A systematic mapping study on crowdsourced requirements engineering using user feedback

Chong Wang\(^1\,2\) | Maya Daneva\(^2\) | Marten van Sinderen\(^2\) | Peng Liang\(^1\)

\(^1\) School of Computer Science, Wuhan University, Wuhan, China
\(^2\) University of Twente, Enschede, The Netherlands

Correspondence
Chong Wang, School of Computer Science, Wuhan University, 430072 Wuhan, China.
Email: cwang@whu.edu.cn

Funding information
National Key Research and Development Program of China, Grant/Award Number: 2018YFB1003800; National Natural Science Foundation of China, Grant/Award Numbers: 61702378 and 61672387

Abstract
Crowdsourcing is an appealing concept for achieving good enough requirements and just-in-time requirements engineering (RE). A promising form of crowdsourcing in RE is the use of feedback on software systems, generated through a large network of anonymous users of these systems over a period of time. Prior research indicated implicit and explicit user feedback as key to RE-practitioners to discover new and changed requirements and decide on software features to add, enhance, or abandon. However, a structured account on the types and characteristics of user feedback useful for RE purposes is still lacking. This research fills the gap by providing a mapping study of literature on crowdsourced user feedback employed for RE purposes. On the basis of the analysis of 44 selected papers, we found nine pieces of metadata that characterized crowdsourced user feedback and that were employed in seven specific RE activities. We also found that the published research has a strong focus on crowd-generated comments (explicit feedback) to be used for RE purposes, rather than employing application logs or usage-generated data (implicit feedback). Our findings suggest a need to broaden the scope of research effort in order to leverage the benefits of both explicit and implicit feedback in RE.

KEYWORDS
crowdsourced feedback, evidence-based software engineering, large-scale user involvement, requirements engineering, systematic mapping study, user feedback

1 INTRODUCTION
Crowd-based requirements engineering (RE) is the practice of large-scale user involvement in RE activities. Users are unknown volunteers and massive in number. Their involvement can take a variety of forms. For example, users generate information that becomes freely available for requirements specialists to use for requirements elicitation purposes, or participate in distributed problem-solving where they find workarounds in an application, which in turn may shape the requirements for a subsequent application release. To IT-consulting companies and software development organizations, this opportunity for large-scale user involvement means a way to get good-enough requirements and to achieve just-in-time RE, which implies a significant potential to reduce the cost of RE processes. One promising form of crowdsourcing in RE—that is exploited by businesses and that attracts much research attention—is the use of feedback volunteered by large networks of anonymous users of software systems over a period of time in an RE activity, such as elicitation, validation, or prioritization.\(^1\) So far, prior research suggests that implicit and
explicit user feedback is key to RE practitioners to discover new and changed requirements and to decide what features to add, enhance, or abandon. However, a structured account on the sources and types of user feedback useful for RE purposes and on the characteristics of those feedback types is still lacking. This paper addresses the gap by providing a systematic mapping study to report the state-of-the-art in using crowdsourced user feedback in RE. Implementing the guidelines proposed by Kitchenham et al, we analysed empirical evidence published in literature concerning crowdsourced user feedback in order to consolidate the understanding of this topic and map out areas, which would benefit from future research.

This research makes two contributions. First, it contributes to the emerging literature on the role of crowdsourced feedback in RE. Specifically, it consolidates the empirical research published on the sources of crowdsourced user feedback employed in RE and the ways in which this use was beneficial for RE activities. Second, it indicates RE subareas in which much research work was focused and those where research is scant. To researchers, we indicate directions to which our efforts should expand. To practitioners, we present types of techniques for which much evidence indicates that these are safe to use and seem as good candidates for inclusion in the practitioners' toolbox. To educators, we indicate example concepts that they might possibly include in their RE courses in order to make students aware of those RE activities that leverage the available crowdsourced user feedback.

The rest of this paper is structured as follows: Section 2 provides related work. Section 3 presents the scope of our review, the research questions, and the research process employed. Sections 4 and 5 are on the results of this review and our discussion of the results, respectively. Section 6 is on implications. Section 7 is about validity threats. Section 8 draws the conclusions.

2 | RELATED WORK

In the areas of software engineering (SE) and information systems research, there are 11 literature reviews, that form the related work for our study. For more information on those reviews, we refer interested readers to Table 9 in Appendix B. As part of preparing this paper, we compared them regarding their authors' research goals and scope. The 11 reviews took a variety of perspectives, eg, organizational perspective, innovation process perspective, technical design perspective, and software development process perspective. Below, we summarize these related works.

The review of Asghar et al compared the methods and tools deployed for analysing user feedback and extracting app features and sentiments. Next, the review of Guo et al identified and evaluated a number of data mining techniques used in crowdsourcing environments. However, both studies are not concerned with the actual use of the feedback in the RE activities.

Furthermore, the SLRs of Hosseini et al and Ghezzi et al focused on examining the crowdsourcing phenomenon across multiple disciplines and the various ways in which the notion of crowdsourcing was conceptualized in these disciplines. In addition, Zuchowski et al explored the phenomenon of internal crowdsourcing in organizations (this is when an organization's employees represent the crowd providing feedback that is used to improve the organizational systems).

Among the 11 reviews in Table 9 in Appendix B, two explicitly looked at crowdsourcing in software development. Leicht et al investigated how crowdsourcing in software development was framed as a phenomenon in published literature, and on the basis of their analysis, they developed a first theoretical understanding of crowdsourcing. These authors concluded that most research dealt with the development of sociotechnical systems that enable and support crowdsourcing in terms of collection of feedback from the crowd. Next, Mao et al examined the challenges for crowdsourced SE and mapped out which of those challenges were addressed by existing work and which needed more future research efforts. While the review of Mao et al does include a few literature sources devoted to RE, the central focus of these authors is the broad spectrum of SE activities. In turn, their treatment of RE is from a holistic SE perspective and not from the perspective of RE activities (which are in fact a different level of granularity of literature analysis).

We found only two SLRs that explicitly investigated empirical evidence on using crowdsourced user feedback for the purpose of requirements evolution. Both examined techniques that are applicable on the user review repositories for apps: Rizk et al examined automated tools for sentiment analysis by means of natural language processing, while Genc-Nayebi and Abzan focused on opinion mining techniques for the purpose of requirements evolution. Both studies focus on tools and technical solutions without discussion on (a) the nature of the crowdsourced user feedback, eg, implicit/explicit and (b) the specific ways in which feedback is used in RE activities. The present mapping study addresses these two aspects.

3 | OUR MAPPING STUDY PROCESS

3.1 | Definition of key concepts and formulation of the research questions

To clarify the scope of this mapping study, we provide definition and explanation of two key concepts before formulating the research questions. These two concepts are "crowdsourced user feedback" and "RE activity."
3.1.1 | Crowdsourced user feedback

In crowdsourced RE, the involvement of end users can take a variety of forms. For example, users generate information that becomes freely available for requirements specialists to use for requirements elicitation purposes or participate in distributed problem solving where they find workarounds in an application, which in turn may shape the requirements for a subsequent application release. In the empirical RE literature, quite a few empirical studies report the use of repositories of volunteer user-generated feedback for discovering new and changed requirements and deciding on those features that have to be added, enhanced, or abandoned.¹ For the purpose of this research, we call crowdsourced user feedback the result of various RE tasks that the end users of a software system can perform voluntarily and communicate about to other users or to the software development organization.²¹ Meanwhile, crowdsourced user feedback is a type of critical knowledge in crowdsourced RE. From a knowledge management perspective, a distinction is often made between two types of knowledge: implicit and explicit knowledge.²² Furthermore, in²³, requirements knowledge is defined to “consist of explicit or implicit information that is created or needed while engineering, implementing, or using requirements, and that is useful for answering requirements-related questions in any phase of a software project.” Accordingly, this crowdsourced feedback can be explicit or implicit. Since literature seldom gave definitions of explicit and implicit user feedback, we draw on²⁴ to define explicit and implicit crowdsourced user feedback as below:

Explicit user feedback
If the crowdsourced user feedback that is provided by the crowd after interacting with the software is in visual and readable expressions (eg, text and emoticons), we call it “explicit.” A typical example of explicit feedback are the comments of users of apps in the Apple App Store, which are in a text format and in natural language.

Implicit user feedback
If the crowdsourced feedback is in the form of nonverbal format and is obtainable through monitoring application usage and context, then we call it “implicit.” Examples of implicit feedback are the streams of data generated by an Internet-of-things system that indicates, eg, the intensity of usage or the quality of services provided by this system.

