Exploring Design Parameters for RSVP Reading of Mobile Notifications

HAO-PING (HANK) LEE, Carnegie Mellon University, United States
QIUSHI ZHOU, Aarhus University, Denmark
VASSILIS KOSTAKOS, University of Melbourne, Australia
BENJAMIN TAG, Monash University, Australia
TILMAN DINGLER, TU Delft, Netherlands

Rapid Serial Visual Presentation (RSVP) has been both praised and scorned as an effective reading technique. While it enables text rendering on small screens, its dynamic text display tends to demand heightened attention from users. Previous work has implemented RSVP reading interfaces on a range of devices, such as watches, glasses, and phones, but none have gained widespread adoption. Considering its ability to display information on devices with small screen real-estate effectively, we explored its applicability to reading smartphone notifications. Following a user-centered design process, we first elicited the main design parameters through a focus group. Informed by these insights, we implemented our RSVP notification prototype, *SpeedNotification*. We tested SpeedNotification's specific design parameters in a lab study and evaluated RSVP notifications more broadly in a field study. We present the essential functionalities and challenges for creating a feasible RSVP user experience for mobile notifications and discuss associated issues and opportunities.

CCS Concepts: • **Human-centered computing → Empirical studies in HCI**; **Smartphones**.

Additional Key Words and Phrases: Smartphones, RSVP, Mobile Notifications, Reading

ACM Reference Format:
Hao-Ping (Hank) Lee, Qiushi Zhou, Vassilis Kostakos, Benjamin Tag, and Tilman Dingler. 2018. Exploring Design Parameters for RSVP Reading of Mobile Notifications. In . ACM, New York, NY, USA, 22 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Reading is a complex cognitive skill that — through schooling — is mainly acquired at a very young age. This is where formal reading education stops for most adult readers, and further advancing their reading skills happens mostly through practice [7]. For skilled readers, in fact, "*reading feels so simple, effortless, and automatic that it is almost impossible to look at a word and not read it*" as Rayner *et al.* [33] put it.

In recent years, digital displays and their ability to adapt Rapid Serial Visual Presentation (RSVP) have challenged the traditional notion of reading. Rather than showing text in sentences and paragraphs, RSVP displays one word at a time in sequential order. Because of its benefits of reducing eye movements, RSVP was originally conceived as an experimental model for assessing the temporal characteristics of human attention. In the context of reading, RSVP has been studied for how well it facilitates text processing and comprehension [14, 16] and for its potential for supporting speed-reading [11]. In addition, RSVP allows text to be rendered within limited space, making it a favourable reading interface for small displays such as mobile phones and watches [10, 15]. Previous explorations of RSVP on mobile devices have shown that space is effectively traded for time when reading on small displays, i.e., RSVP maintains its advantages of presenting text in limited space while minimizing the reading workload of the user [9, 18]. While this seems to make it more suitable for reading short bursts of text, such as news feeds and notifications, the systematic research in this field is sparse.

In this paper, we focus on the use of RSVP for mobile notifications. Through a focus group, a controlled lab study, and a field study, we answer the two key research questions:

RQ1 How should RSVP be designed and presented on smartphones for notification reading to achieve better reading experiences?

RQ2 How would users perceive and use RSVP to read notifications on their phones in their daily lives?

To explore this practical context, we first assessed the design parameters of RSVP for notification reading on smartphones through a focus group. Based on the findings, we developed a mobile application, *SpeedNotification*, that renders incoming notifications as dynamic RSVP messages in the phone's notification bar. Using this application, we observe and assess how smartphone users use different RSVP presentation modalities in a field study by informing the design of an alternative notification drawer.

In the focus group, we identified two main parameters considered vital to RSVP notifications: *functionality* (i.e., the control and display of RSVP notifications) and *notification grouping* (i.e., assembling multiple notifications into a single RSVP reading flow). We examined both functionality and grouping in a controlled lab study. Subsequently, we deployed our SpeedNotification app on 23 users' smartphones to conduct a field study. Through interaction logs, we investigated how SpeedNotification was used and how RSVP notification reading was perceived.

Our results provide insights into design parameters for RSVP in the context of notification displays. Confirming previous works, participants preferred reading shorter texts using RSVP, providing a better reading experience and higher reading completion rate. In particular, notifications shorter than ten words were reported to be the most desirable to be presented in RSVP. Participants proactively adjusted the RSVP display speed; however, they tended to modify the global speed over time in the app settings rather than impromptu speed changes during reading. We further found that while grouping notifications by application allowed users to read notifications in batches, it also increased the interaction complexity and was less favored in overall usability. On the other hand, grouping notifications by notification title (i.e., as one thread) was reported to be handy and efficient, thus making them more usable.

In sum, the contribution of our research is three-fold:

- We present the first interactive application for RSVP mobile phone notifications.
- We assess and evaluate RSVP presentations for notification reading through a lab and a field study.
- We provide a detailed discussion of insights and design implications regarding the adaptability of RSVP for notification systems.

2 RELATED WORK

A fundamental way of consuming information on mobile devices remains to be through reading. Mobile screen sizes pose various challenges for effectively displaying text, which in turn affects reading. Previous work has investigated the effects of line numbers, sentence splits, reading speed, text length, as well as variations in background contrast on reading [37]. Display size, however, has been found not to significantly affect comprehension rates, whereas splitting sentences between pages is more detrimental to comprehension [12]: in cases where readers were shown only one or two lines of text simultaneously, only 9% comprehension deterioration was found compared to presenting 20 lines of text at a time. Shneiderman [39] also reported that the number of lines in which text is presented does generally not significantly affect reading speed, which is good news for mobile devices, on which much reading is done these days. Varying screen size has, however, been shown to influence the type of reading users do on different devices. While larger displays are rather used for opportunistic and intensive reading activities, small-screen displays tend to be used for casual reading [24]. With the proliferation of small-screen devices, such as phones and watches, alternative reading techniques have increasingly been subjects of investigation, such as Rapid Serial Visual Presentation.

2.1 Rapid Serial Visual Presentation

Rapid Serial Visual Presentation, a term introduced by Forster [14] is a prominent experimental model, which has been used to investigate the temporal characteristics of human attention. Stimuli are, thereby, presented in one focal point, eliminating the necessity to employ large eye movements (saccades). This technique has quickly been appropriated for studying attention [19]. Since it requires screens to present only a single stimulus at a time it introduces a trade-off between space and time. This trade-off is especially promising where screen space is scarce, as with mobile telephone displays, PDAs, and other wearable displays [5, 15, 23].

Different modes of RSVP have been proposed by Spence [8, 40, 41] to support the user in quickly grasping the contents of folders and catalogs of which the rapid sequence of visual targets in the same location at the same time is often described as *Keyhole RSVP*. Tse *et al.* [43] proposed RSVP as a dynamic slide-show presentation to browse videos based on keyframes in videos. Wittenburg *et al.* used a mode of RSVP for creating collages in the context of web shopping and an online bookstore [47, 49].

With the proliferation of small-screen displays and the fact that RSVP can operate within limited space, it has also quickly been appropriated for reading studies: Georgiev [15] investigated reading speeds on mobile devices while comparing it to other reading mediums, including PC screens and paper. Computer screens and paper allowed for generally higher reading speeds, but RSVP was still found to be a feasible alternative for reading on mobile phones. Dingler *et al.* [11] used RSVP to enforce higher reading speeds, observing a substantial learning effect after initial user alienation stemming from a loss of control over the reading flow. Wittenburg *et al.* [48] had investigated general RSVP control techniques in the context of web browsing, such as mouse rollover and radio button selection. Dingler *et et al.* more recently created a unified gesture set for controlling the RSVP reading flow on wearable devices, including touchscreen phones, watches, and glasses [10]. While de Brujin and Spence [8] examined the effects of different RSVP modes on eye movements, Dingler *et al.* [9] used eye-tracking as an implicit control method to pause and play the RSVP reading flow. Guo and Wang subsequently added physiological signal sensing to adjust RSVP settings to readers' sensed workload [18].

RSVP also comes with a range of challenges. The human visual system and its capabilities to process subsequent stimuli is a limiting factor for the rapid display of RSVP. Presentation rates and the visual similarity of subsequent stimuli influence the effectiveness and utility of RSVP [31]. Subsequent targets, when occurring in rapid successions of 180–450 ms, for example, result in a phenomenon described by Raymond *et al.* as *attentional blink* [32]. While the human visual system can recognize the presence of a target image in as little as 100 ms [19], the higher the presentation speed, the more likely it becomes that crucial content might be missed. Similarly, repetition blindness [22] has been shown to impede users' ability to detect distinct stimuli in rapid succession, a risk also fueled by high presentation speeds. Other than speed and attention-related limitations, RSVP has been investigated with a focus on its effects on reading comprehension. Studies focused on memory effects found that reading with RSVP can lead to a rather coarse recall of text [25] and, if given no control over the text flow, an inability to re-read parts of the text can impede comprehension [38]. Compared to other scrolling techniques, however, reading via RSVP was found to lead to similar comprehension levels by Hedin and Lindgren [20]. Due to challenges, such as its fleeting nature and relatively high cognitive demand [9], RSVP has so far failed to reach a broader customer-level adoption.

