View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Lancaster E-Prints

Social Media through the Requirements Lens: A Case Study of Google Maps

Georgi M. Kanchev
Lancaster University
Lancaster, LA1 4WA, UK
g.kanchev@lancaster.ac.uk

Amit K. Chopra
Lancaster University
Lancaster, LA1 4WA, UK
a.chopra1@lancaster.ac.uk

***Abstract*—Social media serves as an extensive repository of user interaction related to software applications. Users discuss application features and express their sentiments about them in both qualitative (usually in natural language) and quantitative ways (for example, via votes). Further, many social media applications support explicit social networks of users and measures such as user reputation. Naturally, content on social media has the potential to inform requirements engineering. However, models of requirements and associated tools that enable software engineers to make sense of this information are currently lacking.**

In this paper, we present a preliminary study of interaction among users about Google Maps on the forum Reddit. We highlight important artifacts relevant to requirements in these interactions. We discuss goal modeling as an archetypal requirements modeling approach and use that as a basis for enhancing requirements modeling with notions that capture user interaction.

***Index Terms*—social media, user feedback, end-user involvement, requirements modeling, interaction**

I. Introduction

The importance of contact with end users and taking into account their feedback is well established as a success factor in a software development project [14], [15]. Traditional elicitation techniques fail to take advantage of rich user feedback on social media. In this paper, we perform an exploratory analysis on the different aspects of social media and how they can be leveraged to benefit requirements engineering. We are especially interested in systematic approaches of organizing requirements-related information found in social media in a structured way. The *requirements lens* is a metaphor for such approaches.

Research has identified the missing link between software development teams and end users [4], [20]. User feedback about software applications in social media is currently being either ignored or taken into account by developers in ad-hoc ways. The reasons for this state of affairs are varied.

- Interaction in social media is informal, voluminous, often meandering, and varied. Current approaches do not systematically support *making sense* of this interaction in terms of their impact on requirements. Such approaches include techniques and tools to systematically gather, organize, visualize, and reason about such information from a requirements perspective.
- Large developers describe the social media feedback as "useless noise." This may be a consequence of the above point: they do not know how to extract useful information from it [2].

Content in social media is usually in the form of unstructured natural language enhanced with various quantitative attributes, gained from user interaction in an online collaborative setting. Organized, visual end-user feedback will create an interesting, high-level, abstract view that will potentially aid various processes in requirements engineering.

In this paper we present the following contributions.

- We report a preliminary analysis of feedback from social media and explore the information relevant to requirements engineers. For our case study, we choose user interactions about Google Maps on Reddit.com, which is a Web-based forum (Section III). We explore how taking into account feedback on a software product can be used to inform and improve the requirements for its future releases.
- We show examples of requirements-related artifacts that may be extracted from the forum. These include requirements, solutions, expressions of sentiment about those, observations, and so on (Section III).
- We demonstrate the need to augment traditional requirements modeling approaches with new abstractions to capture some of these artifacts. For concreteness, we discuss goal modeling (Section IV).
- Finally, we discuss directions for future research concerning end-user involvement in software projects (Section VI).

II. Case Study: Google Maps and Reddit.com

A. Target Application: Google Maps

Google Maps is a Web application developed, provided, and maintained by Google Inc. It provides information about geographical regions and sites worldwide with a focus on road and traffic systems. In order to achieve this, Google Maps combines aerial satellite imagery with conventional road maps. We list some of the services offered by Google Maps below:

Route Planner enables users to plan a route and receive visual and voice directions on how to make their trip from one location to another.

Driving Directions combines *Route Planner* and information about the current location of the host device to provide real time instructions to the user. It uses a "next action" basis, for example "Turn left in 50 meters and you will have reached your destination." as synthesized voice message combined with on-screen instructions.

Voice Command allows users to execute commands on Google Maps verbally, with minimal physical interaction with the host device. The user is able to navigate through most of Google Maps features by voice and receive synthetic voice as response, thus enabling the driver to keep his eyes on the road at all times.