For these definitions, we preferred the work of Jawaheer et al²⁴ as a reference over other classifications (eg, Claypool et al²⁵) because of its popularity in the crowdsourcing literature, its recency, and its suitability to our research context.

This systematic mapping study intends to investigate how crowdsourced user feedback—be it explicit or implicit—is used for various RE purposes, according to published RE literature.

3.1.2 | RE activities

Because we are exclusively concerned with the use of crowdsourced feedback in RE activities, we define the meaning of “RE activity” as well. We draw from the conceptualization of RE as per the Guide to Software Engineering Body of Knowledge (SWEBOK).²⁶ Therein, the software requirements knowledge area is concerned with the elicitation, analysis, specification, and validation of software requirements as well as the management of requirements during the whole life cycle of a software product. SWEBOK provides the definitions of these RE activities as follows:

- **Requirements elicitation (RElic)** is concerned with the origins of software requirements and how the software engineers can collect them.²⁶ This activity aims to identify sources of information about the system and discover the requirements from these sources.²⁷

- **Requirements analysis (RA)** is concerned with the process of analysing requirements to (a) detect and resolve conflicts between requirements; (b) discover the bounds of the software and how it must interact with its organizational and operational environment; and (c) elaborate system requirements to derive software requirements.²⁶ This activity helps developers and concerned stakeholders to not only understand the requirements, their overlaps, and their conflicts, but also reconcile conflicting views and generate a consistent set of requirements.²⁷

- **Requirements specification (RSp)** is establishing the basis for agreement between customers and contractors or suppliers on what the software product is to do as well as what it is not expected to do.²⁶ This activity writes down the requirements in a way that stakeholders and developers can understand²⁷ and provides standardized expression of software requirements.

- **Requirements validation (RV)** is concerned with the process of validating requirements to ensure that the software engineer has understood the requirements and to verify that a requirements document conforms to company standards and that it is understandable, consistent, and complete.²⁶ This activity checks if the requirements are what the stakeholders really need.²⁷

- **Requirements management (RMgt)** is controlling the requirements changes that will inevitably arise.²⁷ This activity supports planning of software requirements, involving communication between the project team members and stakeholders, and adjustment to requirements changes throughout the course of the project.²⁸
These five RE activities are deemed essential to all types of RE processes, regardless the process model an organization chooses to follow. In this mapping study, we adopt the understanding of the RE activities as in SWEBOK. The scope of our mapping study includes research on the use of crowdsourced user feedback in these RE activities. Research on crowdsourcing platforms and their design are outside the scope of this review.

### 3.1.3 | Research questions

As already indicated, we want to explore the state-of-the-art of existing research on the use of crowdsourced user feedback for RE purposes. To this end, we set out to answer four research questions (RQs):

- **RQ1** What sources of implicit and explicit crowdsourced user feedback have been reported in RE activities according to published literature?
- **RQ2** What metadata of crowdsourced user feedback are reported in published literature as being useful for RE?
- **RQ3** In which RE activities has the crowdsourced user feedback been applied?
- **RQ4** What are the demographics of the research on applying crowdsourced user feedback for crowd-based RE according to published literature?
  - **RQ4.1** In which venues has the research on the use of crowdsourced user feedback been published?
  - **RQ4.2** What are the affiliations that contribute to the body of knowledge in the area?

The answer to **RQ1** is needed to investigate the source of explicit and implicit feedback that is used for RE purposes. Until now, crowdsourced user feedback used for RE purposes is reported in regard to different types of software systems and comes from various sources. We want to know what sources of user feedback the researchers and practitioners are working on and what types of software prompted the generation of crowdsourced user feedback for RE purposes. Next, **RQ2** is motivated by the need to understand those parts of user feedback that matter to requirements specialists for the purpose of RE activities. Crowdsourced user feedback usually provides diverse pieces of information, e.g., comments as text and timestamps. In this paper, the information describing aspects of user feedback is defined as the “metadata” of crowdsourced user feedback. Because of the diversity of user feedback, we want to know what metadata of user feedback have been collected and utilized for various RE purposes. Furthermore, **RQ3** is expected to shed light into those RE activities that in fact employed the crowdsourced user feedback for achieving a specific goal in RE. Answering this RQ would help us understand how useful the crowdsourced feedback was to organizations in regard to the five essential RE activities. Finally, **RQ4** is indicative for the generalizability of the published findings. For example, if it would turn out that the majority of publications come from particular contexts (e.g., geographic regions and types of software systems), then our knowledge on the use of crowdsourced feedback in RE would be limited to those contexts.

### 3.2 | Study search

We employed an automatic search method to search studies in two selected digital libraries, i.e., Scopus and Web of Science (WoS). We chose these two electronic databases because recent bibliographic research indicated Scopus and WoS as the most comprehensive and user-friendly databases. These helped us get a diverse set of publications on the subject of crowdsourced user feedback.

Search strategy is crucial for a mapping study since it affects the quality and completeness of retrieved studies as well as the time cost we need to spend on the selection of primary studies. According to the study topic, the following search query was created by joining keywords with possible synonyms. Plus, we scoped the time period of the related publications from January 2006 to December 2017, since the concept “crowdsourcing” was coined in 2006.

\[
\text{(ALL (requirements) AND TITLE (user OR app OR software) AND TITLE (review OR comment OR feedback)) AND (TITLE (requirements) AND TITLE (crowd OR crowdsourced OR crowdsourcing OR data-driven))}
\]

### 3.3 | Study selection

To make the study selection results as objective as possible, we defined selection criteria that were employed in the study selection process. The inclusion (IC) and exclusion criteria (EC) listed in Table 1. were used in the three rounds of study selections to decide whether a study should be included or not.

As shown in Figure 1, the titles of those 876 publications that were returned from the two digital libraries for this study search (see Section 3.2), were manually reviewed by the first author. This step resulted in excluding 185 publication titles because these were duplicates. This meant that the first round selection started with 691 papers (see Figure 1). Out of these, 502 papers were excluded. At this point, the first and the second
authors started the second round selection by abstract. They reviewed the abstracts of the remaining 189 papers and further excluded 98 papers. Once this was done, the third round commenced in which the two authors reviewed the full text of the remaining 91 papers independently and checked them for relevance to the four RQs. This third round resulted in the final set of 44 papers, which were included in this mapping study. These two authors had differences in their recommendations for inclusion or exclusion, and those issues were resolved through discussion. It is worth noting that the majority of the excluded papers were either technological solutions for user feedback collection or applications of feedback in other areas rather than RE.

### 3.4 Data extraction

To answer the four RQs defined in Section 3.1, we used the data extraction form in Table 2 to extract data items from the 44 primary studies. Specifically, for RQ1 and RQ3, the data items are derived from the research goals and/or main contributions of selected studies. The data items answering RQ2 are directly identified in the empirical evaluation or experiments reported in the 44 primary studies. Regarding RQ4.1 and RQ4.2, the data items are directly detected in the publication information and authors' affiliations of the selected studies.

#### TABLE 2 Data extraction form

| Relevant RQ | Data Item | Description |
|-------------|-----------|-------------|
| RQ1 | Sources of user feedback | Where was the user feedback collected from in this study? Which type of software systems was the user feedback referring to in this study? |
| RQ1 | Software type | |
| RQ2 | Selected metadata of user feedback | Which metadata (including attributes and/or aspects) of user feedback were not only collected but are also actually applied in this study? |
| RQ3 | RE activity/activities | Which activity/activities (i.e., the activities introduced in Section 3.1) was/were mentioned in this study? |
| RQ4.1 | Publication type | Is the study published as a journal, conference, or workshop paper? |
| RQ4.1 | Publication venue | In which journal, conference, or workshop was the study published? |
| RQ4.1 | Publication year | In which year was the study published? |
| RQ4.2 | Author’s affiliation | Which organizations were all the authors of the study working with and which countries were all the authors’ affiliations of the study located in? |
We performed the mappings study according to the steps described in Section 3. In this section, we report on the results of this mapping study to answer each of our RQs defined in Section 3.1.

