Poster: Exploring Relationship-aware Dynamic Message Screening for Mobile Messengers

Seungchul Lee
School of Computing, KAIST
Daejeon, Republic of Korea
seungchul@nclab.kaist.ac.kr

Chulhong Min
Nokia Bell Labs
Cambridge, UK
chulhong.min@nokia-bell-labs.com

Saumay Pushp
School of Computing, KAIST
Daejeon, Republic of Korea
saumay@nclab.kaist.ac.kr

Junehwa Song
School of Computing, KAIST
Daejeon, Republic of Korea
junesong@nclab.kaist.ac.kr

UbiComp/ISWC'18 Adjunct, October 8–12, 2018, Singapore, Singapore
ACM 978-1-4503-5966-5/18/10.
https://doi.org/10.1145/3267305.3267673

Abstract

Mobile instant messengers lack the *social appropriateness* of conversation, which incurs embarrassing situations when an unwanted message is unexpectedly exposed to people nearby. To avoid such situations, we develop a relationship-aware mobile messenger that takes into account the relationship of a receiver to a sender and people nearby. Based on the in-situ relationship, it selectively shows or hides a content of incoming messages on the notification pop-up. We develop a messenger prototype and show its usefulness via a deployment study.

Author Keywords

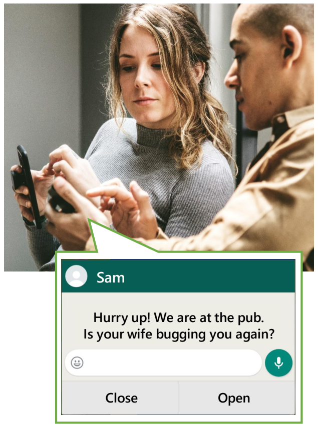
Social appropriateness; Mobile instant messenger; Dynamic message screening; Smartphone; Privacy

ACM Classification Keywords

H.1.2 [Models and principles]: User/Machine Systems;
H.4.m [Information systems applications]: Miscellaneous

Introduction

Mobile instant messengers (MIMs) are not *socially* secure. Despite their convenience of remote conversations, they lack *social appropriateness* in real-life situations, which we naturally consider in offline conversations. Since the sender is unaware of the receiver's situation, an incoming message could embarrass the receiver when a unwanted message is



Hurry up! We are at the pub.
Is your wife bugging you again?

Figure 1: Unwanted message exposure causes.

exposed to people nearby unexpectedly.

While the smartphone is considered as a very personal device, its screen can be easily seen by others, increasing the probability of unexpected message exposure. We often share our smartphone screen with others to view pictures and videos together [3], and leave the phone open in shared spaces [1]. Figure 1 shows an example of inevitable message exposure. While a man was showing his photos to his wife, he accidentally received an embarrassing message from his friend. Current MIMs provide a "turn-off-notifications" option to avoid such situations, but it highly degrades the user experience. Users should tap multiple times to check a message every time or change the option whenever they share their phones.

We develop a *relationship-aware MIM* that takes into account the social relationship of a receiver to a sender and people nearby. Users specify screening rules of incoming messages based on the relationship. The messenger dynamically screens a message content on the notification pop-up depending on the in-situ relationship. For example, in a socially comfortable situation, it shows all of the message content to maximize user convenience. It hides sensitive information in the content when the user is socially uncomfortable.

In our previous work [4], we investigated users' perception on unexpected message exposure to understand the factors in an embarrassing situation. In this paper, we do take a step forward. We first summarize our findings on factors resulting from embarrassing situations and then introduce our prototype implementation of a *relationship-aware* mobile messenger. We also evaluated our prototype with a small-scale deployment study.

Perception on Unexpected Message Exposure

In our previous work [4], we conducted a user study to uncover the seriousness of the unwanted message exposure and potential factors in embarrassing situations. We took the experience sampling method (ESM) to collect participants' lively perception on message arrival. We recruited 14 participants during a two-week period. They were requested to report their surrounding situation and answer the degree of embarrassment for incoming messages on a 5-point Likert scale. From a total of 961 responses, 29% of incoming messages were reported that participants would feel embarrassed if they were exposed, showing daily prevalence of socially inappropriate messages. We also found that the risk of privacy exposure on mobile messaging already pervades in real-life; 39% of the messages were delivered when there were other people nearby and 6% were actually seen by them, respectively. Most important, we explored several factors that could affect the degree of embarrassments and we figured out that the core factors are *a message sender* and *a potential observer*.

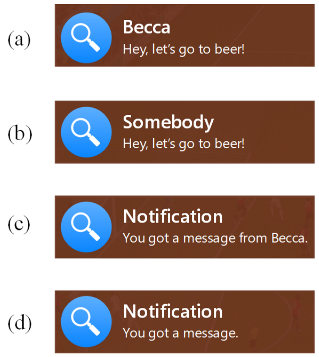
Small-scale Prototype Deployment

Prototype implementation

For the proof-of-concept, we develop a "relationship-aware" messenger prototype on Android. On message arrival, it dynamically screens message preview considering a user's relationships with a message sender and a potential observer. Users can write their own rules describing embarrassing situations and configuring dynamic screening. The rule format is:

$$ScreenOption \text{ with } GroupNearby \text{ from } GroupSender$$

The messenger provides four types of screening, *'show-all'*, *'hide-sender'*, *'hide-message'*, and *'hide-all'* (see Figure 2). When a new message arrives, it searches for matching



Becca
Hey, let's go to beer!

Somebody
Hey, let's go to beer!

Notification
You got a message from Becca.

Notification
You got a message.

