Reading Scheduler: Proactive Recommendations to Help Users Cope with Their Daily Reading Volume

Tilman Dingler
University of Melbourne
tilman.dingler@unimelb.edu.au

Benjamin Tag
Keio University
tagbenja@kmd.keio.ac.jp

Sabrina Lehrer
University of Stuttgart
s.lehrer@gmx.de

Albrecht Schmidt
LMU Munich
albrecht.schmidt@ifi.lmu.de

ABSTRACT
To help deal with daily reading volumes, we present *Reading Scheduler*, a smartphone application linked to people’s reading list, which triggers reading reminders throughout the day. The app suggests articles according to their length, complexity, and the time available for reading as indicated by the user. In a field study, we collected usage data from ten participants over the course of two weeks. During this time, we recorded mobile sensor data and trained a classifier to detect opportune moments for reading. Participants read 182 articles while we collected 787,752 sensor data points. Together with an assessment of the feasibility of proactive reading suggestions, we present a prediction model with close to 73% accuracy, that can be used to build mobile recommender systems for utilizing idle moments for reading throughout the user’s day.

CCS Concepts
•Human-centered computing → **Human computer interaction (HCI); •Computing methodologies** → *Machine learning;*

Author Keywords
Mobile Reading; Reading Scheduler; Reading Context.

MOTIVATION AND BACKGROUND
Life in our knowledge society revolves around effectively managing and consuming information. Mobile devices and their near-constant connectivity allow us to keep up with the information flow anytime and anywhere. While browsing the web at home, at work or on-the-go, we often come across new texts and articles, which we may not be able to read directly at the time we encounter them. To manage such items people often use bookmarks and reading lists [15]. Common tools and ways of bookmarking include online services, such as *Pocket*, or bibliography software, such as *Mendeley*. Systems

MUM ’18, November 25–28, 2018, Cairo, Egypt

DOI: https://doi.org/10.1145/3282894.3282917

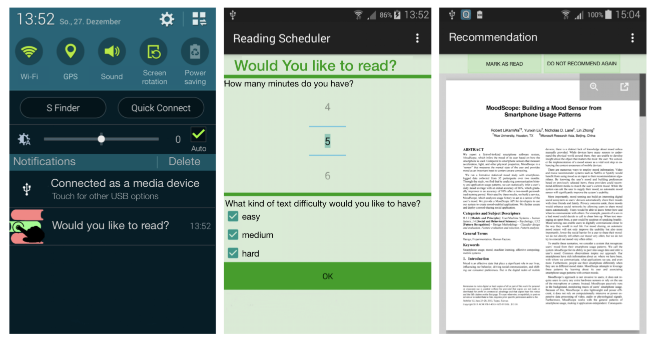

Figure 1. Screenshots of *Reading Scheduler*, an Android app that proactively triggers reminders in the form of notifications (left) to read articles from users’ reading lists throughout the day (right). A quick situation assessment (center) at the beginning of each reading session is matched with the articles available.

have even been proposed, which create and curate reading lists automatically [7, 21]. New items, therefore, often pile up quicker than stored items are taken off, hence contributing to a constant growth of our reading lists. As reading backlogs grow in this way, time and attention throughout the day remain generally limited [17]. While some may reserve dedicated time slots for reading, an increasing number of people spread their readings throughout the day using electronic devices [1]. Pielot *et al.* [16] showed that there is a sufficient number of idle gaps throughout the day, where people tend to be open to stimuli in the form of content suggestions. However, there is a dichotomy between people seeking engagement, for example, in times of boredom [6], *vs.* people feeling overwhelmed by interruptions triggered by smartphones [12, 20].

To help users identify idle times and work on their reading lists throughout the day, we developed the smartphone application *Reading Scheduler*. It accesses people’s online reading lists, triggers reading reminders throughout the day, and matches available time to articles. Because smartphones are within arm’s reach for major parts of the day [4], a number of reading apps have been developed that, for example, target specific groups, such as people with Dyslexia [18]. Others employ classifiers to identify people’s reading type to adjust the text interface to support their reading style [3]. With the goal of identifying context factors for opportune moments for suggesting reading materials, we developed our mobile app, which suggests items from the user's reading list throughout the day according to the current time available and preferred text difficulty. In a user study with ten participants, we collected log and sensors data over the course of two weeks. From this data, we extracted 34 features and trained classifiers to be able to identify opportune reading moments throughout the day. By detecting these moments, reminders in the form of notifications can be triggered in a more targeted way, which has been shown to reduce people's frustration with being overwhelmed by notifications [11] and can lead to faster response rates [8]. Adaptive notification systems based on users' preferences and real-time context factors, such as time and location, have been shown to increase the likelihood of engagement [9, 13]. In this paper, we 1) present *Reading Scheduler*, a mobile phone app that suggests items from users' reading lists throughout the day and 2) discuss context factors that contribute to users' openness to those reading suggestions, based on which we present a classifier to detect opportune moments for reading.