4.1 Sources of crowdsourced user feedback (RQ1)

This subsection presents the distribution of explicit and implicit user feedback that has been employed in RE, according to the origins of those two types of crowdsourced feedback. On the basis of the definition of explicit and implicit user feedback (in Section 3.1), we first classified the selected studies in Table 3. As it shows, 93.2% of the included studies (41 out of 44 studies) employed explicit crowdsourced user feedback, such as online user reviews of Apps or other software products, in RE activities. Whereas, only three studies used implicit feedback for specified RE purposes.

| Feedback Type | No. of Studies | Studies |
|---------------|----------------|---------|
| Explicit      | 41             | S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11, S12, S13, S14, S15, S16, S17, S18, S19, S20, S21, S22, S23, S24, S25, S26, S27, S28, S29, S31, S32, S33, S34, S35, S36, S38, S39, S40, S41, S42, S43 |
| Implicit      | 3              | S30, S37, S44 |

FIGURE 2 Distribution of the selected studies over the platforms of user feedback
Furthermore, Figure 2 zooms in and presents the distribution of the included studies over the sources of user feedback. In this mapping study, we focused on the types of software systems that the crowdsourced user feedback reported about and the platforms that the feedback was collected through. In the left part of Figure 2, there are 32 out of the 44 studies using explicit user feedback of Apps, i.e., online app reviews. More specifically, Apple App Store and Google Play are the two most popular app repositories in our primary studies: six out of the 41 studies employed user feedback from the Apple App Store, 11 studies used Google Play, and 12 studies employed both of them. Besides, two studies (S1 and S23) adopted user feedback in unspecified Android markets, and only one study (S39) extended the collection of user feedback from Microsoft Store. The right half of Figure 2 shows the other platforms from which the crowdsourced user feedback is collected for RE purposes. It is observed that websites (e.g., Amazon used in S27) and platforms/forums (e.g., Steam forum for action games used in S28 and SourceForge.net for Open Source Software used in S21 and S22) were two main sources for researchers and practitioners to crawl explicit user feedback of other types of software systems, with seven and four studies respectively. Next, only one study worked on user feedback from social media (i.e., LinkedIn in S40).

Regarding the implicit user feedback, S30 and S44 adopted user behaviors of a certain app and user logs of a certain application software, respectively. While S37 employed service performance, usage, and feedback knowledge for requirements management, it did not specify any details of user feedback.

4.2 Metadata of crowdsourced user feedback (RQ2)

Generally, the collected dataset of crowdsourced user feedback may contain several pieces of information. However, in the set of 44 included studies, we found that not all the pieces of information stored in the database were used for RE purposes. This subsection presents those metadata of user feedback that has been reported in published literature for RE purposes.

In the 41 out of 44 included studies employing explicit user feedback, Table 4 indicates six pieces of metadata that are explored to support RE activities. It shows that all the 41 studies employed the text content analysis at the level of words, phrases, and/or whole sentences. However, there are some particularities: eight out of these 41 studies exploit the length of text (see the second row in record No. 1), and two studies consider the tense of verbs in the content of user feedback (see the third row in record No. 1).

In addition, 23 out of the 41 studies complemented the text content analysis with the rating of the user feedback; seven studies—with the submission date of a user feedback; seven other studies—with version number of the software system that the user feedback reported about; six studies—with the title of the user feedback; and five studies—with total number of user feedback. An interesting observation is that some included studies employed more than one piece of metadata for RE purposes. For example, see row number 4 in Table 4: therein, except S25, the other six studies that account for the version number of the software that the user feedback points to, also account for the "submission date." Another example is that five studies that report total number of user feedback (see row number 6) also indicate the specified software version. This makes good sense in these six studies’ contexts because therein the purpose of analysing crowdsourced user feedback is to check if a new release of software actually implemented the requirements that were collected from the crowd, on the basis of the crowd’s experience with the previous release.

Furthermore, we investigated how many pieces of metadata are combined within each of our 41 selected studies employing explicit user feedback. Figure 3 shows that 15 studies employed only one piece of metadata, i.e., the text content of the user feedback itself; 13 studies used two pieces of metadata; three studies adopted three pieces of metadata; three studies employed four pieces of metadata; six studies used five pieces of metadata.

### Table 4 Distribution of metadata of explicit user feedback

| No. | Metadata and its Attributes | No. of Studies | Studies |
|-----|-----------------------------|----------------|--------|
| 1   | Text content                | 41             | S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11, S12, S13, S14, S15, S16, S17, S18, S19, S20, S21, S22, S23, S24, S25, S26, S27, S28, S29, S31, S32, S33, S34, S35, S36, S38, S39, S40, S41, S42, S43 |
|     | length of text              | 8              | S5, S10, S14, S15, S16, S18, S24, S34 |
|     | tense of text               | 2              | S14, S15 |
| 2   | Rating                      | 23             | S3, S4, S5, S7, S9, S10, S11, S13, S14, S15, S16, S17, S18, S19, S21, S22, S25, S26, S27, S36, S38, S40, S43 |
| 3   | Submission date             | 7              | S3, S5, S9, S18, S19, S26, S35 |
| 4   | Version                     | 7              | S5, S9, S18, S19, S25, S26, S35 |
| 5   | Title                       | 6              | S7, S8, S10, S14, S15, S16 |
| 6   | Total number                | 5              | S5, S16, S19, S25, S26 |
of metadata; and one study adopted six pieces of metadata. One observation is that 63.4% of these 41 studies with explicit user feedback (26 out of 41 studies) employed more than one piece of metadata of user feedback for RE purposes. The other observation is that user feedback from other types of software (not Apps) usually adopted one or two pieces of metadata.

Regarding the three studies (S30, S37, and S44) using implicit crowdsourced user feedback, S30 employed behavior records of Apps and user context data (eg, location and motion information), S37 concentrated on service performance, usage, and feedback knowledge, and S44 employed user logs of the specified software.

### 4.3 Types of the RE activities using user feedback (RQ3)

This subsection presents those RE activities that used crowdsourced user feedback and realized its benefits according to the published literature.

We applied the data extraction strategies defined in Section 3.4 to identify the RE activities mentioned in the included studies. Figure 4 shows the distribution of included studies over RE activities and time period, where the number in the bracket under the name of each RE activity denotes the number of included studies that explicitly reported this type of RE activities. We found that RElic and RA are the two most reported RE activities, covering 38 and 33 out of 44 studies, respectively. Regarding RElic, for example, it was reported as one of the keywords of both S12 and S13, represented as "identification of requirements" in the title of S39, and derived from the aims of S24—"we propose SAFE, a novel uniform approach to extract app features from the single app pages, the single reviews and to match them." Similarly, S14, S15, and S20 reported "requirements classification," one subactivity of RA, in their titles and research objectives. Considering RMgt, it got support from eight out of 44 studies by using

![Distribution of selected studies over the number of used metadata](image1.png)

**FIGURE 3** Distribution of selected studies over the number of used metadata

![Distribution of the selected studies over RE activities and time period](image2.png)

**FIGURE 4** Distribution of the selected studies over RE activities and time period
crowdsourced user feedback, whereas, only two studies (S18 and S38) mentioned RSp, and one study (S38) reported RV. Table 5 provides the exemplary details on how the included studies used explicit/implicit feedback in different types of RE activities.

Furthermore, it was observed that 72.7% of the included studies (32 studies) reported more than one type of RE activities. Table 6 lists the distribution of RE activities over the included studies. More specifically, 12 included studies reported only one RE activity. Of those, RA was reported in five studies and RElic was reported in eight studies. RElic and RA are forming the most common “pattern” of RE activities in the included studies, since 23 studies reported both activities. Moreover, four studies mentioned both RElic and RMgt, and only one study (S18) used crowdsourced user feedback to support both RA and RSp. Considering S12, S32, and S37, each of them identified three types of RE activities, ie, RElic, RA, and RSp. Finally, only S38 stated to support all the five RE activities.
We make three observations from Figure 4: (a) RElic and RA are reported to be two popular RE activities where the role of crowdsourced user feedback is extensively explored; (b) other types of RE activities, including RSp, RV, and RMgt, also benefit from crowdsourced user feedback, although with less evidence; and (c) RElic, RA, and RMgt are getting more and more attention in the RE community in the last 3 years.

4.4 | Study classification by publication venue and year (RQ4.1)

This subsection shows the sources and the year in which the primary studies have been published.

The 44 included studies were published in three publication types: Conference, Journal, and Workshop. Table 7 shows the distribution of included studies across types. Conference papers are the most preferred publication type with 75.6% (34 studies), and both workshop papers and journal papers are less preferred with 11.4% (five studies).