B. Social Media Outlet: Reddit.com

Reddit is a Web forum. It uses crowd-sourcing techniques to distribute the work of content creation, moderation, and filtering to its community. People may submit their own content and vote on other user's submissions. Content with more positive votes is ranked higher, and therefore shown to a larger portion of the community. Registered users on Reddit are called *redditors*. Reddit has a complex hierarchical structure of entities, shown in Figure 1. The community of redditors is divided into subreddits, where each subreddit represents a general topic of discussion, such as programming, music, or football. Redditors can submit content to a subreddit in the form of submission. A submission holds information in the form of natural language, image or hyperlink. Redditors can post comments on a submission and other redditors can post comments on existing comments. All comments are stored and displayed in a hierarchical, waterfall-type structure, as seen in Figures 2 and 3, so the progress of any discussion can be easily observed. Other relevant features of Reddit are:

Vote system Users can show their approval or disapproval of a submission or a comment by giving it an up or down vote, respectively. Votes are combined with other data to create the *karma* score (measured in points) of each submission and comment. Karma score is used to order posts within a subreddit and comments within a post, so that the most relevant to the topic and popular information is showed to the largest amount of people.

User reputation. Each redditor has karma score to serve as reputation. User karma is affected by the amount of community approval for comments and submissions the user has created.

C. Data Source

We are currently studying a Reddit submission where redditors are discussing missing features or features that are not well addressed in Google Maps. The submission was created on September 28, 2014. As of April 1, 2015, it has over 5500 upvotes and 1166 comments. We selected this particular submission, because manual analysis showed that it was a rich source of examples related to requirements.

III. Findings on the Forum

We present two forum discussions, found on Reddit, shown in Figures 2 and 3. We added annotations to the figures (A, B, C, and so on) in order to put emphasis on key elements, such as comments that shape the discussion on the forum. Below we analyze the discussions from a requirements perspective.

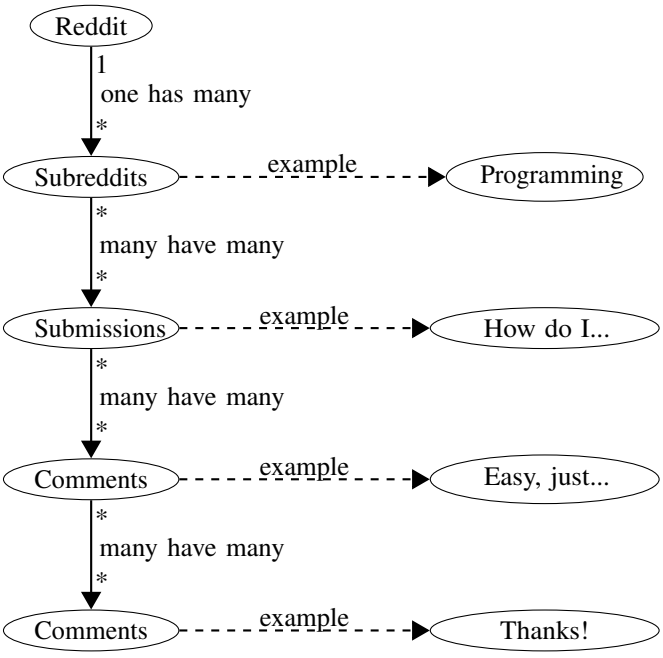

Fig. 1. A hierarchical view on how Reddit artifacts relate to one another

A. Overview

Figure 2 shows that people are having difficulty with finding ways to make the navigation in Google Maps repeat its last instruction to the user. Deeper in the discussion that requirement is broken down into two smaller, easier to achieve requirements: tap the microphone-shaped button and then input a relative voice command. The participants in this discussion don't comment on having to tap the button, but they make an *observation* regarding relative commands—relative commands are unintuitive. This observation has a highly negative influence on the system and has strong support in the community, as illustrated by the comments expressing sentiment towards it.