In our work, we investigate how RSVP reading can be applied to mobile notifications. While a number of previous studies have examined use cases for RSVP, such as reading on head-mounted displays [35], our study takes a user-centered design approach to designing RSVP for displaying mobile notifications.

2.2 Mobile Notifications

Mobile notifications have permeated our lives. Their scope has been widely studied in terms of notification types, numbers, and usage times [3]. Sahami *et al.* [36] conducted a large-scale study to understand the extent and user perceptions of mobile notifications. While users need to deal with a large number of notifications in a day, most of them are viewed within a few minutes of arrival. Users further assign different importance to notifications depending on the specific application and application category triggering them. Pielot *et al.* [30] give detailed accounts on how users perceive these different types of notifications. Notifications from messaging apps are generally more important to users as such applications connect people across space and time [29].

There are, however, concerns about notifications significantly contributing to the notion of information overload [30]. There are generally two aspects to this notion: the receipt and the management of notifications. Delivery and interruptability management focus on inferring opportune moments to trigger notifications and reduce user interruptions [13, 21]. Mehrotra *et al.* [27] investigated different notification groupings, namely by the applications that initiated the notification and the social relationship between sender and receiver. Combined with user context derived from phone sensors, they devised a classifier to detect opportune moments for effective notification delivery. They subsequently built a rule-based system that would filter notifications to effectively reduce the overall number of notifications users have to deal with [26]. Weber *et al.* [46] proposed *Notification Dashboard*, an Android app that allows users to reflect on their notifications, identify, and subsequently reduce unwanted interruptions.

Notification management itself, however, focuses on how we allow users to decide when and how to engage with incoming notifications. Auda *et al.* [1], for example, presented notification summaries based on a user-defined set of rules: Instead of triggering each incoming notification individually, the proposed app collates notifications in a daily summary. Android's notification drawer (or *notification center* on *iOS*) has generally evolved into a central place where mobile users view and attend notifications. Turner *et al.* [45] investigated how stack size and notification position in the notification drawer influence the user's response process. They subsequently devised user strategies for managing the notification stack within usage sessions. Their observation that notifications frequently coexist together informed our thinking about alternative ways to group, organize, and display notifications.

Since notifications come in large amounts and take the form of short text bursts, RSVP presents itself as a potentially effective reading technique to deal with them. Hence, we set out to investigate the design parameters of RSVP to put in the first steps towards an alternative way of engaging with notifications. Before being put into wider use, the design of a notification system based on RSVP presents a broad range of challenges for which informed design decisions need to be taken. Following a user-centered design process, we conducted a series of studies on how to group, display, and navigate RSVP notifications to elicit the first set of design parameters for RSVP reading on mobile notifications.

3 FOCUS GROUP

To ideate RSVP notification design and to answer **RQ1**, we conducted a formative focus group to explore users' notification management behavior with RSVP. We recruited nine participants (three women, six men) between 24 and 34 years of age (M=28.11, SD=3.95) to answer a series of questions and discuss potential design ideas. We explored two major topics to inform the application design: (1) advantages and challenges for RSVP in notification reading and (2) essential components for reading notifications in RSVP.

3.1 Procedure

First, to elicit participants' general notification management strategies, we prepared questions to probe the discussion on their practices in handling mobile notifications. Secondly, we introduced RSVP and asked participants to compare the newly introduced RSVP mechanism to their usual reading habits in mobile notifications. Each question was followed by a 3-5 minute idea collection phase, during which each participant individually wrote down their ideas on post-it notes. We then took those notes and clustered them into idea groups, after which the group discussed them.

Finally, we introduced all participants to the idea of presenting mobile notifications in RSVP and conducted co-design exercises with participants to identify potential designs incorporating RSVP in common notification use cases. Each interface design-related question was followed by a 1-3 minute idea-collection time. Participants were asked to visualize their ideas by drawing their designs on print-outs of a generic smartphone display wire-frame. They were then asked to explain their ideas in a five-minute discussion session.

Throughout the focus group, a moderator guided the discussions while two other researchers documented observations and handled the video recording. The experimental design was approved by the Human Ethics Committee of our University.

3.2 Topic Exploration I: Advantages and Challenges

Throughout the focus group, participants pointed out that several RSVP characteristics could benefit notification reading. For example, seven participants mentioned the increase in reading speed, an advantage also stressed in the literature [11]. Six participants said it could increase their attention and help them stay focused while reading. Five participants said that the presentation did not require much space and could, therefore, free up space or present more information: *"It saves space, while the traditional system only shows part of [a] message"* (P1). While participants' statements were in favor of RSVP usage overall, all participants agreed that RSVP did not necessarily benefit their mobile notification reading. The challenges cited include: **lack of reading flexibility**, **requirement for continuous concentration**, and **infeasible for long texts**.

Regarding challenges in adapting to reading speed for mobile RSVP notifications, three participants suggested methods to adjust the speed in case the default is too fast or too slow. Also, the nature of presenting texts in RSVP forces users to read one notification after another, which may lead to a loss of context of a conversation: *"It (RSVP) will miss the context as it forced me to read notifications in order. If I miss the beginning of a conversation, it may result in problems"* (P6). Further, participants expressed worries regarding the reading mechanism being too rigid as *"missing certain words may result in missing the whole content, and it is hard to go back and re-read it"* (P7). Switching attention between different notifications could also be problematic, as P5 pointed out: "A bundle of notifications requires you to read one after another (in RSVP); how can messages be skipped?" In summary, while these concerns aligned with text comprehension challenges in RSVP as pointed out in the literature [11], our participants implicitly stressed the importance of such systems' flexibility to enable users to read and navigate between notifications at their desired pace and order. This may also enable users to have total control of their reading speed and go back to notifications that were not fully read because of distractions or drops in attention.

While seven out of nine participants mentioned that RSVP supports improved concentration while reading, it could also create pressures as it requires continuous attention and certain reading ability: *"you have to constantly look at the message to catch the whole content"* (P6). Two participants further pointed out that reading lengthy texts in RSVP could be challenging. The reason lay in the fact that RSVP displays texts word by word, which is different from users' reading practices: *"sometimes I don't want to read the whole text, just keywords [...] it (RSVP) forced me to read one word after another [...]"* (P2).

3.3 Topic Exploration II: Essential Components

Several interaction components appeared across every participant's interface design when debriefing designs in each brainstorming session. These include functions to **play/pause notification contents**, **dismiss notifications**, **switch to next/previous notification**, and **progress bar** to indicate the current RSVP's play status. Besides using buttons to interact with the notification interface, participants also provided a wide range of interaction ideas, such as using eye-gaze to switch back and forth among notifications (P6) or positioning the smartphone such as tilting to adjust RSVP reading speed impromptu (P7).

As users cannot read the contents of RSVP notifications before clicking the "play" button, P2 and P9 suggested that notifications should be ordered based on their importance to the user. In contrast, P1 suggested ordering notifications chronologically, the same as traditional notification systems. Whereas the discrepancy in ordering notification, all participants agreed that there should be only one notification presented in RSVP at a time, e.g., P1 stated that *"playing two at the same time doesn't make sense, stop one automatically, when a new one is selected to play."*.

Finally, as messaging applications (e.g., Facebook Messenger) usually trigger the most mobile notifications [36], we asked participants to design RSVP notification reading specifically for such applications. Unlike notifications from other types of applications, those from messaging applications create clusters (i.e., a stack of multiple notifications) more easily in notification drawers based on the name of message senders. Interestingly, our participants proposed two approaches to grouping original notifications into RSVP notifications. Six participants separated RSVP notifications based on senders' names (i.e., notification titles). The other three participants grouped incoming notifications by their applications (i.e., application name). This latter group also proposed additional functionalities to help users navigate messages from different senders from the same messaging application.

3.4 Design Implications

The two main findings on **functionality** and **notification grouping** emerged from our focus group. We then translated these findings into the design goals for our RSVP notification system prototype, SpeedNotification.

In general, participants expected RSVP notification reading to be flexible. Therefore, by providing the **ability to switch between notifications** in the system, we could provide a better notification navigation experience, decreasing users' burden when initiating or undergoing RSVP reading sessions. Meanwhile, playing and reading notifications in RSVP may require additional actions from users. Thus, an **intuitive RSVP control panel** is not only necessary for user inputs but also provides users with a better overview of their notifications and their RSVP reading progress.

Notifications are sometimes contextually continuous, such as a series of connected instant messages. Thus, when building a system that transforms traditional notifications into the RSVP presentation format, the system needs to identify different incoming notifications and connect them to create a continuous reading flow. To maximize the advantage of RSVP reading, the system should aggregate related notifications into one RSVP reading flow. This enables users to start a reading flow and automatically read through multiple notifications. However, a trade-off of such a notification aggregation is the potential of increasing the system complexity induced by additional controls, which may negatively influence the user experience. To this end, we explore two approaches to grouping notifications as suggested by the focus group: **1) grouping by title** and **2) grouping by application**. In the former, all notifications with the same title will be aggregated into one RSVP notification (i.e., the same reading flow). We argue that this design supports efficient RSVP reading by requiring less user input when reading; for example, users will no longer need to press *play* for every single message. Also, concatenating notifications based on their titles usually preserves the context, such as related information from the same application or continuous messages from the same conversation. In the second approach, grouping by application, the system additionally groups notification titles from the same application into the same RSVP notification. This design may further facilitate RSVP reading's strength, as more notifications will be grouped into one RSVP notification compared to grouping by title. However, the control panel interface for such a grouping mechanism also requires additional controls for users to navigate notifications in this two-layer grouping: grouping original notifications based on their title and an additional grouping by their applications.