Furthermore, Figure 5 shows the distribution of included studies published from 2007 to 2017. The first study on this research topic was published in 2007. However, very little research output was published until 2012. One reason for this could be that both the Apple App Store and Google Play were launched in 2008, and several years are needed to attract users to the Apps and make them willing to post feedback to those Apps. The number of studies using crowdsourced user feedback in RE activities has been increasing rapidly for the past 2 years. We note that the data point on 2017 is not indicative since the data of 2017 is incomplete. The main reason is that although the automatic search was performed in the beginning of 2018 to scope the related studies from 2007 to 2017, the studies accepted or published in the last quarter of 2017 might not be covered in the digital databases at that time.

Table 8 shows the publication sources of all the 44 included studies, their types, and the number of studies published in these sources. Overall, 32 publication sources are identified, which means that the review topic has received a wide attention in multiple subareas of computer science and information systems. Of these 32 sources, 22 publication sources (marked in italics in Table 8) are the journals, conferences, and workshops on SE and RE (such as ICSE, ASE, and RE), with 33 included studies (75% of 44 studies).

4.5 | Study classification by affiliation (RQ4.2)

Figure 6 shows the distribution of included studies by authors’ affiliations. We found that 41 out of the 44 selected studies (93%) were authored by academic researchers from universities or research institutes. Two studies (S2 and S34, 5%) came out of industry-university collaboration, and one study (S4, 2%) only was from industry.

TABLE 7  Distribution of the selected studies over publication types

| Publication Type | No. of Studies | %   |
|------------------|----------------|-----|
| Conference       | 34             | 77.3|
| Workshop         | 5              | 11.4|
| Journal          | 5              | 11.4|
| Total            | 44             | 100 |

![FIGURE 5  Distribution of the selected studies over the time period of 2007 to 2017](image)
| Publication Source                                                                 | Type                  | No. of Studies | Publication Source                                                                 | Type                  | No. of Studies |
|----------------------------------------------------------------------------------|-----------------------|----------------|-----------------------------------------------------------------------------------|-----------------------|----------------|
| International requirements engineering conference                                 | Conference            | 5              | Asia Pacific symposium on requirements engineering in the big data era              | Conference            | 1              |
| International conference on software engineering                                 | Conference            | 3              | International symposium on software reliability engineering                          | Conference            | 1              |
| International conference on automated software engineering                        | Conference            | 3              | Working conference on software visualization                                        | Conference            | 1              |
| International conference on evaluation and assessment in software engineering      | Conference            | 3              | International conference on information science and security                        | Conference            | 1              |
| International symposium on foundations of software engineering                    | Conference            | 2              | International conference on computer and information science                        | Conference            | 1              |
| International conference on software maintenance and evolution                    | Conference            | 2              | Australasian software engineering conference                                        | Conference            | 1              |
| Journal of systems and software                                                  | Journal               | 1              | International joint conference on computer science and software engineering         | Conference            | 1              |
| IEEE software                                                                     | Journal               | 1              | Pacific-Asia conference on advances in knowledge discovery and data mining           | Conference            | 1              |
| Empirical software engineering                                                    | Journal               | 1              | Asia-Pacific symposium on Internetware                                             | Conference            | 1              |
| Requirements engineering                                                          | Journal               | 1              | International conference on software engineering, artificial intelligence, networking and parallel/distributed computing | Conference            | 1              |
| Information systems and E-business management                                     | Conference            | 1              | Workshop on software evolution and Evolvability at international conference on automated software engineering | Workshop              | 1              |
| International conference on software analysis, evolution, and reengineering       | Conference            | 1              | Workshop on software evolution and Evolvability at international conference on automated software engineering | Workshop              | 1              |
| International symposium on empirical software engineering and measurement         | Conference            | 1              | Workshop on software engineering methods for service-oriented architecture           | Workshop              | 1              |
| Annual computer software and applications conference                               | Conference            | 1              | International workshop on crowd-based requirements engineering                      | Workshop              | 1              |
| Australian computer-human interaction conference                                   | Conference            | 1              | Workshop on social media world sensors at international conference on language resources and evaluation | Workshop              | 1              |
| International conference on software engineering and knowledge engineering        | Conference            | 1              | Workshop on privacy preserving IR at annual international conference on Research and Development in information retrieval | Workshop              | 1              |
| International working conference on requirements engineering: Foundation for Software Quality | Conference            | 1              | Workshop on privacy preserving IR at annual international conference on Research and Development in information retrieval | Workshop              | 1              |
| Subtotal:                                                                         | 29                    |                | Subtotal:                                                                         | 15                    |                |
| Total:                                                                           | 44                    |                |                                                                                   |                       |                |

**FIGURE 6** Distribution of the included studies over authors’ associations
Next, Figure 7 shows the distribution of included studies over the countries of the authors’ affiliations. Fifteen countries are reported to be active in crowd-based RE community. In particular, Germany, China, and the United States are the three main contributors with 10 studies, nine studies, and six studies, respectively. Moreover, we observed that five European countries—Germany, Italy, Switzerland, the Netherlands, and Poland—contributed 18 studies (40.9% of 44 studies). Also, we found that eight out of the 44 selected studies came out of collaboration between countries.

5 | DISCUSSION

This section presents our reflection on the answers to the four RQs.

5.1 | The use of explicit and implicit feedback (RQ1)

The results of this mapping study make us think that despite its acknowledged importance, our knowledge on the use of explicit and implicit user feedback in RE is limited. In turn, this calls for more investigation in multiple directions.

First, regarding the use of explicit feedback, we found that more than three quarters of the included studies on explicit user feedback (32 out of 41 studies), did concentrate on app reviews. In fact, we know very little about the use of explicit feedback in the context of other types of software systems. In turn, more research in a variety of contexts is needed in order to gain a more complete understanding of the benefits and the applicability of user feedback in RE.

Second, our results clearly call for more investigation on employing implicit user feedback in RE. As observed in Section 4.1, only 7% of the included studies (three out of 44 studies) used implicit user feedback. One reason could be that normally, implicit feedback, such as user usage and behaviors, is stored in private databases supported by software vendors. In contrast to explicit user feedback, researchers and/or practitioners cannot get implicit feedback in an easy and convenient manner. Another possible reason for having just around 7% of the included studies on implicit feedback is that researchers working on the use of implicit feedback, publish in other venues that have their own specific terminology and vocabulary and, hence, do not use the key words that the RE community uses. For example, it might well be possible that living labs that generate streams of behavior data are also leveraged for RE purposes. A hint to this is the paper of Coetzee et al.,30 which describes the principles of a living lab approach to the design-level requirement for information platforms in Africa’s rural areas. As living labs are a component of many interuniversity projects funded by the European Union research agencies, we assume that it might be likely to expect more RE research based on crowdsourced feedback in living lab settings.

5.2 | Attributes of user feedback (RQ2)

Regarding the different attributes of crowdsourced user feedback, the results presented in Section 4.2 identified six pieces of feedback metadata. Specifically, text content was the metadata reported the most frequently (in 41 out of 44 studies). This is not surprising, because in RE, content analysis has been widely used in the past to process the verbal output of requirements elicitation interviews.
Furthermore, more than 60% of the primary studies employing explicit user feedback (26 out of 41 studies) combined at least two pieces of metadata for various RE purposes. One might assume that the effectiveness of those combinations of metadata might well be contingent on the RE activity that the feedback is meant to support. However, more empirical research is needed to understand how to combine identified metadata of user feedback to support various RE purposes and what combinations would be more effective in what kinds of research contexts, on the basis of various purposes of using the crowdsourced feedback in RE.

Moreover, six out of seven studies using release number of software also combined submission date, and three studies (S5, S19, and S26) using submission date have reported to use both version of software and total number of reviews collected from the specified software. This indicates that there could be some combination patterns of the metadata of user feedback for specific RE activities. To know for sure, more empirical research with companies is needed.

5.3 Activities employing user feedback (RQ3)

As observed in Section 4.3, our mapping study reported that RElic and RA were the two most popular RE activities in which crowdsourced user feedback was employed. We think that the reasons for this observation could be the following. First, it makes a lot of business sense to leverage the crowdsourced feedback for RElic and RA. Second, these two activities are relatively easy to investigate by using the empirical research methods available for feedback analysis.

Unlike RElic and RA, we found that RSp, RV, and RMgt did not get enough attention from researchers. Therefore, we think more exploration is needed on how crowdsourced feedback could support these three RE activities.