Figure 3 shows that the community would prefer having an alternative to tapping the microphone for activation of *listening* mode (state in which the device awaits a voice command). A redditor proposes voice activation of *listening* mode by saying a key phrase as a possible solution. Another redditor mentions how another application developed by Google Inc. solved the problem (OK Google), and also states an observation on the solution, that the device must always be charging when the solution is used. Further in the discussion another observation is made about voice activation of *listening* mode—screen must be on.

B. Key Findings.

During the analysis we found several recurring artifacts and made note of some aspects of social media that can be of use to requirements engineering.

Requirements. In Figure 2.A, we found a usability problem. The problem is then broken down as the discussion progresses and the reason for it is identified through end-user experience and observation of the system. Requirements

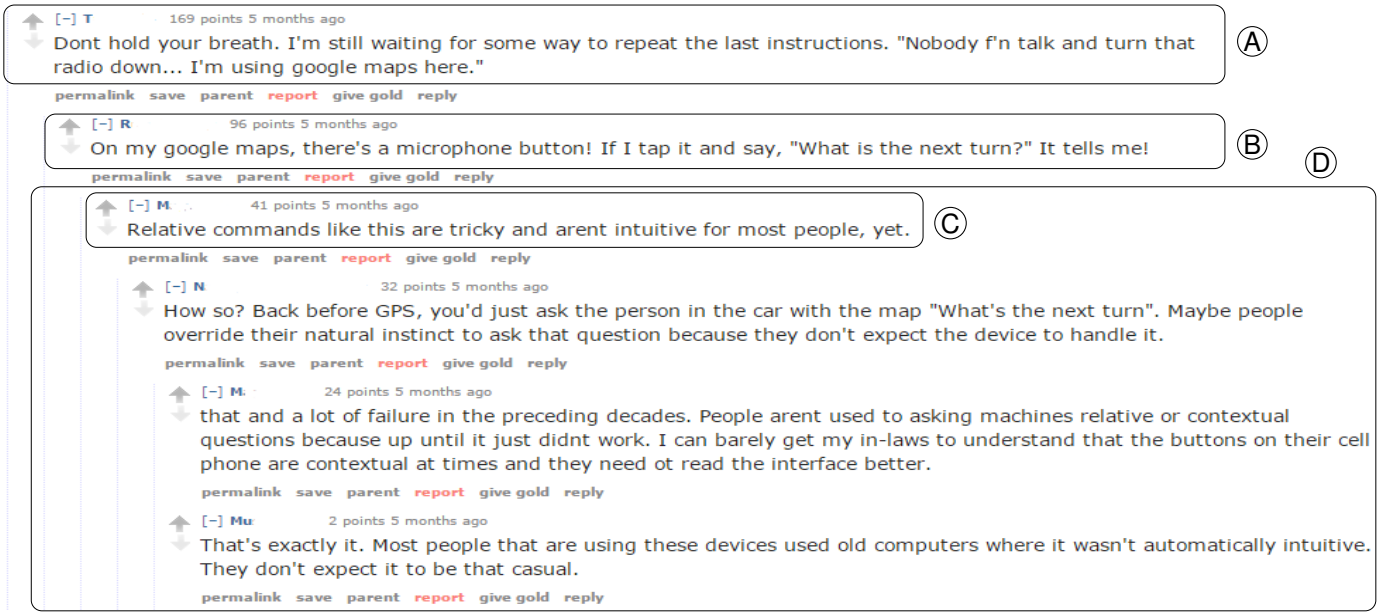

Fig. 2. A discussion about Google Maps on Reddit. Participants identify and debate on the unintitiveness of relative voice commands in the application.

on social media are not expressed in a clear-cut sense, unlike in traditional requirements specification documents found in software projects. Often requirements need to be derived from expressions of ideas, needs or desires.