METHOD

To attain context data with which to train a prediction model, but also to collect initial user feedback, we conducted a user study, during which we invited users to install our app for a period of two weeks.

Apparatus

We designed and developed *Reading Scheduler* as an Android app with the goal of proactively triggering reading reminders throughout the day matching time available to corresponding articles. It runs on Android devices with OS version 4.3 (API version 18) or higher and connects to users' *Pocket*[1] account. *Pocket* provides an API and comes with a browser plugin, through which users can add articles and manage their reading lists. *Reading Scheduler* proactively downloads articles from the user's reading list and makes them available even when the device is offline. For web pages, *Pocket* trims the articles by automatically removing ads, menu items, and banners. On its first launch, the app presents a consent form and informs users about the data that is being collected (see Table 1). Throughout the day (between 8am and 11pm), the app triggers a maximum of six reading reminders in the form of notifications spread out randomly, but evenly, throughout the day (see Figure 1, *left*). When a notification is clicked, a quick assessment is shown (see Figure 1, *center*) asking the user about the amount of time available and the desired difficulty of the text. Such in-situ assessments have been widely used to collect information on user context, also called Ecological Momentary Assessment (EMA) [2]. Based on these assessments and considering an average reading speed of 250 words per minute (wpm), the app selects an article from the reading list taking into account the article's word count and the text's difficulty calculated by the Automated Readability Index (ARI) as proposed by Senter and Smith [19]. If no article matches any of the criteria, a random article is selected from the reading list. App usage and sensor data are locally stored on the device and sent over a secure connection to our server when the user connects to a WiFi network.

[1] https://getpocket.com/

Phone Context	
ringer mode	mute, vibration, ringer
charging mode	unplugged, charging
battery status	related to usage intensity
display orientation	portrait or landscape mode, orientation changes
light sensor	changes in lightness allow us to derive whether phone is covered (carried in a pocket or taken out)
proximity sensor	phone in pocket
semantic location	GPS data allows the inference of locations visited, and in-place vs. on-the-go states
linear motion	motion sensor, change of position
Phone Usage	
Calls	incoming, outgoing
SMS	incoming, outgoing
Notifications	received, dismissed, ignored, interacted with
Unlocks	phone unlocks
Data usage	upload/download
Applications	applications in foreground, switches, usage duration

Table 1. List of sensor data collected by *Reading Scheduler*.

Procedure

We recruited ten participants (4 female) with a mean age of 25 ($SD = 5$) years through personal networks and university mailing lists. Participants' backgrounds ranged from computer science students to IT and business professionals, all of which indicated German to be their mother tongue. For the two weeks long participation, we gave out 20 EUR to everyone completing the study. We invited participants to the lab for an initial briefing, where we introduced them to the purpose of the study, the extent of the data collection, and had them sign a consent form. We then installed the app on their personal phones and helped them to register a *Pocket* account in case they were new to the service. For successfully completing the study we asked participants to use the app on a regular basis over the course of two weeks. This entailed reading at least one article per day and adding at least two articles to their reading list per week (popular science articles from German newspapers and magazines). At the end of the study, we administered a final survey to collect qualitative user feedback.

RESULTS

In total, participants read 182 ($M = 18.2, SD = 7.9$) articles which constitutes to a compliance rate of 39.6%. Participants engaged in on average 18 ($SD = 7.9$) reading sessions each and read for for 7.7 minutes ($SD = 5.8$) per session. We collected 787,752 sensor data points, on average 79,023 ($SD = 34,646$) per participant. Before selecting an article the app asked users how much time was available for reading (see Figure 1, middle), for which our participants selected on average 7.4 minutes ($SD = 5.9, Mode = 5$). Figure 2 plots the available time windows participants indicated by the hour of day. Average minutes available for reading per participants lied between 4.8 and 12.4 minutes. As for explicitly filtering texts by difficulty, participants barely used this feature and left all boxes checked (easy, medium, hard) in 87.5% of the cases. Of all the articles completely read by users, 84% were finished in less than the time initially indicated as being available meaning that either users had a tendency to underestimate their reading speed or our matching algorithm tended to overestimate a text's difficulty. In 72% of the cases, the algorithm retrieved an article matching the user's criteria, in the remaining 28% of cases the app chose a random article when no match could be found.