Furthermore, nearly 75% of the studies (32 out of 44) reported that the use of feedback was beneficial for more than just one RE activity. For example, S29 reported requirements identification (covered in RElic) and classification (covered in RA) in the title and research objectives; while RElic and RMgt can be derived from the main contribution and aims of S7, stating as “We use natural language processing techniques to identify fin-grained app features in the reviews” and “The extracted features were coherent and relevant to requirements evolution tasks,” respectively. This opens up the question of which other activities could be combined to jointly benefit from the crowdsourced feedback in order to maximize its value in the end-to-end RE process.

5.4 Demographics of the studies (RQ4)

This mapping study clearly indicates that the application of crowdsourced user feedback for RE purposes is a rising research area for scholars. We conclude this from the growth of published output (Figure 5) in the recent years. Moreover, 87% of the included studies are published in conferences or workshops, and only five out of the 44 included studies are journal publications. In fact, two of four included journal papers are the extensions of the corresponding published conference papers.

Despite the rise observed, producing more diverse evidence on how user feedback is useful and tracing the use of feedback to the SWEBOK’s fundamental RE activities seems urgent, if we would like to understand completely the possible range of roles that feedback may play in RE. Next, it is disappointing to see that there were only two studies (around 4% of 44 studies) authored by practitioners from industry. This is surprising knowing that app development organizations actively have been using crowdsourced feedback for years. This also signals a question regarding the extent to which the results of the studies are industry-relevant and realistic. We think therefore, that more case study research with companies is necessary to understand what crowdsourced feedback aspects best feed into RE and how this happens in real life.

In addition, the 44 included studies were published in 32 different venues in multiple disciplines and contributed by the authors from 15 countries in Asia, Europe, and North America. This indicates that extensive attention on this research topic is being paid from researchers not only in a broad range of research interests, but also with affiliations spreading in different continents. Furthermore, 22 publication venues out of 32 are classified into the fields of RE and SE, indicating that the current application domains of our research topic are narrow and focused, although the exploration of other disciplines is emerging.

6 IMPLICATIONS

This mapping study has some implications for RE researchers, practitioners, and RE educators.

From a research standpoint, our results have at least three implications. First, RE researchers should expand their research to cover other RE activities. Specifically, the extent to which crowdsourced user feedback could be effectively employed in activities such as requirements prioritization and management is still arguable. While focusing on RElic, the RE community left the other subareas of RE much under-researched. This opens up a number of interesting research questions to be explored in the future. For example, would our prioritization practices change if we deal with the crowd and not with real stakeholders and their goals in organizations (which is the case of “traditional” RE where stakeholders play
a pivotal role? Assuming the crowd is treated similarly as an important stakeholder, who in a software development organization could represent the crowd and its requirements? Who could or should negotiate on behalf of the crowd? These questions form a line for future research. Second, knowing the RE is a social process in nature, it is a surprise that no study addresses organizational questions related to how RE takes place in companies that exploit crowdsourced user feedback, be it app development firms. We therefore think that the RE community needs to focus on tackling the use of using crowdsourced user feedback in RE, from a “process perspective”: if RE researchers want their research to be industry-relevant, they need to position their studies on exploitation of user feedback against a larger organizational process, namely, the RE process, which is composed of activities having their inputs and outputs and being coordinated. One might assume that RE based on crowdsourced feedback might call for using coordination models different compared with those used in the RE processed described in RE textbooks. To confirm or disconfirm this, more research is required. Third, from the standpoint of generalizability, our current knowledge of crowd-sourced RE is skewed. This is because the published scientific evidence comes out of academic research that is mostly treated the use of explicit feedback and that originated in European and Asian contexts. More realism of the results could and should be achieved if empirical research efforts focus on studies with industry and in other countries.

Our study has some implications for practicing requirements engineers. First, our results suggest that it seems safe to employ crowdsourced feedback in requirements elicitation and requirements analysis for app development. We therefore think that practitioners should consider the use of crowdsourced user feedback in their RE processes as a viable option for learning. One can imagine that this way companies could quickly assemble improvement ideas for the evolution of their apps independent of the specific life cycle models they use for the app development itself. Second, an interesting question from practitioners’ perspective is how to balance the voice of the crowd against the requirements that RE specialists could collect through user focus groups or user surveys. We encourage practitioners to try out these alternative techniques and compare the ways in which the requirements that come out are prioritized later on. Only then, practitioners will know if the crowdsourced user feedback can repeatedly provide the most relevant requirements in a cost-effective way. Third, from a software development organization’s perspective, crowdsourced feedback is a resource available for systematic use that practitioners should incorporate into the larger process of RE. Leveraging crowdsourced user feedback organizationally in terms of roles, responsibilities, and processes seems however to be an organization’s tacit knowledge. This means, if a practitioner wants to learn RE techniques utilizing crowdsourced user feedback, it might be a good idea to join an app development organization that uses feedback instead of searching for RE how-to textbooks.

Our study has some implications for RE educators. Until now, RE textbooks have no content dedicated to RE processes relying on crowdsourced user feedback. In turn, many teachers in computer science schools do not provide their students with adequate knowledge on the potential value of crowd-based RE techniques. Assuming that companies’ adoption of these techniques would grow—especially in the era of Internet-of-Things systems—we as teachers need to help students be prepared for the current market developments and provide them with the skills that match it. This implies creating awareness of the use of explicit and implicit crowdsourced feedback in RE as a viable approach and informing them about its strengths and limitations. At least, our students should know more about using crowdsourced feedback in specific RE activities such as RElic and RA.

7 | THREATS OF VALIDITY

The results of this mapping study may be influenced by the coverage of study search, bias on study selection and personal judgment in study data extraction. Therefore, according to the guidelines in Shull et al and Wohlin et al, four types of threats to validity of the review results are discussed below.

7.1 | Conclusion validity

For our systematic mapping study to be reproducible by other researchers, we developed a study protocol defining our search strategy and study selection procedure with use of IC and EC. However, different researchers may have different understanding on these criteria, and in turn, might bring different results of their study selection. To reduce researchers’ bias, our study protocol was discussed by all authors, which assured a common understanding on study selection. Plus, in the second round study selection, the first author performed a pilot selection and the other two authors joined the discussion to reach a consensus on understanding the selection criteria. After that, the second round selection was conducted by the first two authors. Furthermore, in the third round study selection, the first and the second authors conducted the selection process in parallel and independently, and then harmonized their selection results to mitigate the personal bias in study selection caused by individual judgments.

Second, our data extraction might influence the classification results of the selected studies as it included the researchers’ personal judgment. To mitigate this, we used a template (as per Dybå et al) to describe the data retrieved from the primary studies. The template was an input to our qualitative synthesis.
7.2 | Construct validity

In this work, user feedback, crowdsourcing, and requirements are the most important terms under consideration. To ensure that all authors had a shared interpretation of these key terms, we discussed the definitions of these concepts and reached a consensus on their understanding. Moreover, to make sure the high coverage of potentially relevant studies in automatic search, we improved the search terms according to the result of the trial search before the formal search.

7.3 | Internal validity

Since the data analysis in this systematic review only uses descriptive statistics, the threats to internal validity are minimal.

7.4 | External validity

The results of this mapping study were considered regarding the application of user feedback in crowd-based RE. Therefore, the presented classification of the selected studies and the conclusions drawn are only valid in the review topic. The predefined protocol is helpful to collect representative studies in the given review topic.

8 | CONCLUSIONS

On the basis of 44 selected publications, this mapping study provided an overview on the types of user feedback that have been employed for crowdsourced RE activities. Our study revealed the following. First, current research mainly concentrated on explicit user feedback of Apps. Our knowledge on the use of feedback in other types of systems is, in turn, incomplete. Plus, little is known about the use of implicit feedback in RE. Second, nine pieces of metadata are identified from the selected studies using explicit user feedback, however, nearly half of these metadata are rarely used. Third, requirements elicitation and requirements analysis are the most investigated RE activity in which crowdsourced feedback was employed. Fourth, only 7% of the included studies in this review came from industry settings, which indicates lack of industry-university research collaborations that led to published empirical research outcomes. Fifth, the topic of crowd-based RE using user feedback has received the most attention from researchers, most of whom are working at universities spreading in 12 countries, mostly located in Europe and Asia.

ACKNOWLEDGEMENTS

This research was supported by the National Key Research and Development Program of China (No. 2018YFB1003800) and the National Natural Science Foundation of China (Nos. 61702378 and 61672387).