Observations. The preliminary analysis showed that end-users express tacit knowledge in online forums in the form of observations: system behavior, requirements interdependencies, and others. They explicitly state information that may not be apparent to developers. Such information is referred to as unknowns [5], [19]. Discovery of unknowns is challenging, as it is highly exploratory. End-user feedback is an excellent candidate source of unknowns because end users have know-how and experience with interacting with the system. Examples of observations can be seen in Figures 2.C, 3.B, and 3.D.

Lack of awareness of existing solutions. The comments in Figure 2.A and Figure 3.A both show examples of a requirement statement from redditors. They both briefly describe and request functionality. In the next comments, found in Figure 2.B and Figure 3.B, other redditors inform them that the required functionality is already implemented in the application. The unawareness of existing functionality can used as a surrogate for identification of user interface problems. It shows that end-users are unable to find out how to use specific functionality that is readily available to them.

Expressions of sentiment in comments. End-users express sentiment towards solutions or requirements in online forums. Figure 2.D shows a string of comments expressing sentiment and support towards the non-intuitiveness of relative commands. Negative sentiment towards a solution can be treated as a sign that the solution is not good enough and positive sentiment towards a requirement can be seen as indication that the requirement must be considered with high priority. Sentiment is a direct representation of end-user satisfaction (towards solutions) and desire (towards requirements).

Community support. Social media uses collaborative filtering to filter content. Voting on Reddit can be used to measure the support of the community towards a requirements artifact using quantitative metrics (upvotes and karma score). For example the comment in Figure 2.A, asking for a way to repeat the last instruction from navigation, has 169 points (karma score). In contrast, the comment in Figure 3.A, asking for voice activation of *listening* mode, has 2 points. Repeating the last instruction receives larger support from the community, and can potentially be treated with priority.

IV. User Feedback and Goal Modeling

During the analysis we labeled sections of data that contain abstract artifacts. Since feedback is communication initiated by end-users and aimed at developers, we looked at goal modeling abstractions to serve as our entities. Goal (or intentional) modeling is intended to introduce a low-level abstraction over system requirements that is easy to communicate to stakeholders, without committing to a technological solution [21].

In Figure 4 we present a simple goal model of the domain knowledge found by studying Figures 2 and 3. In Figure 4 ellipses and clouds represent goals and soft goals respectively. Arrow labeled "-" and "- -" represents negative and highly negative contribution, respectively. The motivation behind creating the model is to help identify gaps in current goal modeling that prevent accurate representation of the information found on social media. We aim to illustrate how current