4 SPEEDNOTIFICATION

We implemented our RSVP notification prototype, SpeedNotification, based on an Android open-source project, Spritzer-TextView [1]. The resulting app intercepts notifications triggered on the mobile device using NotificationListenerService [2] and replaces them with RSVP notifications.

RSVP notifications present words in RSVP, which sequentially display a word at a time, with a progress bar indicating its playing progress, as shown in Figure 1a. Following the common RSVP text presentation (e.g., [9, 10, 35]), one letter of the word will be colored in red, which marks the optimal viewing point and acts as an attention pivot for the reader's eyes to focus on. The system also considers word lengths and punctuation to determine the duration of the presentation of each word. The duration is three times as long for words with more than six characters and words followed by specific characters, including comma, colon, semicolon, period, question mark, or exclamation mark.

The open-source framework originally allowed to actively start the reading flow of RSVP and the reading speed setting in words per minute (WPM) inside the application. We modified the source code and enabled the control of these features inside a custom notifications layout [3]. The user interface design and implementation were based on the insights regarding design parameters elicited from the focus group. For its functionality, we provide *play/stop* to start or stop the RSVP reading flow, *switch back/next* to switch back and forth among notifications with the same notification title, *speed up/down* to adjust reading speed impromptu with a ±10 WPM step. Note that the speed modified in this way would not be saved and carried over to the next reading event (see Figure 2).

Another main parameter for mobile RSVP notification we learned from the focus group is the grouping mechanism. As a result, we created two versions of SpeedNotification based on the two mechanisms to group incoming notifications: (1) grouping by title and (2) grouping by application. For app version (2), we also added *switch next title* to enable the navigation to the following notification title and *expand/collapse* to show and hide all notification titles from the same application. When the user hits *play* in an RSVP notification in-app version (2), as shown in Figure 1b, all notifications

[1] https://github.com/andrewgiang/SpritzerTextView
[2] https://developer.android.com/reference
[3] https://developer.android.com/training/notify-user/

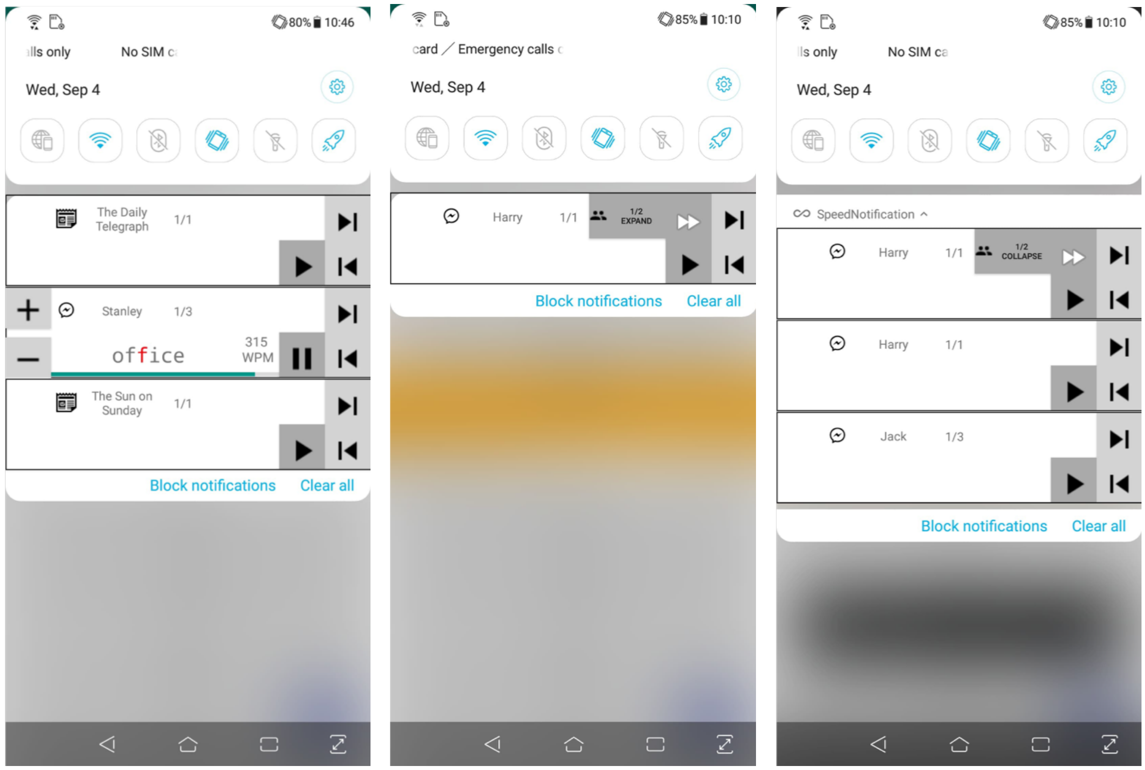

Fig. 1. The interface of SpeedNotification. (a) RSVP notifications grouped by notification title (the middle notification is currently being played); (b) RSVP notifications grouped by application and notification title (collapsed); (c) RSVP notifications grouped by application and notification title (expanded).

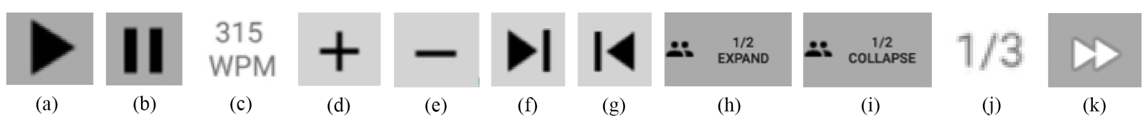

Fig. 2. SpeedNotification functionality: (a) play; (b) pause; (c) current speed; (d) increase speed; (e) decrease speed; (f) next notification; (g) previous notification; (h) expand notifications from one app; *current index* & *total amount* of notifications (title) from the app; (i) collapse notifications; (j) *current index* and *total amount* with the same title; (k) next notification (title) of the app;

from that cluster (i.e., notifications from the same application) will be displayed sequentially in one RSVP flow. If the user selects *play* in the "child notification," reachable through expanding the "parent notification" (see the two lower notifications shown in Figure 1c), only notifications from that title will be played in that specific RSVP flow, same as the mechanism in app version (1). For both app versions, we separate different notifications displayed in one reading flow by implementing a 3-second time-out before the start of each notification, of which users could also click the *next notification* button to skip.

To better understand users' interactions with RSVP notifications, SpeedNotification captures users' usage logs in the app, such as click actions, speed configurations, and records of each word presented (e.g., reading speed, reading progress at the moment). We also logged notifications' meta-data, such as title, arrival timestamp, and package name.

5 LAB STUDY

We conducted a controlled lab study to understand the usability of SpeedNotification. Specifically, we investigated factors of RSVP notification reading that we hypothesized might affect user experience, including **notification type**, **notification length**, and **distraction**. Additionally, we evaluated the design proposals for the two main design parameters: **notification grouping** and functionalities specifically for impromptu **reading speed adjustment**. We also collected qualitative feedback from participants to better understand how they read notifications in RSVP.

Notification Type. We categorized notifications as *standalone* notifications, which independently convey a message such as a news headline, and *consecutive* notifications, which can only be fully interpreted when read in sequence, such as a contact's text messages received in short succession.

Notification Length. Our focus group discussions suggest that the length of notifications can influence the RSVP reading experience. Following the size and resolution limitations of the notification displays on typical iOS and Android devices[4], we defined four levels of notification lengths in this study: 1) single word (including emoji), 2) short: less than 40 characters ($\approx$ 10 words), 3) medium: 100 to 150 characters ($\approx$ 30 words), and 4) long: around 450 characters ($\approx$ 70 words). Specifically, (2) and (3) are typical caps for notification length in Android and iOS systems, respectively, and (4) is the limit for expanded notifications, which are designed for text-rich notifications in Android devices[5].

Distraction. These are notifications that are irrelevant to the one sought by the user, potentially inducing distraction in the notification bar.

Notification Grouping. We explored three approaches to generating RSVP notifications that create different RSVP reading flows. Informed by our focus group findings, these include two grouping mechanisms implemented in SpeedNotification: 1) **grouping by title**, which consecutively groups by notification title (i.e., contact name, headlines; see Figure 1a), and 2) **grouping by app**, replicates (1) and groups additionally by the triggering applications (see Figure 1b and 1c). As a baseline comparison, we also tested notifications displayed in chronological order with **no grouping** — users read RSVP notifications separately and click the *play* button to read each notification individually.

Reading Speed. Reading speed is defined as the number of words per minute (WPM). SpeedNotification requires users to configure their initial default RSVP displaying speed after installing the application, while such a speed is adjustable in the settings as users continue using the application. In contrast to the global default speed, the system, as mentioned earlier, also provides functionalities of changing the speed impromptu during reading events ((d) and (e) in Figure 2). In the lab study, we compare user experience between the conditions where this impromptu speed control is available and where it is not.