ORCID

Chong Wang  
https://orcid.org/0000-0003-4576-5392

Maya Daneva  
https://orcid.org/0000-0001-7359-8013

REFERENCES

1. Maalej W, Nayebei M, Johann T, Ruhe G. Toward data-driven requirements engineering. IEEE Software. 2016;33(1):48-54.

2. Kitchenham BA, Budgen D, Brereton P. Evidence-Based Software Engineering and Systematic Reviews. Chapman & Hall; 2015.

3. Asghar MZ, Khan A, Ahmad S, Kundi FM. A review of feature extraction in sentiment analysis. Journal of Basic and Applied Scientific Research. 2014;4(3):181-186.

4. Guo X, Wang H, Song Y, Hong G. Brief survey of crowdsourcing for data mining. Expert Systems with Applications. 2014;41(17):7987-7994.

5. Genc-Nayebei N, Abran A. A systematic literature review: opinion mining studies from mobile app store user reviews. Journal of Systems and Software. 2017;125:207-219.

6. Hetmank L. Components and functions of crowdsourcing systems-a systematic literature review. 2013. In Proceedings of International Conference on Wirtschaftsinformatik. (A3):55-69.

7. Hosseini M, Shahr A, Phalp K, Taylor J, Ali R. Crowdsourcing: a taxonomy and systematic mapping study. Computer Science Review. 2015;17:43-69.

8. Leicht, N, Durward, D, Blohm, I, Leimeister, JM. Crowdsourcing in Software Development: A State-of-the-Art Analysis. In Proceedings of the 28th Bled eConference (ISD'15), 2015:389-430.

9. Mao K, Capra L, Harman M, Jia Y. A survey of the use of crowdsourcing in software engineering. Journal of Systems and Software. 2017;126(4):57-84.

10. Morschheuser B, Hamari J, Koivisto J, Maedche A. Gamified crowdsourcing: conceptualization, literature review. And Future Agenda International Journal of Human-Computer Studies. 2017;106:26-43.
11. Rizk, NM, Ebada, A, Nasr, ES. Investigating mobile applications’ requirements evolution through sentiment analysis of users’ reviews. In Proceedings of the 11th International Computer Engineering Conference (ICENCO’15), 2015:123-130.

12. Zuchowski O, Posegga O, Schlagwein D, Fischbach K. Internal crowdsourcing: conceptual framework, structured review, And Research Agenda. Journal of Information Technology. 2016;31(2):166-184.

13. Ghezzi A, Gabelloni D, Martini A, Natalichio A. Crowdsourcing: a review and suggestions for future research. International Journal of Management Reviews. 2018;20(2):343-363.

14. Kitchenham, BA. Guidelines for Performing Systematic Literature Reviews in Software Engineering. EBSE Technical Report EBSE-2007-01. 2007.

15. Webster J, Watson RT. Analyzing the past to prepare for the future: writing a literature review. Management Information Systems Quarterly. 2002;26(2):xxiii-xxiii.

16. Brocke, JV, Simons, A, Niehaves, B, Riemer, K, Plattfaut, R, Clevien, A. Reconstructing the giant: on the importance of rigour in documenting the literature search process. In Proceedings of the 17th European Conference on Information Systems (ECIS’09), 2009:2206-2217.

17. Petticrew M, Roberts H. Systematic Reviews in The Social Sciences: A Practical Guide. John Wiley & Sons; 2008.

18. Boell SK, Cecez-Kecmanovic D. On being “systematic” in literature reviews in IS. Journal of Information Technology. 2015;30(2):161-173.

19. Davis MS. That’s interesting: towards a phenomenology of sociology and a sociology of phenomenology. Philosophy of the Social Sciences. 1971;1(2):309-344.

20. Short J. The art of writing a review article. Journal of Management. 2009;35(6):1312-1317.

21. Groen EC, Seyff N, Ali R, et al. The crowd in requirements engineering: the landscape and challenges. IEEE Software. 2017;34(2):44-52.

22. Nonaka I, Takeuchi H. The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation. Oxford University Press; 1995.

23. Maaie W, Thurimella AK. Managing Requirements Knowledge. Springer; 2013.

24. Jawaeer G, Weller P, Kostkova P. Modeling user preferences in recommender systems: a classification framework for explicit and implicit user feedback. ACM Transactions on Interactive Intelligent Systems. 2014;4(2):1-26.

25. Claypool, M, Le, P, Waseda, M, Brown, D. Implicit interest indicators. In Proceedings of the 6th International Conference on Intelligent User Interfaces (IUI’01), 2001:33-40.

26. Bourque, P, Fairley, RE. Guide to the software engineering body of knowledge, Version 3.0 (SWEBOK Guide V3.0), IEEE Computer Society. 2014.

27. Sommerville I. Integrated requirements engineering: a tutorial. IEEE Software. 2005;22(1):16-23.

28. Project Management Institute (PMI). A Guide to the Project Management Body of Knowledge. 4th ed. Newtown Square, Pennsylvania, USA: Project Management Institute, Inc.; 2008.

29. Harzing A-W, Satu A. Google scholar, Scopus and the web of science: a longitudinal and cross-disciplinary comparison. Scientometrics. 2016;106(2):787-804.

30. Mongeon P, Paul-Hus A. The journal coverage of web of science and Scopus: a comparative analysis. Scientometrics. 2016;106(1):213-228.

31. Howe J. The rise of crowdsourcing. Wired Magazine. 2006;14:1-4.

32. Coetzee M, Goss H, Meiklejohn C, Mlangeni P. A living laboratory approach in the design of the user requirements of a spatial information platform. African Journal of Town and Regional Planning. 2014;64:10-18.

33. Shull F, Singer J, Sjöberg Dag I.K. Guide to Advanced Empirical Software Engineering. Springer; 2008.

34. Wohlin C, Runeson P, Höst M, Ohlsson MC, Regnell B, Wesslén A. Experimentation in Software Engineering. Springer; 2012.

35. Dybå, T, Dingsøyr, T, Hanssen, GK. Applying systematic reviews to diverse study ypes: an experience report. In Proceedings of the 1st International Symposium on Empirical Software Engineering and Measurement (ESEM’07), Madrid, Spain, 2007:225-234.

How to cite this article: Wang C, Daneva M, van Sinderen M, Liang P. A systematic mapping study on crowdsourced requirements engineering using user feedback. J Softw Evol Proc. 2019;31:e2199. https://doi.org/10.1002/smr.2199

APPENDIX A

PRIMARY STUDIES

[S1] Carreño, Laura V. Galvis, Kristina Winbladh, “Analysis of user comments: an approach for software requirements evolution”, in Proceedings of the 35th International Conference on Software Engineering (ICSE’13), IEEE, San Francisco, USA, 2013, pp. 582-591.

[S2] Lei Cen, Luo Si, Ninghui Li, Hongxia Jin, “User comment analysis for android apps and CSPI detection with comment expansion”, in Proceedings of the 1st International Workshop on Privacy-Preserving IR: When Information Retrieval Meets Privacy and Security, co-located with 37th Annual International ACM SIGIR conference on Research and Development in Information Retrieval (PIR@SIGIR), Gold Coast, Australia, 2014, pp. 25-30.
[S3] Ning Chen, Jialiu Lin, Steven C. H. Hoi, Xiaokui Xiao, Boshen Zhang, "AR-miner: mining informative reviews for developers from mobile app marketplace", in Proceedings of the 36th International Conference on Software Engineering (ICSE'14), ACM, Hyderabad, India, 2014, pp. 767-778.

[S4] Emanuele di Rosa, Alberto Durante, "App2Check: a machine learning-based system for sentiment analysis of app reviews in Italian language", in Proceedings of the 2nd International Workshop on Social Media World Sensors, co-located with the 10th International Conference on Language Resources and Evaluation (LREC’16), Portoroz, Slovenia, 2016, pp. 8-13.

[S5] Cuiyun Gao, Baoxiang Wang, Pinjia He, Jieming Zhu, Yangfan Zhou, Michael R. Lyu, "PAID: Prioritizing app issues for developers by tracking user reviews over versions", in Proceedings of the 26th IEEE International Symposium on Software Reliability Engineering (ISSRE’15), IEEE Computer Society, Gaithersburg, USA, 2015, pp. 35-45.

[S6] Judith Gebauer, Ya Tang, Chaiwat Baimai, "User requirements of mobile technology: results from a content analysis of user reviews", Information Systems and e-Business Management, Vol. 6, No. 4, 2008, pp. 361-384.