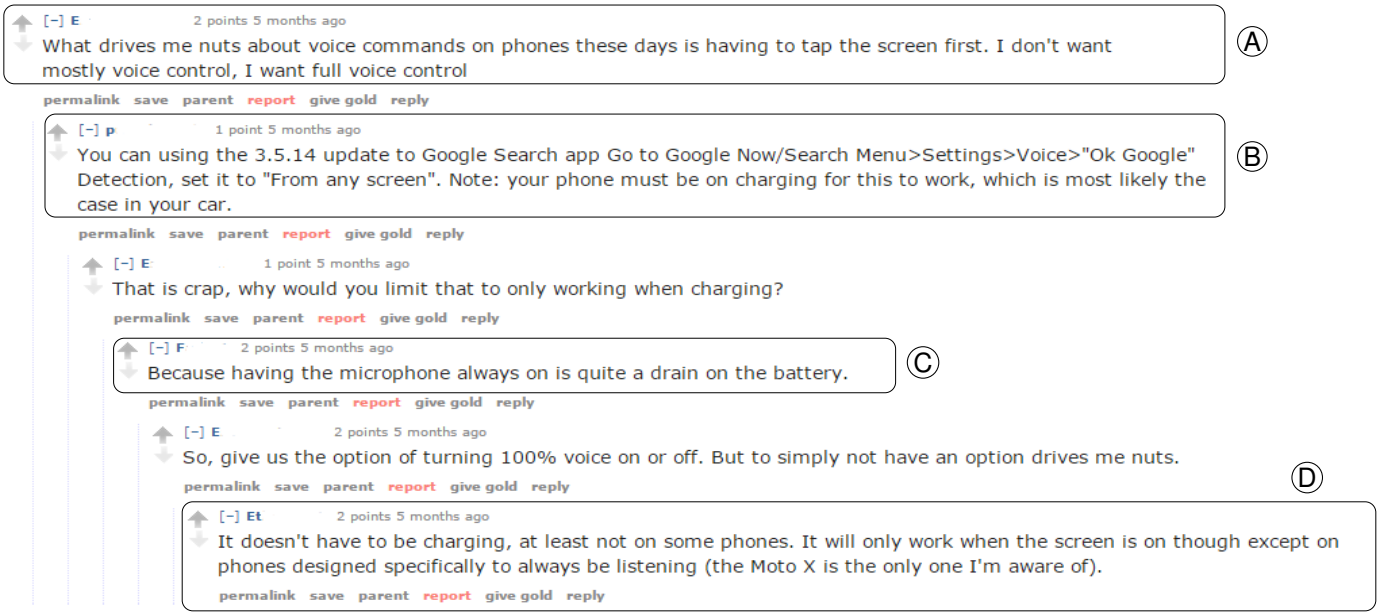

Fig. 3. A discussion about Google Maps from Reddit. Participants are discussing the plausibility of having voice control enabled at all times

requirements modeling techniques are unable to capture specific information that is available in social media. We use goal modeling as a concrete example. We present our findings below:

A. *Quantitative information loss.*

The design of current requirements modeling techniques, including goal modeling, doesn't allow metadata to be freely added to the representation. This leads to loss of information available in the source. If represented the information can be useful to requirements elicitation, prioritization and others.

Community support and sentiment. There is no visual representation of community support for an artifact in the model. Also, community support doesn't *propagate* through the relations of the target artifact. In Figure 2.A we see a usability issue, because users are unable to find out how to repeat the last navigational instruction from Google Maps. In Figure 2.C the fault for the issue is placed on an the observation that relative commands are not intuitive. In Figure 2.D, there are comments showing support (positive and negative) for the observation, but the goal model cannot represent that information. The link between the usability issue of repeating the last instruction and the unintuitiveness of relative commands is not made evident in the model.

User reputation. In Section II, the personal karma score of each user is mentioned. It can be used as a quantitative metric for further filtering and elicitation of entities. User reputation is also based on collaborative filtering. Users with high reputation are users who have provided valuable to the community input in previous discussions. It is a new means of adding context to individuals in an otherwise homogeneous crowd of end-users.

Controversial entities. Entities causing high controversy, such as large discussions or fluctuating approval and sentiment, can be detected and brought forward in the model. An interpretation of such information can detect usability issues for certain demographics. The comments in Figure 2.D indicate agreement that the age of users is a factor when considering the intuitiveness of relative commands. The amount of social media content (comments, votes) generated in regards to a specific artifact can be used as a metric for controversy.

B. *The bigger picture.*

Combining discussions on the same topic, such as Google Maps, into the same model gives us a much richer understanding of the problem domain. The model in Figure 4 is the result of combining domain knowledge gained from Figures 2 and 3. The combined model gives a better picture of the system. It represents a larger portion of the domain and shows how requirement entities are intertwined on a larger scale. Relationships between the entities are well represented using goal modeling, but we noticed that interaction on the forum is structured as argumentation. Further research may prove argumentation-related notions to be more informative for modeling of interdependencies of entities from multiple sources.

V. Literature

In order to give a better understanding of who the end user is in different software projects, Lubras et al. [12] introduce the terms market-driven and customer-specific development. In customer-specific development the project has a *specific*, contractual customer (who will be the end user), who can be addressed directly. In the case of market-driven projects there is more than one *potential* end user.