[4]https://support.airship.com/hc/en-us/articles/213491643
[5]https://developer.android.com/training/notify-user/expanded

5.1 Experimental Design

The lab study is in two parts, lasting about 30 and 10 minutes, respectively. In the first set of tasks, we investigated factors of notification types, extent of distraction, and notification grouping mechanisms on RSVP notification reading. In the second set of tasks, we evaluated the effects of different notification lengths along with the availability of the impromptu speed controller. The experimental design was approved by the Human Ethics Committee of our University.

Part 1 employed a $2 \times 3 \times 3$ repeated-measures design with three independent variables: notification type (standalone and consecutive), distraction (zero, two, and five distracting notifications), and notification grouping (no grouping, grouping by title, and grouping by application). The experimental task simulated smartphone use and employed news headlines as standalone notifications (e.g., *The nightmare ends: Italy allows migrant rescue ship held for weeks off Sicily to dock*), and groups of consecutive instant messages from the same contact (e.g., *What is the name of the hotel?*, *It does not show up on my search*, *Maybe it has been booked in the meantime?*). We used notifications with different content for each trial.

Part 2 adopted a 4×2 repeated-measures design with two independent variables: notification length (single word, short, medium, and long) and impromptu reading speed control (enabled or disabled). The experimental task required participants to read text messages using RSVP on a mobile device with SpeedNotification. Again, we used notifications with different content for each trial.

5.1.1 *Measures.*

We measured *perceived satisfaction* using the part of the Questionnaire for User Interaction Satisfaction (QUIS) called "overall reactions to the software" a tool widely used to assess interactive systems [6]. We also measured *perceived usability* with the System Usability Scale (SUS) [4] for its reliability in its measurement [44], and considered the usability benchmarks suggested by Bangor et al. [2]: "not acceptable" (<50), "low marginal acceptable" (50-62), "high marginal acceptable" (62-70), and "acceptable" (>70). In addition, we collected qualitative feedback through interviews after the completion of each part. In part 2, we recorded user activity related to impromptu reading speed changes.

5.2 Participants

We recruited 18 participants (nine women, nine men) with a mean age of 25.8 years (SD=3.6) through university mailing lists and bulletin boards. 50% of the participants reported iOS as their mobile operating system, while the remaining 50% reported Android. On a scale from one (unfamiliar) to five (familiar), participants rated their familiarity with RSVP with a median rating of 1.5 (SD=1.21). The study took about an hour to complete, for which participants received shopping vouchers worth $10 (AUD).

5.3 Procedure

We welcomed participants upon their arrival and introduced the purpose of our study. Participants read and signed our consent forms, reported their demographic data, and answered a brief survey about their use of mobile notifications. We began the training session by explaining RSVP. Then, we handed out a smartphone running SpeedNotification, explained the interface design, and walked participants through the tasks and interviews they would be asked to complete. We also familiarised participants with the tasks by asking them to read a notification with RSVP and answer a comprehension question about the content to validate a minimum reading ability. We explained that after each reading task, they would be asked such comprehension questions to confirm that they had read the intended text. We adopted this approach to ensure that participants pay attention to reading tasks [11, 35]. Once participants confirmed their

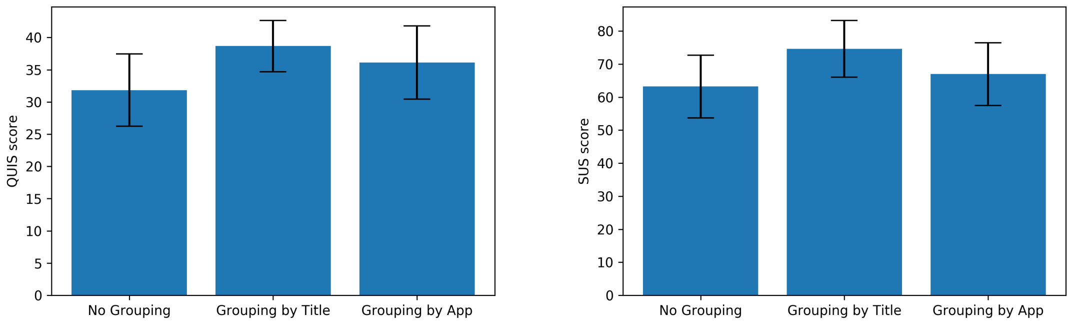

Fig. 3. Average QUIS (left) and SUS (right) scores of different notification groupings in lab study. Shows the mean and its 95% confidence interval.

understanding of the technique and the procedure, they set the default RSVP display speed to their desired comfort level. They were also informed that this global speed would remain the same for the rest of the study.

Before part 1, we informed participants that they could stop or replay any notification in the notification bar during tasks until they were ready to answer the comprehension questions. We also asked them to repeat the reading task until they answered the comprehension questions correctly. To avoid systematic differences regarding reading experience due to the contents, we collated the contents of the reading tasks from common notification scenarios, and both notifications and comprehension questions were tested in pilot studies to ensure they were not too difficult to comprehend and answer. Then, we rolled out the tasks in three rounds, each testing one notification grouping mechanism. Each round contained six trials covering conditions with three distraction levels (zero, two, and five irrelevant notifications) for both notification types (standalone and consecutive). The tasks were either reading one standalone notification (i.e., news headline) or reading all consecutive notifications (i.e., messages from a contact). We balanced the order of the three rounds between participants using a Latin Square design. Participants completed the QUIS and SUS questionnaires after each round. After completing all tasks, we interviewed participants about their experience with standalone and consecutive notifications reading via RSVP. We asked them to describe their experience with different types of notifications, distraction levels, grouping mechanisms, and how each factor affected their reading. We also asked participants to rank their preferences for the three proposed grouping mechanisms.

In part 2, we introduced the functionality of the impromptu RSVP display speed controller. The tasks were mostly equivalent to part 1 but without distractions — only the targeted notification in the notification bar. Tasks were completed in two rounds, each containing four levels of notification lengths, and the functionality of the impromptu speed controller was only enabled in the second round. Participants were also informed that even if they changed the speed via the controller while reading, each new trial would start with the default reading speed they set at the beginning of the study. After all the tasks were completed, we conducted interviews about participants' experience with each notification length and their experiences with the impromptu speed controller. Finally, we interviewed participants to investigate their general experience with SpeedNotification, such as features they liked to keep or change and the benefits and challenges they saw if using the system in their daily lives.

5.4 Results

5.4.1 Subjective Usability Assessment. Participants assessed their perceived satisfaction and usability of each notification grouping via QUIS and SUS scores (see Figure 3). Participants completed a subset of QUIS focusing on "overall reactions to the software" (six rating scales with a 10-point scale from 0 (negative) to 9 (positive)). We calculated the user interaction satisfaction score by summing up all six scores from the QUIS per participant per grouping mechanism. Hence, higher scores indicate increased satisfaction with the system as reported by our participants. A one-way analysis of variance (ANOVA) shows that the grouping condition has a statistically significant effect on usability assessed by QUIS, $F(2, 34) = 5.496$, $p = 0.009$ accounting for 24.4% of the variability. Post-hoc paired t-tests using the Bonferroni correction revealed that there is a statistically significant difference between *grouping by title* ($M = 38.67$ ($SD = 7.97$)) and *no grouping* ($M = 31.83$ ($SD = 11.3$)), $t(17) = -3.172$, $p = 0.006$. The other comparisons were non-significant.

Analysis of the SUS revealed a statistically significant difference between grouping conditions, $F(2, 34) = 5.767$, $p = 0.007$, accounting for 25.3% of the variability, with *no grouping* ($M = 63.19$ ($SD = 19.15$)) and *grouping by app* ($M = 66.94$ ($SD = 19.03$)) achieving the usability benchmark "high marginal acceptable" (score > 62), and *grouping by title* ($M = 74.58$ ($SD = 17.26$)) achieving the usability benchmark "acceptable" (score > 70). Similarly, posthoc tests revealed that there is a statistically significant difference between *grouping by title* ($M = 74.58$ ($SD = 17.26$)) and *no grouping* ($M = 63.19$ ($SD = 19.15$)), $t(17) = -2.969$, $p = 0.009$. Comparing *grouping by title* and *grouping by app* ($M = 66.94$ ($SD = 19.03$)), resulted in $t(17) = -2.969$, $p = 0.019$, *i.e.*, not statistically significant after Bonferroni correction ($alpha = 0.017$). The comparisons between *no grouping* and *grouping by app* were also non-significant.

5.4.2 Reading Log Analysis. Next, we look at how participants change their RSVP display speed impromptu while reading notifications of different lengths. Firstly, no participant changed their speed when reading single-word and short notifications. Such a finding may be reasonable as the time and attention span required to read them are short. On the other hand, we found that both the number of occurrences and the extent of the impromptu speed changes increased as the notifications got longer. Specifically, when reading medium-length notifications, seven participants slightly changed their reading speed. Five of them increased on average by 26 WPM (SD=20.74), and two of them decreased the speed on average by 15 WPM (SD=7.07). A similar pattern was found in long notification reading, where 13 participants changed their speed while reading, with seven of them increasing by an average of 65.71 WPM (SD=62.41), seven decreasing by an average of 27.14 WPM (SD=24.98), and one first increased and then decreased the speed. However, considering these results were only based on the users' first interaction with the system, we need to observe RSVP reading behaviors on a larger scale to infer users' actual reading practices to notifications of different lengths.

