From Overload to Overlord:
Reducing Cognitive Load in a Post app-Apocalyptic world

Eve Huang Thullen, David Kallemeyn, Vanessa Casilla
Claremont Graduate University, Fall 2019

INTRODUCTION

Mobile phones have helped people communicate and organize information in new ways since their introduction.

The mobile application ecosystem offers users the functionality of innumerable analog devices, and has incredible potential to make people's daily lives simpler and easier by removing the need to carry around or use physical objects (such as clocks, calendars, CDs/DVDs, etc.) and reduce the need to rely on human memory. Despite those potential benefits, the mobile computing environment and its applications have not exactly delivered on the promiseof simplified and easier lives. Frequent interruptions and notifications, distractions, and the sheer amount of options and information available can lead to confusion and stress. Instead of reducing the cognitive load on individuals, devices frequently increase the cognitive strain of users. What started out as an incredibly promising tool has yet to live up to that potential. Consider the following definition of a tool: "A tool addresses human needs by amplifying human capabilities." [1] As time goes on, it is less apparent that in sum, across all time spent on phones,the mobile computing environment is indeed amplifying human capabilities.

PROBLEM DESCRIPTION

People need a more efficient way to effectively absorb relevant information from their phones. Relating information overload to the context of mobile computing devices is highly relevant.

As users download and installspecific mobile applications (hereafter 'apps'), each app is an entry point to data and subsequently, information users consume, based on their personal preferences and tastes. Once applications became commercialized, they quickly became established as the de facto mode of user-phone interaction. Now that mobile applications are commodities, there are thousands of applications within the marketplace and those who compete within the same niche or functional categories, tend to offer very small degrees of value propositions between them.

Mobile phoneusers should be able to view information as opposed to applications. Information overload is a pervasive and real phenomenon which inhibits our ability to effectively make decisions or take actions based on the volume of information we may need to process. Specifically, this research focuses on reducing cognitive overload of the mobile user as a result of duplicative application functionality and unnecessary screen transitions.

RESEARCH QUESTIONS.

The project has the following two research questions:

1. Can application data be de-duplicated and aggregated effectively?
2. Can content aggregation and de-duplication (a) reduce cognitive load on the part of mobile phone users (specifically quick, repeatable data checks), and (b) reduce interruption overload via an 'unavailable by default'model?

RELATED WORK.

This section describes potential approaches to the problem of mobile app information overload, and whereapplicable specific examples are identified within each broad area.

Aggregators. There are applications that attempt to organize data within specific categories ad present it in a centrallocation. The advantages of this approach is that information can be found in a single place, which should help reduce cognitive loads that result from entering many different applications and context-switching. One of the drawbacks of such an approach is that the app ecosystem is set up to discourage the development and efficient use ofsuch applications and maximize time spent within each individual app. Another potential downside in the context ofthis research is that the data is not aggregated or distilled to an extent that reduces cognitive load. One example of an aggregator app is the MINT app, which organizes financial information across various financial institutions [2]. Another example for health information is the Apple Health app, which combines health and fitness information from Apple apps, selected third party apps, and even medical record information from participating health providers [3]. In terms of cognitive load, it is not clear that this app is an effective solution - the user is left with a number of decision points and configuration options beyond which apps to include, including which metrics/data points to include and when to receive updates and notices for those metrics. The interface also by design does not necessarily limit the amount of information presented, as it allows arbitrary numbers of "highlight" data points. It also includes search functionality, which can compound rather than relieve overload. This is not necessarily a knock on Apple Health, it is more indicative of differing goals: the goal of Apple Health is to present a maximal amount of health information, whereas the goal of this research is to provide the minimum information necessary to have maximal effectiveness.

Subscription models. RSS is an example of a subscription model, where highlights from user selected feeds are presented. Apple Health is also to some extent a subscription model, as you can subscribe to have external services connected to the app. Subscription models have the potential to reduce overload via allowing only selected items to communicate with the user, and blocking all others by default. However, as the number of subscriptions increases, the more difficult it becomes to sift through and process all of the information being received. An example of a subscription based model specifically designed to address information overload is the research done by Sohn et al, where ‘lenses’ are used to filter and deliver information to inboxes. They implemented a ‘universal inbox’ that collected all messages (email, Facebook, SMS, tweets, RSS, etc.), and users could also define their own ‘lenses’ thatwould sort and filter their inbox [4].

Behavioral models. These can be based on machine learning or non-ML, the idea is to record user behavior and modify communication and messaging based on past user interactions. This includes research into ideal times to deliver messages, including delaying messages until a context switch is detected [5, 6, 7].

Network Traffic Analysis. This approach looks at where network data is sent, and in so doing identifies who is monitoring mobile data. One example of this is the Haystack project, which provides access to network traffic alongwith local context to aid in analysis [8]. This type of analysis could be used to augment some of the approaches above, such as the lenses approach of Sohn and colleagues [4].