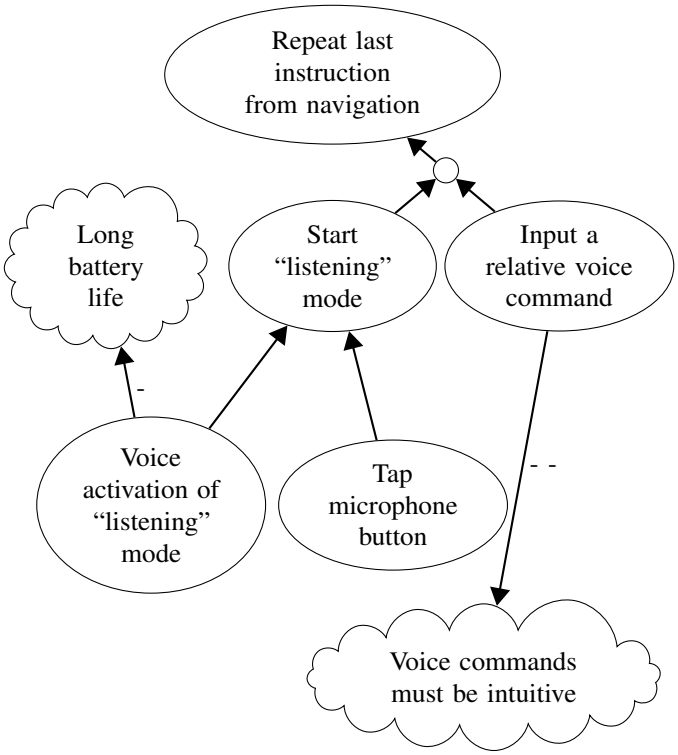

Fig. 4. A model representation using goal modeling techniques of the domain knowledge gained regarding Google Maps from Figures 2 and 3

From then Lubras et al. proceed to identify the different challenges that affect each type of development. For the purposes of this paper we will focus on market-driven development. The participants in the survey reported issues with forming a mental image of who the customer is, producing a product that is not responsive enough to customer needs and prioritizing requirements. The absence of a commissioned customer renders standard requirements elicitation and validation methods inapplicable. Companies are reported to have little to none customer or user involvement and requirements are self-invented [7], [8]. There is need for balance between developer and user elicited requirements [18]. With time, market change, and company evolution the project might lose up-to-date domain knowledge that regular contact with end-users provides.

Modern elicitation techniques are better at extracting an abundant set of requirements. This introduces new problems, such as filtering and prioritization [9], [20]. Developers gather requirements into potentially huge repositories in natural language. The lack of context associated with those requirements creates difficulties for developers in interpreting and prioritizing them. Studies call for more structured (context-aware) input of requirements-related data [17].

User feedback is a candidate for context-rich source of requirements [16]. Maalej et al. [13] inform the types of user feedback. One, *pull* if the feedback is pulled from the user; two, *push* if the feedback is pushed by the user; three, *explicit* if the user has intention to provide the feedback; four, *implicit* if the user unintentionally provides the feedback, for example background usage statistics. To this classification we add *internal* (if the feedback is collected within the system, that is internal bug report signal), *external* (if the feedback is gathered from a third party system, for example social media), *qualitative* (if the feedback gathered is of qualitative nature—natural language, pictures, video, etc.) and *quantitative* (if the feedback is of quantitative nature, such as usage statistics, numerical rating system). In this paper we consider pushed, explicit, external, qualitative (enhanced with various quantitative attributes) feedback in social media.

Ali et al. [1] propose a framework for adding context to goal modeling and introduce contextual goal models. As context they take qualitative specifications of the surroundings of a stakeholder and use it to extend the Tropos goal model. In a much similar way we are trying to enrich requirements modeling, but with quantitative metrics gathered through collective user interaction by a crowd of users.

Various social media techniques have been applied in research for the purposes of requirements engineering. Lim et al. proposed StakerNet [11] as a crowdsourcing solution to stakeholder identification. They create a social network of stakeholders involved in the software project by having each of them to nominate others. Later that idea was expanded into StakeRare [10], where stakeholders in the network were also allowed to propose requirements and prioritize them using collaborative filtering. These techniques are beneficial to customer-specific projects, but their application is limited in market-driven development. StakeNet and StakeRare take advantage of the ability to directly address a commissioned customer. Greenwood et al. introduce UDesignIt [6], where they draw text from social media and apply natural language processing to generate a prioritized feature model.