[S7] Emitza Guzman, Walid Maalej, "How do users like this feature? A fine grained sentiment analysis of app reviews", in Proceedings of the 22nd International Requirements Engineering Conference (RE’14), Karlskrona, Sweden, 2014, pp. 153-162.

[S8] Emitza Guzman, Omar Aly, Bernd Bruegge, "Retrieving diverse opinions from app reviews", in Proceedings of ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM’15), Beijing, China, 2015, pp. 21-30.

[S9] Emitza Guzman, Padma Bhuvanagiri, Bernd Bruegge, "FAVe: Visualizing user feedback for software evolution", in Proceedings of the 2nd IEEE Working Conference on Software Visualization (VISSOFT’14), Victoria, Canada, 2014, pp. 167-171.

[S10] Emitza Guzman, Muhammad El-Haliby, Bernd Bruegge, "Ensemble methods for app review classification: an approach for software evolution", in Proceedings of the 30th IEEE/ACM International Conference on Automated Software Engineering (ASE’15), Lincoln, USA, 2015, pp. 771-776.

[S11] Leonard Hoon, Rajesh Vasa, Jean-Guy Schneider, Kon Mouzakis, "A Preliminary analysis of vocabulary in mobile app user reviews", in Proceedings of the 24th Australian Computer-Human Interaction Conference (OzCHI’12), Melbourne, Australia, 2012, pp. 245-248.

[S12] Wei Jiang, Haibin Ruan, Li Zhang, Philip Lew, Jing Jiang, "For user-driven software evolution: requirements elicitation derived from mining online reviews", in Proceedings of the 18th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD’14), Part II, Taipei, Taiwan, 2014, pp. 584-595.

[S13] Swetha Keerthi, Bastin Tony Roy Savarimuthu, Sherlock A. Licorish, "Approaches for prioritizing feature improvements extracted from app reviews", in Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering (EASE’16), Limerick, Ireland, 2016, pp. 33:1-33:6.

[S14] Walid Maalej, Hadeer Nabil, "Bug report, feature request, or simply praise? On automatically classifying app reviews", in Proceedings of the 23rd IEEE International Requirements Engineering Conference (RE’15), Ottawa, Canada, 2015, pp. 116-125.

[S15] Walid Maalej, Zijad Kurtanovic, Hadeer Nabil, Christoph Stanik, "On the automatic classification of app reviews", Requirements Engineering, Vol. 21, No. 3, 2016, pp. 311-331.

[S16] Stuart McIlroy, Nasir Ali, Hammad Khalid, Ahmed E. Hassan, "Analyzing and automatically labelling the types of user issues that are raised in mobile app reviews", Empirical Software Engineering, Vol. 21, No. 3, 2016, pp. 1067-1106.

[S17] Stuart McIlroy, Weiyi Shang, Nasir Ali, Ahmed E. Hassan, "Is it worth responding to reviews? A case study of the top free apps in the Google Play Store", IEEE Software, Vol. 34, No. 3, 2015, pp. 64-71.

[S18] Dennis Pagano, Walid Maalej, "User feedback in the appstore: An empirical study", in Proceedings of the 21st IEEE International Requirements Engineering Conference (RE’13), Rio de Janeiro-RJ, Brazil, 2013, pp. 125-134.

[S19] Fabio Palomba, Mario Linares Vásquez, Gabriele Bavota, Rocco Oliveto, Massimiliano Di Penta, Denys Poshyvanyk, Andrea De Lucia, "User reviews matter! Tracking crowdsourced reviews to support evolution of successful apps", in Proceedings of IEEE International Conference on Software Maintenance and Evolution (ICSME’15), Bremen, Germany, 2015, pp. 291-300.

[S20] Sebastiano Panichella, Andrea Di Sorbo, Emitza Guzman, Corrado Aaron Visaggio, Gerardo Canfora, Harald C. Gall, "How can I improve my app? Classifying user reviews for software maintenance and evolution", in Proceedings of IEEE International Conference on Software Maintenance and Evolution (ICSME’15), Bremen, Germany, 2015, pp. 281-290.

[S21] Zhenzheng Qian, Beijun Shen, Wenkai Mo, Yuting Chen, "SatIndicator: Leveraging user reviews to evaluate user satisfaction of SourceForge projects", in Proceedings of the 40th IEEE Annual Computer Software and Applications Conference (COMPSAC’16), Atlanta, USA, 2016, pp. 93-102.

[S22] Zhenzheng Qian, Chengcheng Wan, Yuting Chen, "Evaluating quality-in-use of FLOSS through analyzing user reviews", in Proceedings of the 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD’16), Shanghai, China, 2016, pp. 547-552.

[S23] Dong Sun, Rong Peng, "A scenario model aggregation approach for mobile app requirements evolution based on user comments", in Proceedings of the 2nd Asia Pacific Symposium on Requirements Engineering in the Big Data Era (APRES’15), Wuhan, China, 2015, pp. 75-91.
[S24] Timo Johann, Christoph Stanik, Alireza M. Alizadeh B., Wald Maalej, “SAFE: A simple approach for feature extraction from app descriptions and app reviews”, in Proceedings of the 25th International Requirements Engineering Conference (RE’17), Lisbon, Portugal, 2017, pp. 21-30.

[S25] Lorenzo Villarroel, Gabriele Bavota, Barbara Russo, Rocco Oliveto, Massimiliano Di Penta, “Release planning of mobile apps based on user reviews”, in Proceedings of the 38th International Conference on Software Engineering (ICSE’16), Austin, USA, 2016, pp. 14-24.

[S26] Phong Minh Vu, Tam The Nguyen, Hung Viet Pham, Tung Thanh Nguyen, “Mining user opinions in mobile app reviews: a keyword-based approach”, in Proceedings of the 30th IEEE/ACM International Conference on Automated Software Engineering (ASE’15), Lincoln, USA, 2015, pp. 749-759.

[S27] Phong Minh Vu, Hung Viet Pham, Tam The Nguyen, Tung Thanh Nguyen, “Phrase-based extraction of user opinions in mobile app reviews”, in Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering (ASE’16), Singapore, 2016, pp. 726-731.

[S28] Artinat Wattanaburanon, Nakornthip Prompoon, “Method for classifying usability qualities and problems for action games from user reviews using text mining”, in Proceedings of the 15th IEEE/ACIS International Conference on Computer and Information Science (ICIS’16), Okayama, Japan, 2016, pp. 1-6.

[S29] Hui Yang, Peng Liang, “Identification and classification of requirements from app user reviews”, in Proceedings of the 27th International Conference on Software Engineering and Knowledge Engineering (SEKE’15), Pittsburgh, USA, 2015, pp. 7-12.

[S30] Wenjie Liang, Wenyi Qian, Yijian Wu, Xin Peng, Wenyun Zhao, “Mining context-aware user requirements from crowd contributed mobile data”, in Proceedings of the 7th Asia-Pacific Symposium on Internetware (Internetware’15), Wuhan, China, 2015, pp. 132-140.

[S31] Kittiya Srewutmanapitikul, Pornsiri Muengchaisri, “Prioritizing software maintenance plan by analyzing user feedback”, in Proceedings of International Conference on Information Science and Security (ICISS’16), Pattaya, Thailand, 2016, pp. 1-5.

[S32] Adelina Ciurumelea, Andreas Schaufelbuhl, Sebastiano Panichella, Harald C. Gall, “Analyzing reviews and code of mobile apps for better release planning”, in Proceedings of the 24th IEEE International Conference on Software Analysis, Evolution, and Reengineering (SANER’17), Klagenfurt, Austria, 2017, pp. 91-102.

[S33] Nishant Jha, Anas Mahmoud, “Mining user requirements from application store reviews using frame semantics”, in Proceedings of the 23rd International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ’17), Essen, Germany, 2017, pp. 273-287.

[S34] Andrea Di Sorbo, Sebastiano Panichella, Carol V. Alexandru, Junji Shimagaki, Corrado Aaron Visaggio, Gerardo Canfora, Harald C. Gall, “What would users change in my app? Summarizing app reviews for recommending software changes”, in Proceedings of the 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering (FSE’16), Seattle, USA, 2016, pp. 499-510.

[S35] Sherlock A. Licorish, Amjed Tahir, Michael Franklin Bosu, Stephen G. MacDonell, “On satisfying the android OS community: user feedback still central to developers’ portfolios”, in Proceedings of the 24th Australasian Software Engineering Conference (ASWEC’15), Adelaide, Australia, 2015, pp. 78-87.