5.5 Qualitative Findings

For one of the two main design parameters, notification grouping, 12 participants found a *no grouping* option "cumbersome" (P5) because they had to spend more time and energy on finding specific notifications (P4, P6). Fourteen participants found *grouping by title* to be preferable over the *no grouping* design, as P12 commented: *"better than the non-group one cause it decreases the clicks needed when reading notifications, and I feel that I could control these notifications directly and read fast."* The design also received overall positive feedback among participants, citing reasons such as efficiency in reading all related notifications (P3, P5, P8, P9, P11), clear/straightforward (P10, P15, P16), and easy to use/handy (P6, P7). On the other hand, we received mixed feedback from participants for *grouping by app*. It was disliked by 12 participants who found it *"too complicated, too packed"* (P2), while six participants reported a favorable experience, with P9 saying: *"the best, I could use the app as another filter before reading notifications."* Overall, we found a clear preference for grouping notifications, with *grouping by title* being the favorite of the majority. This aligns with our findings from participants' subjective assessment that designs of grouping notifications (e.g., *grouping by title* and *grouping by app*) achieved better-perceived satisfaction and usability scores than their non-group counterpart, with the *grouping by title* achieves the highest scores.

For another design proposal we investigated in the lab study: impromptu display speed controller, among all participants, six stated such functionality was beneficial for their usage. Four participants said they were already satisfied with the initial default speed and would not require any spontaneous changes.

When asked about how notification length affects their RSVP notification reading, 16 participants preferred RSVP notifications of short ($\approx$ 10 words) to medium lengths ($\approx$ 30 words). Within this group, P8 and P14 stated a maximum length of 10 words as the threshold before messages become too long for RSVP. The assessments were particularly negative for long notifications ($\approx$ 70 words) as users perceived reading burdens and lost the context of texts, which harms the comprehension in RSVP [11, 25]. Additionally, readers may lose interest and focus in lengthy word-by-word readings: *"(The reading experience is) Not good. Forget the previous text that I just read. [...] If notification's not interesting or important, I get bored"* (P18). Interestingly, while the prior study suggests that RSVP reading experience could make a boring fairy tale story more interesting [11], our finding suggests the opposite effect in a daily usage scenario, indicating that the benefit of RSVP on the reading experience may be content-dependent.

Among the challenges that participants faced while reading RSVP notifications, the demand for attention was named the biggest problem. When asking participants for a preferred RSVP notification length, the desired reading length was 10 words: *"I feel like that I read faster compared to when I read notifications in Android - I think RSVP is suitable for short texts"* (P12). Seven participants found RSVP ineffective for extremely short messages (*i.e.*, one-word messages). P17 stated that the RSVP notification display was *"Really fast, could not react to that."* and that it *"makes more sense for RSVP to show at least a sentence."*

To conclude the interview, we asked participants to name features of SpeedNotification that they would keep or change, that they find beneficial, and those that cause problems. Sixteen out of 18 participants thought the idea of grouping notifications should be kept. Among these, seven preferred the design of *grouping by title*, and five preferred the design of *grouping by app*. Four participants mentioned an alternative grouping mechanism of notifications only grouped by their application and not by their title [6]. Six participants stated that the impromptu speed controller should be kept, and another six participants named the button to switch between notifications as beneficial. Finally, four participants mentioned the unpleasant experience while reading lengthy notifications in the interface when asked about features that they think are problematic.

6 FIELD STUDY

To consolidate our findings from the lab study and to answer **RQ2**, we conducted a field study to understand the user experience and their interactions with RSVP notifications in the wild. We installed our SpeedNotification prototype on participants' smartphones for one week, recorded participants' activity in the app, assessed their subjective usability assessment, and collected their feedback.

[6]This proposal is similar to *grouping by title*, but simply groups notifications from the same application into one RSVP notification without further distinguishing them by their titles.

6.1 Experimental Design

We employed a between-subjects design by splitting participants into two groups: Group T used the *grouping by title* version of SpeedNotification and Group A used the*grouping by app* one. Our prototype was installed on the participants' smartphones. SpeedNotification reads notifications from applications specified by participants in the settings and presents them as RSVP in the notification bar. During the first three days, we asked participants to use RSVP for notifications received from all applications. For the next four days, participants were allowed to disable RSVP for applications of their own choice. We conducted the field study with the same prototype as in the lab study, and participants reported their experience with QUIS, SUS, and a post-study questionnaire. The experimental design was approved by our university's Human Ethics Committee.

6.2 Participants

We recruited 23 participants[7] (12 women, 11 men) with a mean age of 25.9 (SD = 4.8) through university mailing lists and bulletin boards. We only recruited participants who use Android smartphones to ensure compatibility with our prototype. When asked about the daily number of received notifications, three reported less than ten, 14 reported 10-30, four reported 30-50, and two reported more than 50. Participants also had overall low familiarity with RSVP with a median rating of 1 (SD = 1.23) on a five-Likert scale (1: unfamiliar - 5: familiar). Following this, we instructed participants to use the prototype for at least seven days. The average days of their participation are 8.04 (SD=1.49). Participants received shopping vouchers worth $20 (AUD) after completing the study.

6.3 Procedure

Before the field study commenced, we arranged a briefing with each participant, in which they were introduced to the purpose of our study, read and signed the consent forms, reported their demographic data, and answered a brief survey about their use of mobile notifications. As we had randomly assigned each participant to a group (A or T) to ensure the two groups had balanced men and women participants, we installed the respective version of the prototype on their smartphones and walked through its features. We then instructed them to enable RSVP for all applications so that they would receive notifications. After the 3-day mark, we instructed them to disable RSVP for the applications of their choice via email. We also reminded participants that they could change their RSVP display speed anytime during the study. We arranged a time for a post-study debriefing session one week later.

Debriefing happened in the form of a post-study questionnaire, in which we asked participants to complete the QUIS and SUS assessments. We also asked questions specifically about their experience in reading mobile notifications in RSVP (e.g., what types of notifications were preferred for RSVP reading) and their preference when using SpeedNotification (e.g., strengths and weaknesses of SpeedNotification, features that are helpful/pleasant and problematic/unpleasant).

6.4 Results

6.4.1 Subjective Usability Assessment. The perceived usability and satisfaction of the two notification grouping mechanisms were assessed via QUIS and SUS scores (see Figure 4). Analysis of the **QUIS** using an independent samples T-Test showed no statistically significant difference between *grouping by title* (M = 38.25 (SD = 6.73)) and *grouping by app* (M = 35.55 (SD = 7.33)), $t(21) = 0.923$, $p = 0.83$. Similarly, comparing **SUS** scores between *grouping by title* (M = 70.62 (SD = 12.25)) and *grouping by app* (M = 68.86 (SD = 9.77)) yielded no statistically significant difference (

[7]The initial number was 24. We discarded data from one participant who could not continue due to a compatibility issue after an OS update.

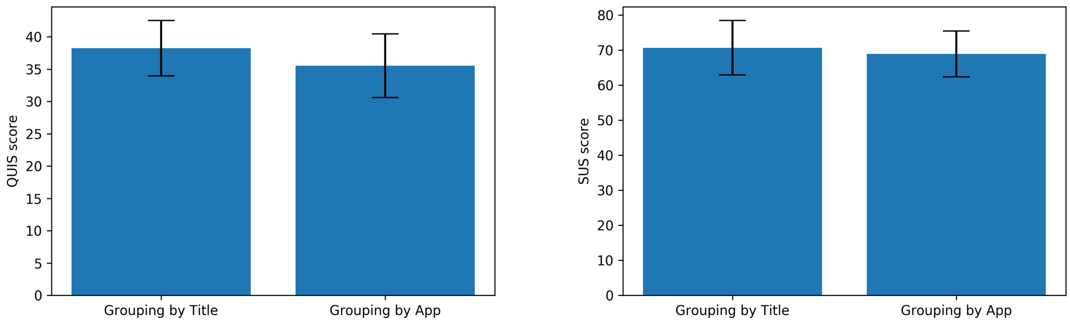

Fig. 4. Average QUIS (left) and SUS (right) scores of the two notification groupings in the field study. Shows the mean and its 95% confidence interval.

$t(21) = 0.379$, $p = 0.606$). The two designs again achieve Bangor *et al.*'s usability benchmark of "acceptable" (70.62) and "high marginal acceptable" (68.86), respectively.

6.4.2 Reading and Interacting with RSVP Notifications In-the-wild. We first categorized notifications into *standalone notifications* and *consecutive notifications* following the same rationale as described in Section 5. Across all participants, a total of 83,795 RSVP notifications were triggered, with an average of 3,643 ($SD = 5,333$) per participant. Among these, the majority (71.07%; $SD = 32.23$) were *consecutive notifications*.

In total, our participants clicked the *play* to read notifications in RSVP 2,284 times, with an average of 99.3 times (SD=93.07) per participant. As the majority of notifications received by participants were *consecutive notifications*, participants using *grouping by app* (Group A) and *grouping by title* (Group T) also "played" the respective types of notifications more often with A: 53.74% ($SD = 30.29$) and T:64.95% ($SD = 21.85$) of the times in average. While participants of either group received around 70% of the *consecutive notifications* (Group A: 69.46%; Group T: 72.55%), we found that participants in Group A had a relatively balanced result of "plays" between *standalone notifications* and *consecutive notifications*. A possible explanation is that participants in Group A were able to read more *consecutive notifications* in a single reading flow, resulting in fewer total clicks while reading them.