A formalization of argumentation is proposed by Chopra and Singh in Colaba [3], where they propose tool-based assistance for collaborative design of cross-organizational processes. Argumentation is presented as a process of six steps that can be repeated recursively until agreement is reached. The steps described can be observed in a forum discussion. A similar notation can be used for specification of argumentation in social media.

VI. Discussion

This paper presents the results of a preliminary analysis of feedback in social media. We carried out a case study about Google Maps on Reddit.com, which is a Web forum. We set out to highlight and bring forth important artifacts relevant to requirements engineering in the interactions between users on forums. We also critically evaluate the effectiveness of goal modeling techniques to capture the information contained within feedback in social media with the purpose of enhancing requirements modeling with notions that capture user interactions.

In order to achieve that we manually explored Reddit.com in search for discussions between users regarding Google Maps. In those discussions we discovered several key findings regarding the information of value to requirements engineering

available in social media. Our key findings include *requirements*, *observations made by end-users* (explicit statements of tacit knowledge), *lack of awareness of existing solutions*, *expressions of sentiment towards previously discovered artifacts* (users tend to express their agreement that a requirement is necessary or that a solution is not good enough), and *community support* (well defined, pertinent artifacts are recognized by the community using the collaborative voting system).

We extracted two discussions from the forum and built a goal model of Google Maps using the domain knowledge gained solely from those discussions. Comparing the information captured in the resulting model to the amount of information available in the source, we concluded that requirements modeling can benefit from enhancements that capture user interactions by taking advantage of several key elements of social media that involve quantitative metrics. Examples of that are *community support and sentiment*, *user reputation*, and *controversy of entities*. In addition to that we also discovered that the interaction between users on the forum is structured as argumentation and that combining multiple sources of domain knowledge in the same model enriches our understanding of the domain.

Our findings can be used to motivate research in designing a requirements model. The notions supported by the model can capture user interaction as context to traditional requirements artifacts, such as goals or requirements. Social media is a highly interactive environment, where users are encouraged to evaluate, rate, and vote on each others contributions to the community. Such a model would require abstractions that can be informed by our findings.

Argumentation emerges as a strong candidate source of abstractions that can capture the qualitative requirements-related information in social media, and can be expanded to accommodate the quantitative metrics mentioned above. Further research is needed on the applicability of argumentation towards understanding user feedback in social media.

The interactivity of social media makes the content within it dynamic. As a result, using it as a source of information can have some interesting implications on the value of the information we extract. New content emerges in social media at a fast paste, so how long after data is created it is still relevant? When is it most relevant? How do the dynamics of social media affect the design of a requirements model intended to capture interactions between the users of that media?

Analyzing the vast quantity of information in social media manually proved to be challenging and error-prone. An interesting area for future research is tool support for assistance and automation of the analysis of feedback in social media. Support can be provided for the finding, filtering, and extraction of information, as well as reducing the amount of manual labor needed for its analysis by applying natural language processing techniques.

REFERENCES

[1] Raian Ali, Fabiano Dalpiaz, and Paolo Giorgini. A goal-based framework for contextual requirements modeling and analysis. *Requirements Engineering*, 15(4):439–458, 2010.

[2] Dejana Bajic and Kelly Lyons. Leveraging social media to gather user feedback for software development. In *Proceedings of the 2nd international workshop on Web 2.0 for software engineering*, pages 1–6. ACM, 2011.

[3] Amit K Chopra and Munindar P Singh. Colaba: Collaborative design of cross-organizational processes. In *Requirements Engineering for Systems, Services and Systems-of-Systems (RESS), 2011 Workshop on*, pages 36–43. IEEE, 2011.

