Attelia: Sensing User's Attention Status on Smart Phones

Tadashi Okoshi
Graduate School of Media and Governance, Keio University
5322 Endo, Fujisawa
Kanagawa, 252-0882 JAPAN
slash@ht.sfc.keio.ac.jp

Jin Nakazawa
Faculty of Environment and Information Studies, Keio University
5322 Endo, Fujisawa
Kanagawa, 252-0882 JAPAN
jin@ht.sfc.keio.ac.jp

Hideyuki Tokuda
Faculty of Environment and Information Studies, Keio University
5322 Endo, Fujisawa
Kanagawa, 252-0882 JAPAN
hxt@ht.sfc.keio.ac.jp

UbiComp'14 Adjunct, September 13 – 17, 2014, Seattle, WA, USA
ACM 978-1-4503-3047-3/14/09.
http://dx.doi.org/10.1145/2638728.2638802

Abstract
In progressing ubiquitous computing where number of devices, applications and the web services are ever increasing, human user's attention is a new bottleneck in computing. This paper proposes Attelia, a novel middleware that senses user's attention status on user's smart phones in real-time, without any dedicated psycho-physiological sensors. To find better delivery timings of interruptive notifications from various applications and services to mobile users, Attelia detects breakpoint[16] of user's activity on the smart phones, with our novel "Application as a Sensor"(AsaS) approach and machine learning technique. Our initial evaluation of Attelia shows it can detect breakpoints of users with accuracy of 80–90%.

Author Keywords
interruption; notification; user attention; mobile sensing; smart phones, machine learning

ACM Classification Keywords
H.3.4 [User profiles and alert services]: Systems and Software

Introduction
"Interruption" for users by notifications in multi-tasking environment has been a greater problem since amount of

user's "attention" remains unchanged while amount of information provided has been increasing in emerging ubiquitous computing age.

The number of versatile networked devices, including user's carrying mobile devices, embedded sensors, or so-called "IoT" devices have been ever increasing[9, 6]. Also the number of used applications, web services, and communication channels for users are increasing based on the technological advancement of rapid software/service development, deployment, and distribution.

Given such backgrounds, limited resource of **human attention** is the new bottleneck[7] in computing. From the view point of the human users, these excessive amount of provided information is often called "information overload" in a broad sense. Particularly in this research, we will focus on **interruption overload**, distraction for users caused by interruptions based on excessive amount and inappropriate delivery of notifications from computing systems.

Interruption Overload

The main cause of interruption overload for users is "notifications" from computing systems. Typical notification systems that deliver notification immediately make various negative affect to user's work productivity and even their emotion according to previous studies[1, 2, 5, 14, 8, 4].

Reflecting the recent trends in ubiquitous computing described in Introduction, there are also several distinctive characteristics observed in the recent notifications.

- **Increasing diversity in types and sources of notifications:** e.g., updates from friends over social network, query of participatory sensing[3],
- **Multiple mobile devices of users as destinations:** e.g., smart phone, tablet, or even wearable devices[9, 6].
- **Wider range of urgency level:** e.g., Early Earth Warning (EEW)[11] to which users need to physically react in a few seconds.
- **All-day-long interruption situation:** User's recent life style always with mobile devices makes interruption overload all day long.

Adaptive handling support of such notifications, including dynamic adaptation of notification timing, media, or content according to both current attention status of user and information to be notified, is clearly needed to ease user's interruption overload. Especially in the given situation, following are the distinctive requirements for such support.

- **Feasibility in mobile devices:** Users carry and use mobile devices, such as smart phones or tablets, as immediate devices for their task applications. Thus the system needs to fit the mobile platform, in terms of energy-efficiency, for example.
- **Real-time sensing:** Toward the adaptation on the fly, the sensing needs to be done in real-time.
- **Applicability for diverse types of notification source:** System needs to be applicable and easy to be deployed for diverse types of notification source.
- **Affinity with all-day-long use:** The sensing needs to be done all day long as long as the user's surrounding notification system is available.

Design of Attelia

Towards the realization of such adaptive notification support, we propose Attelia, a novel middleware that senses user's attention status on user's smart phones.

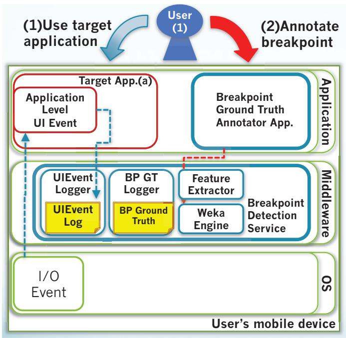

Figure 1: System Architecture of Attelia on Android Platform

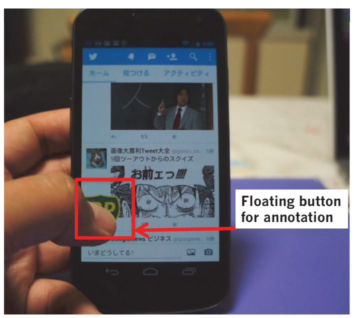

Figure 2: Ground Truth Annotation with Attelia

Attelia detects **"breakpoint"**[16], a concept originally found in psychology field, as a temporal target in sensing appropriate timing for interruption. Breakpoint is a boundary between two adjacent actions that human's perceptual system can segment inside user's goal-directed activity. Related researches have shown that deferral of notifications until users' sensed breakpoints reduces interruption cost in terms of resumption lag and subjective frustration value[1, 13, 12]. Rather than sensing cognitive load with psycho-physiological sensors[10], we take an approach to sense coarse-grained but easy-to-measure indicator from which eventually appropriate timing for notification can be inferred, only with user's existing carrying devices.