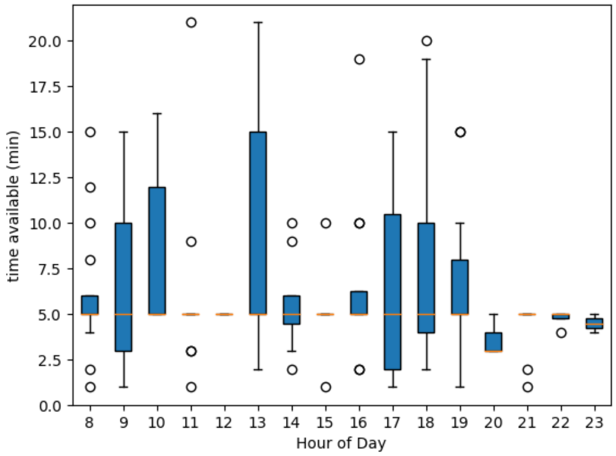

Figure 2. Whisker plots of participants’ available time slots for reading across the day. Orange lines depict the median number of minutes. Outliers were cut off at 25 minutes.

OPPORTUNE MOMENTS FOR READING

Figure 3 shows the distribution of notifications and articles read throughout the day. Peaks at 8am, 5pm, and 7pm (green bars) fall into typical commute times suggesting that articles were often read while in transit corroborating findings by Dingler *et al.* [5]. This hypothesis is supported by analyzing the semantic locations as collected by the phone’s location data and made sense of through frequency distributions across the day: 44.1% of app usages took place at home, 12.1% at work, and 43.8% were classified as ’other locations’.

Based on the sensor data collected and participants’ demographics, we extracted 34 features (similar to [16]). The features can roughly be split into two categories: 1) **context** comprising information about the general phone state, such as battery status, ringer mode, or time and 2) **phone usage** covering user and phone activities related to communication, *e.g.*, time since last phone call or incoming notification, and features related to phone and app usage, such as phone unlocks, app switches, or data usage. To be able to predict whether an article suggestion would be accepted and, therefore, result in a reading session, we trained a model that would classify predicted states of reading (*i.e.* accepted reminders) and not reading (*i.e.* dismissed and ignored reminders). Data about triggered, dismissed, ignored, or accepted notifications served as ground truth. For analyzing the phone’s state before each ground truth collection, we looked at different time windows, namely 1, 3, 5, 10 and 15 minutes. After initial data exploration we focused on a 5 minutes time window, which reflected recent phone activities and resulted in a sufficient amount of data for predicting reading receptivity. For training and evaluating our models, we applied a 10-fold cross-validation with a 90/10 split of training and test data. We used the *Weka* Software [10] to train two classifiers: a Bayesian Network and a Sequential Minimal Optimization (SMO), *i.e.*, a Support Vector Machine. The Bayesian network achieved a precision of 0.460, a recall of 0.729, and an F-score of 0.541. The SMO performed slightly worse with a precision of 0.390, a recall of 0.706, and an F-score of 0.502. Figure 4 depicts the Receiver Operating Characteristic (ROC) curve of the Bayesian Network as its discrimination threshold is varied. It illustrates the model’s relationship between recall and error rate as compared to a purely random classifier (grey line). Red lines represent plots based on a subset ($N = 5$, half of our participants) of the user-dependent models, whereas the blue depicts the user-independent model.

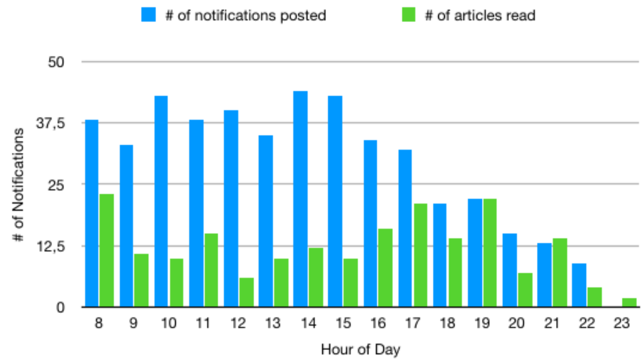

Figure 3. Distribution of notifications triggered and accepted throughout the day.