[S36] Warit Leopairote, Athatis Surarerks, Nakornthip Prompoon, “Evaluating software quality in use using user reviews mining”, in Proceedings of the 10th International Joint Conference on Computer Science and Software Engineering (JCSSE’13), Khon Kaen, Thailand, 2013, pp. 257-262.

[S37] Henk van der Schuur, Slinger Jansen, Sjaak Brinkkemper, “Becoming responsive to service usage and performance changes by applying service feedback metrics to software maintenance”, in Proceedings of 4th International ERCIM Workshop on Software Evolution and Evolvability (EvOL’08) at 23rd IEEE/ACM International Conference on Automated Software Engineering (ASE08), L'Aquila, Italy, 2008, pp. 53-62.

[S38] Daniel Lübke, Eric Knauss, “Dealing with user requirements and feedback in SOA Projects”, in Proceedings of the Workshop on Software Engineering Methods for Service-Oriented Architecture (SEMSOA’07), Hannover, Germany, 2007, p. 16-30.

[S39] Sebastiano Panichella, Andrea Di Sorbo, Emizia Guzman, Corrado Aaron Visaggio, Gerardo Canfora, Harald C. Gall, “ARdoc: app reviews development oriented classifier”, in Proceedings of the 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering (FSE’16), Seattle, USA, 2016, pp. 1023-1027.

[S40] Eduard C. Groen, Sylwia Kopczyńska, Marc P. Hauer, Tobias D. Krafft, Joerg Doerr, “Users - the hidden software product quality experts?: A study on how app users report quality aspects in online reviews”, in Proceedings of the 25th International Requirements Engineering Conference (RE’17), Lisbon, Portugal, 2017, pp. 80-89.

[S41] Mengmeng Lu, Peng Liang, “Automatic classification of non-functional requirements from augmented app user reviews”, in Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering (EASE17), Karlskrona, Sweden, 2017, pp. 344-353.

[S42] Pratyush K. Srivastava, Richa Sharma, “Crowdsourcing to elicit requirements for MyERP application”, in Proceedings of the 1st International Workshop on Crowd-Based Requirements Engineering (CrowdRE’15), Ottawa, Canada, 2015, pp. 31-35.

[S43] Sherlock A. Licorish, Bastin Tony Roy Savarimuthu, Swetha Keeritipati, “Attributes that predict which features to fix: Lessons for app store mining”, in Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering (EASE’17), Karlskrona, Sweden, 2017, pp. 108-117.

[S44] Lin Liu, Qing Zhou, Jilei Liu, Zhanqiang Cao, “Requirements cybernetics: Elicitation based on user behavioral data”, Journal of Systems and Software, 2017, 124: 187-194.
## APPENDIX B
### COMPARISON BETWEEN RELEVANT RESEARCH AND OUR WORK

Table 9. Comparison between relevant research and our work

| Reference | Research Goal | Focus/Purpose of the Literature Study | Type of Study | Is the use of Feedback in RE Activities Explicitly Discussed? |
|-----------|---------------|--------------------------------------|---------------|--------------------------------------------------------------|
| Hetmank² | To understand the crowdsourcing research area from a system point of view. | To identify components and functions of a crowdsourcing system that can be conceptualized. | SLR on the basis of the guidelines of Kitchenham,¹⁴ | No. Technical design of crowdsourcing system is the focus of the study. |
| Ashgar et al³ | To understand the state-of-the-art techniques of feature extraction in sentiment analysis and opinion mining. | (1) To identify approaches for feature extraction in sentiment analysis and opinion mining. (2) To identify areas well-covered by researchers and those the least addressed. (3) To identify most and least commonly used feature selection techniques to find research gaps for future work. | SLR, guidelines are not explicitly mentioned | No. Feature extraction approaches are examined in terms of underlying assumptions and underlying logic. |
| Guo et al⁴ | To understand the state-of-the-arts on the crowdsourcing for data mining. | (1) To identify the challenges and opportunities of data mining tasks using crowdsourcing, and summarize the framework of them. (2) To formulate a general framework of crowdsourcing for data mining, which includes question design, mining process and quality control. | Type of study and guidelines are not explicitly stated. | No. Data mining techniques are examined in terms of their own characteristics when used in crowdsourcing environments. |
| Leicht et al⁸ | To understand the state-of-the-art in crowdsourcing in software development. | To propose a fundamental framework with dimensions to structure the existing. Insights of crowdsourcing in the context of software development and to derive a research. Agenda to guide further research. | SLR based on the guidelines of Webster & Watson¹⁵ and Brocke et al¹⁶ | No. Organizational aspects of crowdsourcing in software development are examined. |
| Rizk et al¹¹ | To understand the state-of-the-art methods and tools of using sentiment analysis on mobile apps users’ reviews in order to extract user requirements for building new applications or enhancing existing ones, ie, requirements evolution. | To identify published studies on sentiment analysis techniques and tools used on mobile apps reviews of users, for the purpose of requirements evolution. | SLR based on the guidelines of Kitchenham.¹⁴ | Yes. The authors examine those techniques that explicitly extract user requirements for building new applications or enhancing existing ones; ie, requirements evolution. |
| Hosseini et al⁷ | (1) To understand the current status and the degree of maturity of the field and how it is being applied, when, where and by whom. (2) To identify gaps in the literature where additional research has to be conducted, as well as areas which got popularity and emphasis more than others. | (1) To identify the fields of study of the papers which have proposed a definition for crowdsourcing, the types of research conducted in those papers, the forms of study those papers have followed, the venues those papers have been published in, the demographics the researchers of those paper represent, and the commonality | SMS based on the guidelines of Petticrew and Roberts¹⁷ | No. Crowdsourcing as a phenomenon across multiple fields is examined. |
| Reference                  | Research Goal                                                                                                                                                                                                 | Focus/Purpose of the Literature Study                                                                                                                                                                                                 | Type of Study                                                                                                                                  | Is the use of Feedback in RE Activities Explicitly Discussed? |
|---------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Zuchowski et al<sup>12</sup> | To provide conceptual development, synthesize the literature, and provide a research agenda. (what exactly is internal crowdsourcing? What do we know about internal crowdsourcing? Where are we lacking in knowledge about internal crowdsourcing?) | To come up with a definition of internal crowdsourcing and a framework for understanding.                                                                                                                                                                                              | SLR based on the guidelines of Kitchenham<sup>14</sup>                                                                                          | No. The so-called "internal crowdsourcing" as an organizational phenomenon is examined.         |
| Mao et al<sup>9</sup>      | To provide a comprehensive survey of the crowdsourcing in SE.                                                                                                                                               | (1) To provide a comprehensive survey of the current research progress on using crowdsourcing to support SE activities.  
(2) To summarize the challenges for crowd-sourced SE and to reveal to what extent these challenges were addressed by existing work.                                      | Type of study and guidelines are not explicitly stated.                                                                                         | No. The authors examined definitions of the term "crowdsourcing" and the application of the crowdsourcing concept to the area of software engineering |
| Genc-Nayebi and Abran<sup>5</sup> | To identify proposed solutions for mining online opinions in app store user reviews, challenges and unsolved problems in the domain, any new contributions to software requirements evolution, and future research directions. | To identify mobile app store studies, the challenges faced when mining app store data according to these studies, how these challenges have been overcome, and any unsolved challenges.                                                   | SLR based on the guidelines of Kitchenham<sup>14</sup>                                                                                          | Yes. The authors examined those techniques that mine opinions for requirements evolution purposes. |
| Morschheuser<sup>10</sup>  | To provide a comprehensive overview and outlook of the usage and study of gamification in crowdsourcing systems.                                                                                           | To examine and compare the characteristics of gamified crowdsourcing systems, the effectiveness of gamification in crowdsourcing, and use this information to formulate an agenda for future research.                                    | Mapping study based on the guidelines of Webster and Watson<sup>14</sup> and Boell and Cecez-Kecmanovic<sup>18</sup>.                       | No. Gamification is examined in crowdsourcing systems.                                          |
| Ghezzi et al<sup>13</sup>  | To revisit the definition of crowdsourcing, identify controversies and common patterns, and highlight strengths and weaknesses in current research in the field, as reported in management and social sciences.         | Focus on articles with management-type implications. Takes an innovation-as-a-process perspective.                                                                                                                        | SLR based on the guidelines of Davis<sup>19</sup> and Short<sup>20</sup>                                                                       | No. Crowdsourcing is examined as an organizational phenomenon from the perspective of a management theory lens. |