As the concrete approach for breakpoint detection, Attelia introduces a novel **"Application as a Sensor" (AsaS) approach** where user's application usage pattern takes the role of sensor as well as the machine learning technique.

Using applications on smart phones as a sensors and running on the user's smart phone, Attelia achieves **affinity with mobile devices**. Using machine learning technique, Attelia achieves its **real-time sensing** capability. Being under the applications as middleware, and inputting UI events from the application currently manipulated, Attelia achieves its **applicability to diverse applications** and easy deployment, without any modification to the applications. Because Attelia does not need any dedicated sensors, such as psycho-physiological sensors, it fits nicely the all-day-long use case by users.

Attelia's System Architecture

Figure 1 shows the system architecture of Attelia implemented on the Android 4 platform. Attelia consists of an Android service including UIEventLogger, BreakPointLogger, FeatureExtractor, and Weka[15] machine learning engine, as well as the GroundTruthAnnotator application, and off-line components for model training.

Attelia service is implemented as a "Service" inside the Android platform. Appropriately setting the permission of the service, without "rooting", it can input and log the stream of UI events, such as tapping, clicking, scrolling or modifications of UI components inside the currently-active Android application. Also, this implementation enables the service itself to be distributed through GooglePlay store and contributes to deployability of the system.

In the ground truth collection phase, ground truth of the breakpoint timings during application usage are collected via voluntary user's manual annotation. Figure 2 shows the screen shot with our Annotation application floating on the screen. While manipulating ordinary Android applications, the user pushes the floating button at the timing of their breakpoints. The Attelia service keeps recording the stream of UI events (excluding those from the annotation button) and breakpoint timestamps (timings that the annotation button was pushed) to the local storage. In the off-line model training phase, 45 defined features shown in Figure3, will be extracted for each time frame (three seconds duration) and will be input to Weka to train a classifier model. Finally in the testing phase, the UI event stream captured by the service will be input to the model in real-time on the mobile device, and the service detects breakpoints on the fly.

Evaluation

As our initial evaluation, we conducted a controlled user study among ten participants on six representative Android applications, and evaluated how accurately

Statistics on events' timing in the frame
TimeGap_min
TimeGap_mean
TimeGap_max
TimeGap_stdev

Statistics on the status of the event source UI components
isEnabled_rate
isChecked_rate
isPassword_rate

Statistics on the location of the event source UI components on the screen
XLeft_min
XLeft_max
XRight_min
XRight_max
YTop_min
YTop_max
YBottom_min
YBottom_max
XLeft_InSmallestRect
XRight_InSmallestRect
XWidth_InSmallestRect
YTop_InSmallestRect
YBottom_InSmallestRect
YHeight_InSmallestRect

Rate of occurrences of different UI event types
TYPE_VIEW_CLICKED_rate
TYPE_VIEW_LONG_CLICKED_rate
TYPE_VIEW_SELECTED_rate
TYPE_VIEW_FOCUSED_rate
TYPE_VIEW_TEXT_CHANGED_rate
TYPE_WINDOW_STATE_CHANGED_rate
TYPE_NOTIFICATION_STATE_CHANGED_rate
TYPE_VIEW_HOVER_ENTER_rate
TYPE_VIEW_HOVER_EXIT_rate
TYPE_TOUCH_EXPLORATION_GESTURE_START_rate
TYPE_TOUCH_EXPLORATION_GESTURE_END_rate
TYPE_WINDOW_CONTENT_CHANGED_rate
TYPE_VIEW_SCROLLED_rate
TYPE_VIEW_TEXT_SELECTION_CHANGED_rate
TYPE_ANNOUNCEMENT_rate
TYPE_VIEW_ACCESSIBILITY_FOCUSED_rate
TYPE_VIEW_ACCESSIBILITY_FOCUS_CLEARED_rate
TYPE_VIEW_TEXT_TRAVERSED_AT_MOVEMENT_GRANULARITY_rate
TYPE_GESTURE_DETECTION_START_rate
TYPE_GESTURE_DETECTION_END_rate
TYPE_TOUCH_INTERACTION_START_rate
TYPE_TOUCH_INTERACTION_END_rate