To take a closer look at participants' RSVP reading patterns in the two prototypes, we calculated the ratio of the number of *play* clicks to the number of notifications read. More specifically, how many notifications could a user read, on average, in an RSVP reading flow (i.e., per one *play* click). In general, participants read 1.76 (SD=1) notifications per one *play* click. Interestingly, we found differences in such a ratio between Group T and Group A. Users in Group A could read roughly two notifications ($M = 2.02$ ($SD = 1.39$)) per *play* click, whereas those in Group T read less ($M = 1.52$ ($SD = 0.31$). The result suggested that *grouping by app* may help users to aggregate more notifications in their RSVP reading flows, which was further supported by the fact that users in Group A almost exclusively play their notifications within "parent notifications" ($M = 93.38\%$ ($SD = 8.47$)) instead of in "child notifications." As a reminder, the former creates longer RSVP reading flows and displays all notifications from the same application. The latter, same as that in *grouping by title* of Group T, only displays notifications under the same title.

6.4.3 Investigation of RSVP Notification Reading In-The-Wild.

Default Global Display Speed Changes: Among 23 participants, the average initial display speed was set to 305 WPM (SD=95.85). The average speed after finishing the study was 320.57 WPM (SD=128.89). In total, eight participants

Table 1. Completion rate, reading portion, reading span, and aborted reading span of each type of notification.

Notification Type	Short	Medium	Long
Completion Rate (% of the type of notification)	91.74 (SD=7.55)	70.68 (SD=19.7)	37.67 (SD=29.4)
Reading Portion (% of a notification)	95.83 (SD=3.94)	82.33 (SD=12.92)	54.32 (SD=27.25)
Reading Span (second)	0.94 (SD=0.42)	5.77 (SD=2.8)	32.07 (SD=29.19)
Aborted Reading Span (second)	0.48 (SD=0.65)	1.88 (SD=1.67)	9.65 (SD=15.03)

changed their speed during the study: among which five increased it by an average of 85 WPM (SD=49.25); three decreased it by an average of 22.33 WPM (SD=18.82).

Impromptu Display Speed Changes: Consistent with the lab study's findings, our participants seldom changed their display speed impromptu while reading short notifications (≤10 words). More specifically, among 2,888 such reading events, only nine times (0.31%) was the speed changed while reading. Regarding findings for reading medium-length (11-30 words) and long (>30 words) notifications, however, a discrepancy was found between the lab study and the field study. In the former, one-third of participants adjusted their speed while reading, and two-thirds of the participants did so when reading long notifications; in the latter, respectively, participants rarely changed their speed impromptu. Out of 426 reading events of medium length, only four times (0.94%) were the speed adjusted; from 254 long notifications, only seven times (2.76%) were the speed adjusted. On average, participants adjusted their speed impromptu 0.87 times (SD=1.74) during the study, with 13 participants never changing it impromptu. In other words, in most cases, participants keep the same speed to read through notifications.

Attention Span Based on Notification Length: Participants, on average, read 9.26 words (SD=4.8) per notification and finished reading on average 89.5% (SD=6.1) of a notification's contents once they "played" it. In terms of the reading completion rate (i.e., the ratio of notifications that participants read through), participants finished reading 83.54% (SD=10.78) of the notifications that they played, which spent an average of 3.36 seconds (SD=5.05). When notifications were not finished, they were aborted after an average of 4.02 seconds (SD=4.59). However, we found different reading patterns with different lengths when we separated notification reading events based on their length: short (≤10 words), medium (11-30 words), and long (>30 words) (see Table 1).

For short notifications, participants, on average, finished reading through 95.83% (SD=3.94) of such notifications and finished reading 91.74%(SD=7.55) of the contents once they played. They spent less than a second (Mean=0.94; SD=0.42) to finish such reading events, and all of the unfinished reading events were aborted after an average of 0.48 seconds (SD=0.65). As the notification length increased, perhaps expected, fewer portions of notifications were read, with an average of 82.33% (SD=12.92) and 54.32% (SD=27.25) for medium and long notifications, respectively. Furthermore, a decreasing trend in the averaged reading completion rate (medium: 70.68% (SD=19.72); long: 37.67% (SD=29.4)), and an increasing trend in the averaged period (medium: 5.77 seconds (SD=2.8); long: 32.07 seconds (SD=29.19)) were found in longer notification reading events: medium and long notification reading. For those longer reading events left unfinished, participants aborted them after 1.88 seconds (SD=1.67) and 9.65 seconds (SD=15.03) on average in medium and long notification reading, respectively.

The general trend of: "the longer the text, the more likely the reading event will be aborted" could be further supported by the decreasing trend of the reading retention rate (i.e., the ratio of a notification still being read at ℓ where $\ell \leq$ the

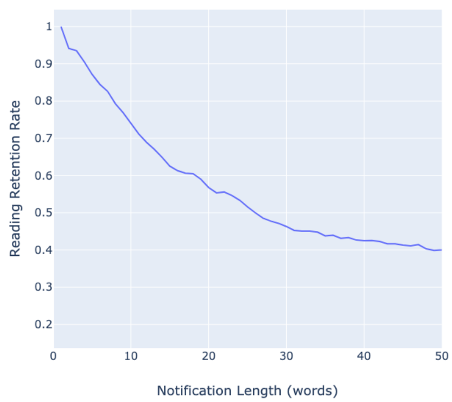

Fig. 5. RSVP notification reading retention rate at each length. Notifications with lengths no greater than 50 words comprise 95% of all reading events.

length of the notification) as the length increases (see Figure 5). Note that the time users abort a reading of medium or long notifications is generally much shorter than the average time they could finish such reading events. We speculate that instead of having a generic "attention threshold" for reading notifications in RSVP, the attention resources may be adjusted based on their in-situ reading experiences. More specifically, during a mobile RSVP notification reading, they may quickly decide, at least faster than the time they need to finish the reading, to either continue reading, give up halfway, or finish reading and perform the following action (e.g., click on or dismiss a notification). Longer notifications could also mean higher chances that users will change their attention level during reading events.

6.5 Qualitative Feedback

General feedback on SpeedNotification reflected easier and more efficient access to notifications enabled by our proposed notification grouping, *grouping by title*, as noted by P8: *"You can consume more of the notification without having to click into the application. Scrolling isn't needed."*. When asked about what features they especially liked, *grouping by title* was mentioned again: *"The clear classification made it easier for me to know what kind of information it was"* (P10). For *grouping by app*, two participants indicated that being able to expand and collapse grouped notifications was helpful. However, one participant in Group A complained about the extra complexity induced by the feature: *"For the normal (traditional) notifications, I just need to slide down and I can see all the notifications. It's much easier and more flexible. But with this app, I have to click the button."* Seven participants praised the rapid presentation of text with RSVP notifications, indicating that it allows them to read faster. Better focus and better comprehension are other advantages mentioned: *"Since we are reading it word-by-word, it's less likely that we miss a keyword, especially in a shorter message."* (P3), indicating that SpeedNotification inherited the advantages of RSVP reading found in the literature [11].

Participants also reported how various factors affected their mobile RSVP notification reading experiences. Regarding the effect of notification length, overall, shorter notifications were widely preferred over longer ones, with eleven participants confirming this in their comments, aligning with our lab study findings. With a similar token, most participants expressed negative sentiment towards longer notifications, citing reasons such as demanding time and effort, lack of efficiency (P2), and more difficulty in focusing when reading (P4). Other reasons for the unpleasant experience include difficulty in remembering the whole message due to lack of patience (P13), and irrelevant contents when reading, such as Email closings and styling characters (P14, P23). Finally, mobile RSVP notification reading experiences were reported to be influenced by notification types: fourteen participants preferred instant messages, specifying that they are shorter and demand only fragmented attention.

7 DISCUSSION

The design of SpeedNotification was focused on the two main design parameters we identified in mobile RSVP notification reading: notification grouping and functionality. For the former, the feature *grouping by app* explored a more efficient notification clustering concept. Indeed, our studies showed that by aggregating more notifications into one, we could facilitate users' RSVP reading flow by allowing them to read more notifications within one *play* event. However, potential usability issues could be induced by interacting with more buttons to overview all notifications collected in a cluster. Those issues made the reading experience less user-friendly than its *grouping by title* counterpart. This was also reflected in the achieved usability scores, as *grouping by app* was rated lower than "acceptable" in both the lab and field study. The implication for future systems is to apply notification grouping mechanisms that enable an overview of all notifications without inducing additional interaction complexity. Further, the feature with "acceptable" usability scores: *grouping by title* favors notifications stemming from messaging apps, which are often consecutive short prompts and are the type of notifications most valued by users [10, 36]. By considering users' reading habits, difficulty in comprehension, and familiarity with the contents of different notifications, future systems could adjust the reading mechanism accordingly to fully reveal the benefits of RSVP reading for different notification types.

To assess SpeedNotification's functionality, we investigated 23 users' RSVP notification reading in both a lab and field study. From the field study, we found that even within one week, our participants' reading speed in RSVP increased by up to 36%, which corroborates with prior work [34]. As these adjustments happened over the first few days while using the app, future systems may include a "training period" for users to adapt to RSVP reading and adjust its speed gradually. Our manual speed adjustment system was simply a preliminary design proposal, and future systems should consider more implicit options for speed controller, e.g., based on text length and integration of physiological signals indicative of attention levels, such as eye gaze, eye blink, or heart rate [9, 18, 42].