Figure 2: Four types of message screening:
(a) *show-all*, (b) *hide-sender*,
(c) *hide-message*, (d) *hide-all*.

rules. If a message is sent by *GroupSender* and *GroupNearby* is nearby a user, the message preview is displayed with *ScreenOption*.

To identify the existence of a nearby person in real time, we use continuous Bluetooth scanning, which is widely adopted for peer detection [2]. We select 3 minutes as an interval, compromising energy consumption and encounter detection latency. To capture the reception event of mobile messages, we use an accessibility service on Android. Our prototype system selectively captures the reception event of KakaoTalk messages, extracts the sender information, and displays a preview via customized pop-up window.

Small-scale deployment study

We recruited 4 participants (aged 26-33) who frequently used KakaoTalk for instant messaging and deployed the prototype for two weeks. Since people usually do not have their Bluetooth turned on, we asked each participant to choose up to 10 people whom they are likely to meet within a week. We also asked them to install our application which keeps Bluetooth always activated. Each participant was provided with KRW 50000 (44 USD) and each acquaintance of the participant with KRW 5000 (4.4 USD) as well. After the deployment, we have 1-hour long semi-structured interview to obtain their experiences.

Findings

Overall, all participants were satisfied with the prototype. They had concerns or experiences on message exposure, e.g., leaving a phone on the desk. However, during the study, they were happy with securing messages. For example, P4 successfully hid messages sent by her boyfriend from her mother, who always asked about the message.

Rule & group configuration: During the deployment, they configured 3-6 groups and 3-4 rules. P1 added a rule to hide messages from her close friends when she was with parents. The other participants had a common rule to screen messages from their significant others. Interestingly, they had different opinions on the number of rules and groups. P1 stated, *"I firstly configured two rules. But now I feel I need more rules."* She actually added a new rule afterwards. However, P3 and P4 mentioned that three rules were enough.

Rule specification & interface: Though the participants expressed satisfactions with the system, they sometimes felt inconvenient in manual specification for every group and rule. Since it was impossible to create a new rule right before the exposure, they had to predict potential observers for upcoming days. We may utilize machine learning techniques to infer such situations and generate a new rule at the moment. We leave it future work.

Satisfaction on screening methods: Interestingly, *hide-message* and *hide-sender* were rarely used. During the deployment, P1 and P3 changed some rules with *hide-sender* to *hide-all*. They stated that it was because people nearby sometimes inferred the message sender by seeing only the message content. We then asked about content analysis for selective filtering of messages. P1 answered, *"Still, I would prefer 'hide-all'. It seems also weird if my phone understands the message."* However, P2 and P4 were satisfied with the screening methods.

Detection of nearby people: Out prototype discovers nearby people whose device should turn on Bluetooth. We compensate the acquaintances of the participants with incentive during the study. However, all participants commonly had a concern about asking their acquaintances of installing an application. They told us, *"I liked it, but is it possible to make it standalone?"* To complement automated detection, we can leverage just-in-time privacy provisioning using subtle user inputs [5].

Privacy consciousness from message sender: We also observed the participants' privacy concern when they send a message. P1 said, *"While using this service, I realize that my messages can be also seen by others."* She suggested an option to hide message pop-ups on the receiver's phone before sending a message.

Discussion and Future Work

Other Factors Which Determines Privacy Concern: There would be other factors affecting privacy concerns than factors we investigate, such as intimacy with acquaintances, user's temporary emotional status, and personality of an observer. During the ESM, a male participant shared his opinion about different privacy concern depending on relationship variation with his girlfriend. A female participant also mentioned that her daily feeling affects the level of privacy concern. These factors can be handled as well, by incorporating advanced technology such as affective computing. We leave this to future work.

Content Analysis: It would be also possible to analyze type or content of the message to better specify private situations. For example, when a user is with his parent, the system automatically hide messages from his friends containing slang words, or change them into other keywords. However, there are two challenges that made us not consider this; its technical difficulty as well as users' privacy concern on the analysis. We leave this to future work.

Acknowledgements

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT (NRF-2017M3C4A7065963).

REFERENCES

1. Anind K Dey, Katarzyna Wac, Denzil Ferreira, Kevin Tassini, Jin-Hyuk Hong, and Julian Ramos. 2011. Getting closer: an empirical investigation of the proximity of user to their smart phones. In *Proceedings of the 13th international conference on Ubiquitous computing*. ACM, 163–172.

2. Youngki Lee, Seungwoo Kang, Chulhong Min, Younghyun Ju, Inseok Hwang, and Junehwa Song. 2016. CoMon+: A cooperative context monitoring system for multi-device personal sensing environments. *IEEE Transactions on Mobile Computing* 15, 8 (2016), 1908–1924.

3. Yunxin Liu, Ahmad Rahmati, Yuanhe Huang, Hyukjae Jang, Lin Zhong, Yongguang Zhang, and Shensheng Zhang. 2009. xShare: supporting impromptu sharing of mobile phones. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*. ACM, 15–28.

4. Chulhong Min, Saumay Pushp, Seungchul Lee, Inseok Hwang, Youngki Lee, Seungwoo Kang, and Junehwa Song. 2014. Uncovering embarrassing moments in in-situ exposure of incoming mobile messages. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. ACM, 1045–1054.

5. Saumay Pushp, Yunxin Liu, Mengwei Xu, Changyoung Koh, and Junehwa Song. 2018. PrivacyShield: A Mobile System for Supporting Subtle Just-in-time Privacy Provisioning through Off-Screen-based Touch Gestures. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 2 (2018), 76.