitation," *IEEE Transactions on Software Engineering*, vol. 38, no. 3, pp. 707–735, 2012.