Figure 3: Features used in Attelia

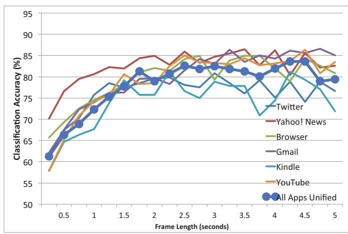

Figure 4: Cross-validation result with J48

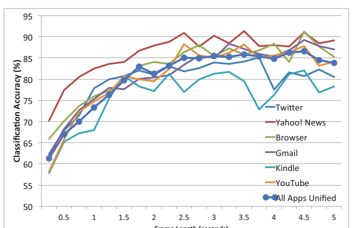

Figure 5: Cross-validation result with RandomForest

Attelia can detect breakpoints. The participants were university workers, students, and staffs with at least intermediate knowledge on the Android smart phones. The applications were (1)Twitter, (2)Yahoo! news, (3)Browser, (4)Gmail, (5)Kindle, and (6)YouTube. For each application, each participant was asked to manipulate it "naturally as usual" on Galaxy Nexus with our system for five minutes, and asked to annotate breakpoints according to their own subjective sense.

Figure 4 shows the accuracy results from 50 fold cross-validation with the models based on C4.5 decision tree (J48), while Figure 5 shows the results with RandomForest. The models were created for each of six applications, with different frame length configurations from 0.25 seconds to 5 seconds. In both cases, with frame lengths of 2–3 seconds, accuracy values are above 80–90%. We also tried to make a unified model based on the ground truth of all six applications. Although we need further evaluation and analysis with more applications, the current unified model shows the accuracy around 80–85% which brings us quite positive insight.

Conclusion

This paper proposed Attelia, a novel middleware that senses user's attention status on user's smart phones, in real-time, without any dedicated psycho-physiological sensors. Attelia detects breakpoint[16] of user's activity on smart phones, with our novel "Application as a Sensor"(AsaS) approach and machine learning technique. Our initial evaluation showed quite optimistic accuracy around 80–90%.

Acknowledgement

This research was partly supported by National Institute of Information and Communications Technology (NICT).

References

[1] Adamczyk, P. D., and Bailey, B. P. If not now, when?: the effects of interruption at different moments within task execution. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2004), 271–278.
[2] Bailey, B. P., and Konstan, J. A. On the need for attention-aware systems: Measuring effects of interruption on task performance, error rate, and affective state. *Computers in Human Behavior 22*, 4 (2006), 685 – 708.
[3] Burke, J., Estrin, D., Hansen, M., Parker, A., Ramanathan, N., Reddy, S., and Srivastava, M. B. Participatory sensing. In *In: Workshop on World-Sensor-Web (WSW 2006): Mobile Device Centric Sensor Networks and Applications* (2006), 117–134.
[4] Craik, F. I., Govoni, R., Naveh-Benjamin, M., Anderson, N. D., et al. The effects of divided attention on encoding and retrieval processes in human memory. *Journal of Experimental Psychology-General 125*, 2 (1996), 159–179.
[5] Czerwinski, M., Cutrell, E., and Horvitz, E. Instant messaging: Effects of relevance and timing. In *People and computers XIV: Proceedings of HCI*, vol. 2, British Computer Society (2000), 71–76.
[6] Deloitte Touche Tohmatsu Limited. Deloitte global mobile consumer survey 2012, Sept. 2012.
[7] Garlan, D., Siewiorek, D., Smailagic, A., and Steenkiste, P. Project aura: toward distraction-free pervasive computing. *Pervasive Computing, IEEE 1*, 2 (april-june 2002), 22 –31.
[8] Gillie, T., and Broadbent, D. What makes interruptions disruptive? a study of length, similarity, and complexity. *Psychological Research 50*, 4 (1989), 243–250.
[9] Google Inc. The new multi-screen world — think with google, Aug. 2012.
[10] Haapalainen, E., Kim, S., Forlizzi, J. F., and Dey, A. K. Psycho-physiological measures for assessing cognitive load. In *Proceedings of the 12th ACM international conference on Ubiquitous computing* (2010), 301–310.
[11] Hoshiba, M., Kamigaichi, O., Saito, M., Tsukada, S., and Hamada, N. Earthquake early warning starts nationwide in japan. *Eos, Transactions American Geophysical Union 89*, 8 (2008), 73–74.
[12] Iqbal, S. T., and Bailey, B. Leveraging characteristics of task structure to predict the cost of interruption. In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (2006), 741–750.
[13] Iqbal, S. T., and Bailey, B. P. Investigating the effectiveness of mental workload as a predictor of opportune moments for interruption. In *CHI'05 extended abstracts on Human factors in computing systems*, ACM (2005), 1489–1492.
[14] Kreifeldt, J. G., and McCarthy, M. E. Interruption as a test of the user-computer interface. In *JPL Proceeding of the 17 th Annual Conference on Manual Control* (1981), 655–667.
[15] Machine Learning Group at the University of Waikato. *Weka 3: Data Mining Software in Java*.
[16] Newtson, D., and Engquist, G. The perceptual organization of ongoing behavior. *Journal of Experimental Social Psychology 12*, 5 (1976), 436–450.