For analyzing the feature ranking and their expressiveness for predicting users’ receptivity for reading suggestions, we first applied the SMO classifier to each participant’s individual data set. For generalizing the results, we took each feature’s influence and calculated the average of each weight. We only took into account features that were present in more than half of participants’ data sets. Some more specific applications or app categories did not show in all participants’ datasets. Time since last phone unlock turned out to be the most predictive feature ($importance = 0.535, correlation = -0.057$). The negative correlation value indicates that users were more receptive to reading suggestions the less time had passed since the last phone unlock. Another highly predictive feature was the usage of the communication app *WhatsApp* ($importance = 0.411, correlation = -0.130$). When users were engaged in communication, they were less likely to accept a reading notification. The more apps were used in the last 5 minutes, the more likely users were open to reading suggestions ($importance = 0.329, correlation = 0.068$), which is in line with idle states of doodling around with the phone shown by users jumping between apps [16]. Other prominent features seem to be time-related, such as the day of the week ($importance = 0.327, correlation = -0.047$) and the hour of the day ($importance = 0.223, correlation = -0.023$). Data usage ($importance = 0.247, correlation = 0.025$), user movement ($importance = 0.240, correlation = 0.020$) and lighting ($importance = 0.231, correlation = 0.016$), *i.e.* presumably phone out of pocket and during the day, are weakly positive indicators for predicting a user’s openness towards reading reminders.

To collect **subjective feedback** from all participants we used a combination of intermittent surveys, triggered by the app, and semi-structured interviews conducted after the study was completed. Throughout the two weeks, the app triggered surveys every three days. Each survey consisted of four five point Likert-style scale statements (1=strongly agree, 5=strongly disagree) . The obtained results show rather neutral opinions

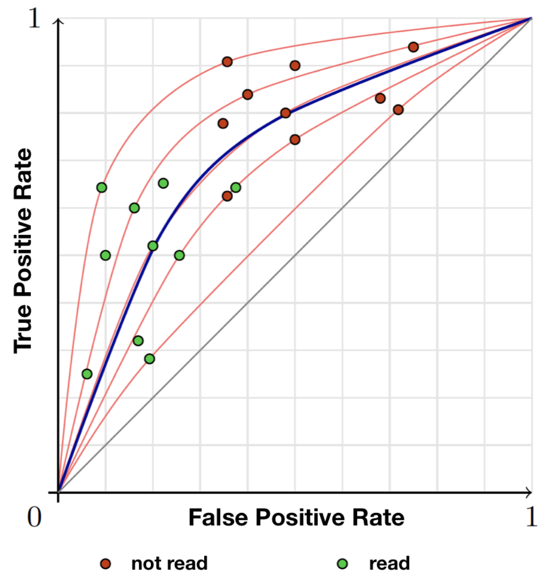

Figure 4. Depicts the ROC room for the classification via Bayesian Network. The red curves depict single participants, blue represents the user-independent model.

for the following three statements:"*I like being reminded to read in regular intervals*" ($Mdn = 3$ ($SD = 1.9$)) , "*The app helps me to read more*" ($Mdn = 3$ ($SD = 0.9$)), and "*I want to keep using this app in the future*" ($Mdn = 3$ ($SD = 1.3$)). Strong agreement was expressed for "*I like the idea behind this app*" ($Mdn = 1$ ($SD = 1.3$)).

Additionally, we conducted a semi-structured interview with each participant after using *Reading Scheduler* for two weeks. Most participants expressed appreciation for being motivated to spend idle moments reading and having a central place to collect their readings: "*The idea to combine articles from any source in one app increases the reading comfort.*" (P1). Reading behaviors, however, seemed to differ significantly between individuals. Two participants indicated that they would have spent their commute reading anyways and felt rather forced to use the smartphone during the course of the study instead of other means and that they did not really need to be reminded to read in general. The remaining participants ($N = 8$) said that while they were on their phone, reading was just one option for them to spend their time. It sometimes competed with other activities such as social media usage, surfing the Web, and playing games. Nevertheless, these users stressed that the reading reminders helped them spend their commute in a more productive way.