[4] Bill Curtis, Herb Krasner, and Neil Iscoe. A field study of the software design process for large systems. *Communications of the ACM*, 31(11):1268–1287, 1988.

[5] Ricardo Gacitua Mark Rouncefield Peter Sawyer Leonid Kof L. Ma P. Piwek et al. Gervasi, Vincenzo. Unpacking tacit knowledge for requirements engineering. In *Managing requirements knowledge*, pages 23–47. Springer, 2013.

[6] Phil Greenwood, Awais Rashid, and James Walkerdine. Udesignit: Towards social media for community-driven design. In *Software Engineering (ICSE), 2012 34th International Conference on*, pages 1321–1324. IEEE, 2012.

[7] Sami Jantunen, Donald C Gause, and Ragnar Wessman. Making sense of product requirements. In *Requirements Engineering Conference (RE), 2010 18th IEEE International*, pages 89–92. IEEE, 2010.

[8] Erik Kamsties, Klaus Hörmann, and Maud Schlich. Requirements engineering in small and medium enterprises. *Requirements engineering*, 3(2):84–90, 1998.

[9] Lena Karlsson, Åsa G Dahlstedt, Björn Regnell, Johan Natt och Dag, and Anne Persson. Requirements engineering challenges in market-driven software development–An interview study with practitioners. *Information and Software technology*, 49(6):588–604, 2007.

[10] Soo Ling Lim and Anthony Finkelstein. Stakerare: Using social networks and collaborative filtering for large-scale requirements elicitation. *Software Engineering, IEEE Transactions on*, 38(3):707–735, 2012.

[11] Soo Ling Lim, Daniele Quercia, and Anthony Finkelstein. Stakenet: Using social networks to analyse the stakeholders of large-scale software projects. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1*, pages 295–304. ACM, 2010.

[12] Mitch Lubars, Colin Potts, and Charles Richter. A review of the state of the practice in requirements modeling. In *Requirements Engineering, Proceedings of IEEE International Symposium on*, pages 2–14. IEEE, 1993.

[13] Walid Maalej, Hans-Jörg Happel, and Asarnusch Rashid. When users become collaborators: Towards continuous and context-aware user input. In *Proceedings of the 24th ACM SIGPLAN conference companion on Object oriented programming systems languages and applications*, pages 981–990. ACM, 2009.

[14] Laurie McLeod and Stephen G MacDonell. Factors that affect software systems development project outcomes: A survey of research. *ACM Computing Surveys (CSUR)*, 43(4):24, 2011.

[15] Mohd Hairul Nizam Nasir and Shamsul Sahibuddin. Critical success factors for software projects: A comparative study. *Scientific research and essays*, 6(10):2174–2186, 2011.

[16] Dennis Pagano and Walid Maalej. User feedback in the appstore: An empirical study. In *Requirements Engineering Conference (RE), 2013 21st IEEE International*, pages 125–134. IEEE, 2013.

[17] Björn Regnell and Sjaak Brinkkemper. Market-driven requirements engineering for software products. In *Engineering and managing software requirements*, pages 287–308. Springer, 2005.

[18] Pete Sawyer, Ian Sommerville, and Gerald Kotonya. Improving market-driven re processes. In *VTT SYMPOSIUM*, volume 195, pages 222–236. VTT; 1999, 1999.

[19] Alistair Sutcliffe and Pete Sawyer. Requirements elicitation: Towards the unknown unknowns. In *Requirements Engineering Conference (RE), 2013 21st IEEE International*, pages 92–104. IEEE, 2013.

[20] Krzysztof Wnuk, Björn Regnell, and Brian Berenbach. Scaling up requirements engineering–Exploring the challenges of increasing size and complexity in market-driven software development. In *Requirements Engineering: Foundation for Software Quality*, pages 54–59. Springer, 2011.

[21] Eric Yu and John Mylopoulos. Why goal-oriented requirements engineering. In *Proceedings of the 4th International Workshop on Requirements Engineering: Foundations of Software Quality*, page 15–22.