Our findings in medium and long notifications reading's impromptu speed adjustment elicited a discrepancy between the lab study and the field study. We found that the field study participants exhibited a much lower tendency to adjust speed impromptu compared to those in the lab study. A possible explanation is that participants in the lab study were informed that their notification comprehension was to be tested. Thus, they may have paid more attention to the reading to ensure better understanding, leading to adjusting the speed more often while reading. On the other hand, users may not necessarily pay a similar amount of cognitive attention when reading notifications in everyday settings. Such fluctuating attention levels could potentially be caused by distractions and fatigue. Consequently, the speed was less often adjusted in real-time. However, considering our small sample size and relatively short experiment period, we cannot simply conclude that changing speed impromptu is not important for RSVP notifications. Instead, we encourage future systems to explore such functionality with a broader population and for longer periods and explore more input and output modalities on mobile devices to support the design, which we elaborate on as follows.

7.1 Design Space for Future Notification Systems Incorporating RSVP

Indeed, comparing an early-stage design with established systems can be unreasonable and harmful to novel ideas [17, 28]. Thus, this work does not intend to compare RSVP notifications to the current standards (e.g., Android, iOS). Especially, RSVP notification assessments are likely to be negatively influenced by the novelty effects that RSVP introduces and by the alienating effect induced by forced reading speeds [11]).

RSVP notification reading, at this point, is still immature and under-explored. To this end, this paper serves as a preliminary exploration of the potential and design space of RSVP mobile notification systems. Specifically, our findings across the three studies revealed a preference and feasibility of RSVP for short (within 10 words) and medium (within 30 words) length notifications — common length caps for notifications displayed in Android and iOS systems. This suggests a potential RSVP applicability to mobile notification systems that users are familiar with. Moreover, via our proposed notification grouping mechanisms, RSVP notifications could potentially mitigate information overload by reducing the number of stacked notification prompts [30]. Future research will benefit from these findings by extending the concept of mobile RSVP notifications with SpeedNotification as a baseline for comparative investigations. Based on our investigations, we propose three design themes for future notification systems utilizing RSVP:

7.1.1 Notification Grouping Mechanism. Identified as one of our main design parameters, notification grouping could maximize the RSVP reading advantage by aggregating notification reading flows. For this purpose, we have explored two grouping mechanisms: grouping by title and grouping by application. Building upon our findings, future systems should also explore other possibilities to better aggregate notifications displayed in RSVP while providing an intuitive navigation experience (e.g., application category, notification length), striking a finer balance between reading flow and interaction complexity.

7.1.2 Flexible and Adjustable Reading Experience. While RSVP benefits notification reading in certain contexts, users generally expect more flexibility, such as quickly grasping the topic before reading a lengthy message or switching back to non-RSVP reading when needed. A one-fit-for-all RSVP experience may not be feasible all the time. Future RSVP notifications should explore a better utilization of the screen space freed up by displaying only one word at a time and provide better reading flexibility, such as displaying notification keywords when the RSVP display is idle. As users' RSVP reading experiences vary depending on the notification types, future systems should also consider more user-specific RSVP configurations, for example, by enabling the customization of the control panel based on notification meta-data (e.g., application category, notification length) to adjust to different reading contexts.

7.1.3 Interaction Augmentations. Future mobile RSVP notification systems could explore using smartphone sensor data to improve the reading experience through, e.g., *Intangible Controls* and *Context-aware Adjustment*. For the former, gyroscope input could be used to adjust impromptu RSVP display speed or to switch among different notifications, and microphones could enable voice input. By freeing users from touch-based interactions [10], future RSVP notification systems could provide more intuitive, seamless, and unobtrusive interactions. For *Context-aware Adjustment*, as users may have different preferences for RSVP presentations based on their activity context, such as sitting and walking [35], future systems could adjust to users' identified on-going activities and adapt a suitable RSVP configuration accordingly.

8 CONCLUSION

We presented a series of three studies as a user-centered design approach to incorporate RSVP into mobile notification reading. First, we conducted a formative focus group and identified two main design parameters for RSVP notifications: functionality and notification grouping. Based on those insights, we built an RSVP notification system prototype — SpeedNotification — and tested its perceived usability and satisfaction in a controlled lab study. Finally, we evaluated users' experiences and interactions with RSVP notifications via SpeedNotification by conducting a one-week field study.

Our findings throughout the three studies suggest that RSVP is more feasible for shorter notifications (e.g., instant messages) than longer ones (e.g., lengthy emails). In addition, users want functionalities that enable reading flexibility (e.g., overviewing and navigating among notifications). Finally, grouping notifications by title (e.g., names of messaging contacts, news headlines) creates an effective RSVP reading flow and can avoid overly complex interfaces to navigate notifications. With our findings, we inform the design of future systems that incorporate RSVP for notifications.

REFERENCES

[1] Jonas Auda, Dominik Weber, Alexandra Voit, and Stefan Schneegass. 2018. Understanding User Preferences towards Rule-Based Notification Deferral. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI EA '18)*. Association for Computing Machinery, New York, NY, USA, Article LBW584, 6 pages. https://doi.org/10.1145/3170427.3188688

[2] Aaron Bangor, Philip T. Kortum, and James T. Miller. 2008. An Empirical Evaluation of the System Usability Scale. *International Journal of Human–Computer Interaction* 24, 6 (2008), 574–594. https://doi.org/10.1080/10447310802205776 arXiv:https://doi.org/10.1080/10447310802205776

[3] Matthias Böhmer, Brent Hecht, Johannes Schöning, Antonio Krüger, and Gernot Bauer. 2011. Falling Asleep with Angry Birds, Facebook and Kindle: A Large Scale Study on Mobile Application Usage. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services* (Stockholm, Sweden) *(MobileHCI '11)*. ACM, New York, NY, USA, 47–56. https://doi.org/10.1145/2037373.2037383

[4] John Brooke. 2013. SUS: A Retrospective. *J. Usability Studies* 8, 2 (Feb. 2013), 29–40. http://dl.acm.org/citation.cfm?id=2817912.2817913

[5] Monica S Castelhano and Paul Muter. 2001. Optimizing the reading of electronic text using rapid serial visual presentation. *Behaviour & Information Technology* 20, 4 (2001), 237–247.

[6] John P. Chin, Virginia A. Diehl, and Kent L. Norman. 1988. Development of an Instrument Measuring User Satisfaction of the Human-computer Interface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Washington, D.C., USA) *(CHI '88)*. ACM, New York, NY, USA, 213–218. https://doi.org/10.1145/57167.57203

[7] Anne E Cunningham and Keith E Stanovich. 1998. What reading does for the mind. *American educator* 22 (1998), 8–17.

[8] Oscar de Bruijn and Robert Spence. 2002. Patterns of Eye Gaze during Rapid Serial Visual Presentation. In *Proceedings of the Working Conference on Advanced Visual Interfaces* (Trento, Italy) *(AVI '02)*. Association for Computing Machinery, New York, NY, USA, 209–217. https://doi.org/10.1145/1556262.1556295

[9] Tilman Dingler, Rufat Rzayev, Valentin Schwind, and Niels Henze. 2016. RSVP on the Go: Implicit Reading Support on Smart Watches Through Eye Tracking. In *Proceedings of the 2016 ACM International Symposium on Wearable Computers* (Heidelberg, Germany) *(ISWC '16)*. ACM, New York, NY, USA, 116–119. https://doi.org/10.1145/2971763.2971794

[10] Tilman Dingler, Rufat Rzayev, Alireza Sahami Shirazi, and Niels Henze. 2018. Designing Consistent Gestures Across Device Types: Eliciting RSVP Controls for Phone, Watch, and Glasses. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. ACM, New York, NY, USA, Article 419, 12 pages. https://doi.org/10.1145/3173574.3173993

[11] Tilman Dingler, Alireza Sahami Shirazi, Kai Kunze, and Albrecht Schmidt. 2015. Assessment of Stimuli for Supporting Speed Reading on Electronic Devices. In *Proceedings of the 6th Augmented Human International Conference* (Singapore, Singapore) *(AH '15)*. ACM, New York, NY, USA, 117–124. https://doi.org/10.1145/2735711.2735796

[12] Robert L Duchnicky and Paul A Kolers. 1983. Readability of text scrolled on visual display terminals as a function of window size. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 25, 6 (1983), 683–692. https://doi.org/10.1177/001872088302500605 arXiv:https://doi.org/10.1177/001872088302500605

[13] Joel E. Fischer, Chris Greenhalgh, and Steve Benford. 2011. Investigating Episodes of Mobile Phone Activity as Indicators of Opportune Moments to Deliver Notifications. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services* (Stockholm, Sweden) *(MobileHCI '11)*. Association for Computing Machinery, New York, NY, USA, 181–190. https://doi.org/10.1145/2037373.2037402

[14] Kenneth I Forster. 1970. Visual perception of rapidly presented word sequences of varying complexity. 8, 4 (01 Jul 1970), 215–221. https://doi.org/10.3758/BF03210208

[15] Tsvetozar Georgiev. 2012. Investigation of the User's Text Reading Speed on Mobile Devices. In *Proceedings of the 13th International Conference on Computer Systems and Technologies* (Ruse, Bulgaria) *(CompSysTech '12)*. ACM, New York, NY, USA, 329–336. https://doi.org/10.1145/2383276.2383324

[16] Luther C Gilbert. 1959. Speed of processing visual stimuli and its relation to reading. *Journal of Educational Psychology* 50, 1 (1959), 8.