DISCUSSION AND LIMITATIONS

In our debriefings, participants reported to often choose their reading material depending on a number of factors, such as time of day, time available, type of text, but also on current mood and alternative entertainment options. *Reading Scheduler* allowed us to collect more information on context from phone sensor data and participants' subjective assessments. Study participants predominantly used it in the hours before and after work and rarely read for more than 12 minutes on it, which suggests that commuting time yields opportune windows for engaging with reading material. As our participants' feedback showed, this is also the time during which people chose to browse social media and play games. A targeted reading reminder in such moments, however, can nudge users to balance their time spent doodling with their phone *vs.* engaging in more productive activities. Participants further reported on how the day of the week and the time of the day influenced their reading behavior, an intuition our feature ranking confirmed. The earlier during the day, the more likely participants tended to accept reading suggestions. The same accounts for earlier days during the week, which confirms previous work stating that general focus is higher earlier of the week and during the day [14]. *Reading Scheduler* stopped, however, triggering notifications after six notifications had been clicked in one day. This explains the decline of notifications and consequently of reading sessions as the day progressed (see Figure 3), but may not represent peoples' refusal to engage with readings at later hours of the day. We were, however, mainly interested in opportune reading moments while people go about their day rather than dedicated reading sessions at night. This also gave rise to one of the major criticisms voiced by participants, namely that their phone might not be their preferred medium for reading certain texts. By asking participants to use the app for a dedicated amount of time might have shifted their habits only temporarily. A longer study conducted in-the-wild, *e.g.*, by releasing *Reading Scheduler* in an App Store will produce more data regarding reading sessions on-the-go. The most common drawback of the app voiced by participants concerns the online management of reading material, which naturally does not cover paper-based articles. For a reading scheduler to be comprehensive, offline media needs to be taken into account as well by, for example, linking paper-based text to resources available online (*e.g.*, the *Kindle* version of a book). The current *Reading Scheduler* was further unable to parse PDFs in the user's reading list, and therefore could not calculate the expected reading duration, nor text difficulty. Hence, PDFs were only recommended in case none of the existing articles fit the user's selection criteria.

CONCLUSION

Reading Scheduler–a mobile app that triggers proactive reading recommendations–was designed to help users keep up with their daily reading volume. We investigated text selection strategies and moments in which users can be engaged with reading content. By training a machine-learning model we were able to predict such opportune moments for reading with an accuracy of close to 73%. Recent phone unlocks and app usage can be strong indicators for users' receptivity, but disrupting communications should be avoided. Proactive recommendations especially appeal to people who have strong intentions to deal with their readings, but lack innate drive. Context-aware reminders can constitute an effective means to help people deal with their readings during opportune times throughout the day.

ACKNOWLEDGMENTS

This work is supported by the Future and Emerging Technologies of the European Commission under grant number 612933 and by JST CREST under grant number JPMJCR16E1.

REFERENCES

1. Annette Adler, Anuj Gujar, Beverly L. Harrison, Kenton O’Hara, and Abigail Sellen. 1998. A Diary Study of Work-related Reading: Design Implications for Digital Reading Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI ’98)*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 241–248. DOI: http://dx.doi.org/10.1145/274644.274679
2. Sunny Consolvo and Miriam Walker. 2003. Using the experience sampling method to evaluate ubicomp applications. *IEEE Pervasive Computing* 2, 2 (2003), 24–31.
3. Marios Constantinides, John Dowell, David Johnson, and Sylvain Malacria. 2015. Exploring Mobile News Reading Interactions for News App Personalisation. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI ’15)*. ACM, New York, NY, USA, 457–462. DOI: http://dx.doi.org/10.1145/2785830.2785860
4. Anind K Dey, Katarzyna Wac, Denzil Ferreira, Kevin Tassini, Jin H 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*. 163–172. DOI: http://dx.doi.org/10.1145/2030112.2030135
5. Tilman Dingler, Dominik Weber, Martin Pielot, Jennifer Cooper, Chung-Cheng Chang, and Niels Henze. 2017. Language Learning On-The-Go: Opportune Moments and Design of Mobile Microlearning Sessions. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI ’17)*. ACM, New York, NY, USA. DOI:http://dx.doi.org/10.1145/3098279.3098565
6. John D Eastwood, Alexandra Frischen, Mark J Fenske, and Daniel Smilek. 2012. The unengaged mind defining boredom in terms of attention. *Perspectives on Psychological Science* 7, 5 (2012), 482–495.
7. Michael D. Ekstrand, Praveen Kannan, James A. Stemper, John T. Butler, Joseph A. Konstan, and John T. Riedl. 2010. Automatically Building Research Reading Lists. In *Proceedings of the Fourth ACM Conference on Recommender Systems (RecSys ’10)*. ACM, New York, NY, USA, 159–166. DOI: http://dx.doi.org/10.1145/1864708.1864740
8. 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 (MobileHCI ’11)*. ACM, New York, NY, USA, 181–190. DOI: http://dx.doi.org/10.1145/2037373.2037402
9. Qi Guo, Eugene Agichtein, Charles LA Clarke, and Azin Ashkan. 2009. In the Mood to Click? Towards Inferring Receptiveness to Search Advertising. In *2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, Vol. 1. 319–324. DOI: http://dx.doi.org/10.1109/WI-IAT.2009.368
10. Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software: An Update. *SIGKDD Explor. Newsl.* 11, 1 (Nov. 2009), 10–18. DOI: http://dx.doi.org/10.1145/1656274.1656278
11. Shamsi T. Iqbal and Brian P. Bailey. 2008. Effects of Intelligent Notification Management on Users and Their Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI ’08)*. ACM, New York, NY, USA, 93–102. DOI: http://dx.doi.org/10.1145/1357054.1357070
12. Shamsi T. Iqbal and Eric Horvitz. 2010. Notifications and Awareness: A Field Study of Alert Usage and Preferences. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work (CSCW ’10)*. ACM, New York, NY, USA, 27–30. DOI: http://dx.doi.org/10.1145/1718918.1718926
13. Hugo Lopez-Tovar, Andreas Charalambous, and John Dowell. 2015. Managing Smartphone Interruptions Through Adaptive Modes and Modulation of Notifications. In *Proceedings of the 20th International Conference on Intelligent User Interfaces (IUI ’15)*. ACM, New York, NY, USA, 296–299. DOI: http://dx.doi.org/10.1145/2678025.2701390
14. Gloria Mark, Shamsi T. Iqbal, Mary Czerwinski, and Paul Johns. 2014. Bored Mondays and Focused Afternoons: The Rhythm of Attention and Online Activity in the Workplace. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI ’14)*. ACM, New York, NY, USA, 3025–3034. DOI: http://dx.doi.org/10.1145/2556288.2557204
15. Jennifer Pearson, George Buchanan, and Harold Thimbleby. 2011. The Reading Desk: Applying Physical Interactions to Digital Documents. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI ’11)*. ACM, New York, NY, USA, 3199–3202. DOI: http://dx.doi.org/10.1145/1978942.1979416
16. Martin Pielot, Tilman Dingler, Jose San Pedro, and Nuria Oliver. 2015. When Attention is Not Scarce - Detecting Boredom from Mobile Phone Usage. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp ’15)*. ACM, New York, NY, USA, 825–836. DOI: http://dx.doi.org/10.1145/2750858.2804252
17. Michael I. Posner. 1982. Cumulative development of attentional theory. *American Psychologist* 37, 2 (1982), 168–179. DOI: http://dx.doi.org/10.1037/0003-066X.37.2.168

18. Luz Rello and Ricardo Baeza-Yates. 2014. Evaluation of DysWebxia: A Reading App Designed for People with Dyslexia. In *Proceedings of the 11th Web for All Conference (W4A '14)*. ACM, New York, NY, USA, Article 10, 10 pages. DOI: http://dx.doi.org/10.1145/2596695.2596697

19. RJ Senter and EA Smith. 1967. *Automated readability index*. Technical Report. DTIC Document.

20. SungHyuk Yoon, Sang-su Lee, Jae-myung Lee, and KunPyo Lee. 2014. Understanding notification stress of smartphone messenger app. In *Proceedings of the extended abstracts of the 32nd annual ACM conference on Human factors in computing systems - CHI EA '14 (CHI EA '14)*. ACM, New York, NY, USA, 1735–1740. DOI:http://dx.doi.org/10.1145/2559206.2581167

21. Bo-Wen Zhang, Xu-Cheng Yin, Fang Zhou, and Jian-Lin Jin. 2017. Building Your Own Reading List Anytime via Embedding Relevance, Quality, Timeliness and Diversity. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. ACM, New York, NY, USA, 1109–1112. DOI: http://dx.doi.org/10.1145/3077136.3080734