[17] Saul Greenberg and Bill Buxton. 2008. Usability Evaluation Considered Harmful (Some of the Time). In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08)*. Association for Computing Machinery, New York, NY, USA, 111–120. https://doi.org/10.1145/1357054.1357074

[18] Wei Guo and Jingtao Wang. 2018. Towards Attentive Speed Reading on Small Screen Wearable Devices. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction* (Boulder, CO, USA) *(ICMI '18)*. ACM, New York, NY, USA, 278–287. https://doi.org/10.1145/3242969.3243009

[19] Christopher G. Healey, Kellogg S. Booth, and James T. Enns. 1996. High-Speed Visual Estimation Using Preattentive Processing. *ACM Trans. Comput.-Hum. Interact.* 3, 2 (June 1996), 107–135. https://doi.org/10.1145/230562.230563

[20] Björn Hedin and Erik Lindgren. 2007. A Comparison of Presentation Methods for Reading on Mobile Phones. *Distributed Systems Online, IEEE* 8, 6 (June 2007), 2–2. https://doi.org/10.1109/MDSO.2007.34

[21] Joyce Ho and Stephen S. Intille. 2005. Using Context-Aware Computing to Reduce the Perceived Burden of Interruptions from Mobile Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Portland, Oregon, USA) *(CHI '05)*. Association for Computing Machinery, New York, NY, USA, 909–918. https://doi.org/10.1145/1054972.1055100

[22] Nancy G Kanwisher. 1987. Repetition blindness: Type recognition without token individuation. *Cognition* 27, 2 (1987), 117–143.

[23] Kevin Lam and Robert Spence. 1997. *Image Browsing: a Space-Time Trade-off.* Springer US, Boston, MA, 611–612. https://doi.org/10.1007/978-0-387-35175-9_105

[24] Catherine C. Marshall and Christine Ruotolo. 2002. Reading-in-the-small: A Study of Reading on Small Form Factor Devices. In *Proceedings of the 2Nd ACM/IEEE-CS Joint Conference on Digital Libraries* (Portland, Oregon, USA) *(JCDL '02)*. ACM, New York, NY, USA, 56–64. https://doi.org/10.1145/544220.544230

[25] Michael E. J. Masson. 1983. Conceptual processing of text during skimming and rapid sequential reading. *Memory & Cognition* 11, 3 (01 May 1983), 262–274. https://doi.org/10.3758/BF03196973

[26] Abhinav Mehrotra, Robert Hendley, and Mirco Musolesi. 2016. PrefMiner: Mining User's Preferences for Intelligent Mobile Notification Management. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Heidelberg, Germany) *(UbiComp '16)*. Association for Computing Machinery, New York, NY, USA, 1223–1234. https://doi.org/10.1145/2971648.2971747

[27] Abhinav Mehrotra, Mirco Musolesi, Robert Hendley, and Veljko Pejovic. 2015. Designing Content-Driven Intelligent Notification Mechanisms for Mobile Applications. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Osaka, Japan) *(UbiComp '15)*. Association for Computing Machinery, New York, NY, USA, 813–824. https://doi.org/10.1145/2750858.2807544

[28] Dan R. Olsen. 2007. Evaluating User Interface Systems Research. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology* (Newport, Rhode Island, USA) *(UIST '07)*. Association for Computing Machinery, New York, NY, USA, 251–258. https://doi.org/10.1145/1294211.1294256

[29] Martin Pielot, Karen Church, and Rodrigo de Oliveira. 2014. An In-situ Study of Mobile Phone Notifications. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services* (Toronto, ON, Canada) *(MobileHCI '14)*. ACM, New York, NY, USA, 233–242. https://doi.org/10.1145/2628363.2628364

[30] Martin Pielot, Amalia Vradi, and Souneil Park. 2018. Dismissed!: A Detailed Exploration of How Mobile Phone Users Handle Push Notifications. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services* (Barcelona, Spain) *(MobileHCI '18)*. ACM, New York, NY, USA, Article 3, 11 pages. https://doi.org/10.1145/3229434.3229445

[31] Mary C Potter and Ellen I Levy. 1969. Recognition memory for a rapid sequence of pictures. *Journal of experimental psychology* 81, 1 (1969), 10.

[32] Jane E Raymond, Kimron L Shapiro, and Karen M Arnell. 1992. Temporary suppression of visual processing in an RSVP task: An attentional blink? *Journal of experimental psychology: Human perception and performance* 18, 3 (1992), 849.

[33] Keith Rayner, Barbara R Foorman, Charles A Perfetti, David Pesetsky, and Mark S Seidenberg. 2001. How psychological science informs the teaching of reading. *Psychological science in the public interest* 2, 2 (2001), 31–74.

[34] Gary S. Rubin and Kathleen Turano. 1992. Reading without saccadic eye movements. *Vision Research* 32, 5 (1992), 895–902. https://doi.org/10.1016/0042-6989(92)90032-E

[35] Rufat Rzayev, Paweł W. Woźniak, Tilman Dingler, and Niels Henze. 2018. Reading on Smart Glasses: The Effect of Text Position, Presentation Type and Walking. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. ACM, New York, NY, USA, Article 45, 9 pages. https://doi.org/10.1145/3173574.3173619

[36] Alireza Sahami Shirazi, Niels Henze, Tilman Dingler, Martin Pielot, Dominik Weber, and Albrecht Schmidt. 2014. Large-scale Assessment of Mobile Notifications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. ACM, New York, NY, USA, 3055–3064. https://doi.org/10.1145/2556288.2557189

[37] Lauren FV Scharff, Alyson L Hill, and Albert J Ahumada. 2000. Discriminability measures for predicting readability of text on textured backgrounds. *Optics express* 6, 4 (2000), 81–91.

[38] Elizabeth R Schotter, Randy Tran, and Keith Rayner. 2014. Don't Believe What You Read (Only Once) Comprehension Is Supported by Regressions During Reading. *Psychological science* (2014), 0956797614531148. https://doi.org/10.1177/0956797614531148 arXiv:https://doi.org/10.1177/0956797614531148

[39] Ben Shneiderman. 1987. *User interface design and evaluation for an electronic encyclopedia.* University of Maryland.

[40] R Spence. 1998. A content explorer. *Information Engineering Report* 98, 08 (1998).

[41] Robert Spence. 2002. Rapid, serial and visual: a presentation technique with potential. *Information visualization* 1, 1 (2002), 13–19.

[42] Benjamin Tag, Andrew W Vargo, Aman Gupta, George Chernyshov, Kai Kunze, and Tilman Dingler. 2019. Continuous Alertness Assessments: Using EOG Glasses to Unobtrusively Monitor Fatigue Levels In-The-Wild. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, New York, NY, USA, 464:1—-464:12. https://doi.org/10.1145/3290605.3300694

[43] T Tse, G Marchionini, W Ding, L Slaughter, and A Komlodi. 1998. Dynamic Key-frame presentation techniques for augmented video browsing (pp. 185–194). In *ACM, Proceedings of Conference on AVI.*

[44] Thomas S Tullis and Jacqueline N Stetson. 2004. A comparison of questionnaires for assessing website usability. In *Usability professional association conference*, Vol. 1. Minneapolis, USA.

[45] Liam D. Turner, Stuart M. Allen, and Roger M. Whitaker. 2019. The influence of concurrent mobile notifications on individual responses. *International Journal of Human-Computer Studies* 132 (2019), 70–80. https://doi.org/10.1016/j.ijhcs.2019.07.011

[46] Dominik Weber, Alexandra Voit, Huy Viet Le, and Niels Henze. 2016. Notification Dashboard: Enabling Reflection on Mobile Notifications. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct* (Florence, Italy) *(MobileHCI '16)*. Association for Computing Machinery, New York, NY, USA, 936–941. https://doi.org/10.1145/2957265.2962660

[47] Kent Wittenburg, Wissam Ali-Ahmad, Daniel LaLiberte, and Tom Lanning. 1998. Rapid-Fire Image Previews for Information Navigation. In *Proceedings of the Working Conference on Advanced Visual Interfaces* (Lapos;Aquila, Italy) *(AVI '98)*. Association for Computing Machinery, New York, NY, USA, 76–82. https://doi.org/10.1145/948496.948508

[48] Kent Wittenburg, Carlos Chiyoda, Michael Heinrichs, and Tom Lanning. 1999. Browsing through rapid-fire imaging: requirements and industry initiatives. In *Internet Imaging*, Vol. 3964. International Society for Optics and Photonics, 48–56.

[49] K. Wittenburg, J. Nicol, J. Paschetto, and C. Martin. 1999. Browsing with dynamic key frame collages in Web-based entertainment video services. In *Proceedings IEEE International Conference on Multimedia Computing and Systems*, Vol. 2. 913–918 vol.2. https://doi.org/10.1109/MMCS.1999.778610