Operating Systems. Mobile operating systems have their own functionality, to varying degrees, to allow aggregation of information. Certain system preferences may touch on this, and Apple’s latest iOS includes ‘personal automation’ functionality that is intended to reduce the friction that occurs throughout the day due to transitions. For example, it can start a specified app when a certain trigger criteria is met (say geographic location),which does have the potential to streamline mobile device use to an extent. This approach does put the onus on the user to configure and manage these triggers and it is too early to tell what the overall impact on cognitive load will be, but it is certainly an approach worth noting [9]. This type of approach is promising in that it is not app-based, and embedded in the functionality of the operating system. By virtue of not being an app, it has higher privilege levels to manage data and messaging generated by apps, which would be incredibly valuable for the current research. New mobile operating systems are being developed, and one such example, PureOS (which is a convergent OS running on both desktop and mobile environments), has privacy and system controls that may be adaptable to reducing cognitive load [10]. Additional functionality, such as eBPF, an in-kernel virtual machine, alsoshows promise in profiling and adjusting data flows.

Other Mobile Applications: There are a number of stand alone applications that attempt to assist with organization and productivity, and some may be able to reduce cognitive load of mobile users. These range from apps to manageother apps, productivity, utility apps as well as mindfulness apps. These apps are mentioned for the sake of completeness, but did not really impact the current research.

SYSTEM OVERVIEW.

The Overlord system presented here consists of two primary parts - a backend that will classify, sort, and aggregate application information (and eliminate redundant information) in order to reduce cognitive load, and a frontend GUIthat will enable additional functionality, including a hold-notices-by-default approach designed to reduce interruptions and allow cognitive focus on the task at hand.

Another important element of the system is a subscription functionality, whereby an application agrees to share datawith Overlord and is pre-configured with type information to allows seamless communication between the system and the application. The intention behind this is not only to allow for frictionless use of data, but it is hoped that it will result in a new application ecosystem based on subscription (as in data subscription, not a paid subscription) access. If Overlord were to become widely adopted, a business model whereby apps that are installed and utilized by the Overlord system are able to generate revenue by virtue of their provision of valuable (to the mobile user) data, it would not only encourage more Overlord-compliant apps to be developed but could result in a reduction of the number of duplicative and fraudulent apps as it would de-couple apps from the data streams desired by the user and the extraneous elements of the apps. The use of such apps outside of the Overlord interface would be reduced, thereby reducing extraneous data collection/fraudulent activity on behalf of such apps.

Unique features of the artifact include a "limiting-by-default" functionality where unless a user explicitly indicatesthey are 'available', they will be assumed to be engaged in a cognitive task and application interruptions and notifications are held for later viewing. The UI is unique due to its split screen nature that permits both user selection of functionality and data presentation within a single screen.

The Overlord system will initially be programmed as an Android application using the Kotlin programminglanguage, with the hope that it will be adapted to be part of a mobile OS in the future.

SYSTEM DESIGN.

Backend. The backend will do all of the needed data sorting and aggregation, presented to the user in the GUI. Additionally, data specifications will be outlined such that application developers can enable/add compatibility withour system for inclusion in the data summary screens. This may have the added impact of creating an alternative revenue stream for app developers based on user attention/use of the data streams.

Aggregation will be based on general application categories, and will only be offered for a selected set of categories.The number of categories actively available to users at any one time is five, and the total they can select from is expected to be no more than ten. This is for a couple reasons: first, there are only certain categories that are amenable to be aggregated. For example, games are unable to be aggregated in any meaningful way - you are eitherengaged in the game or you are not. Second, there is a strong desire to limit the decisions that a user needs to make,and by restricting the total number of categories available the decision space is reduced. The reduction in decision space comes at the expense of extended functionality, but given the stated research goals it is aligned with a reduction in cognitive load.

Categories were selected based on the research of Thomas W. Jackson and Pourya Farzaneh, who identify seven factors related to information overload and derive a state-based (overloaded or not overloaded) model of overload.The factors are: quantity of information, characteristics of information, quality of information, information processing capacity, available time, task and the process parameters, and personal factors [11].

The aforementioned factors and their sub-components were used to create a matrix to evaluate app categories for thepotential to obtain reductions in cognitive overload. The initial set of categories selected were: health/medical, finance, news, energy & utilities, external devices (smart X), social, travel/weather.

Categorization and Types. Units of measure will be of the utmost importance in making sense of isomorphic measures presented in differing units. Researchers have explored the relationship between types in programming languages and units of measure, which will be relied upon to implement type-based implementations of measurement that ensure sense making across different apps and units [12]. Further work has advanced the discussion to center around fundamental underlying elements, or dimensions; units are a derived as relations among the fundamental dimensions [13]. The primary dimensions envisioned for Overlord consist of time, distance, acceleration, mass, and volume. Compound dimensions will be defined as relationships between primarydimensions (i.e. concentration as mass per volume).

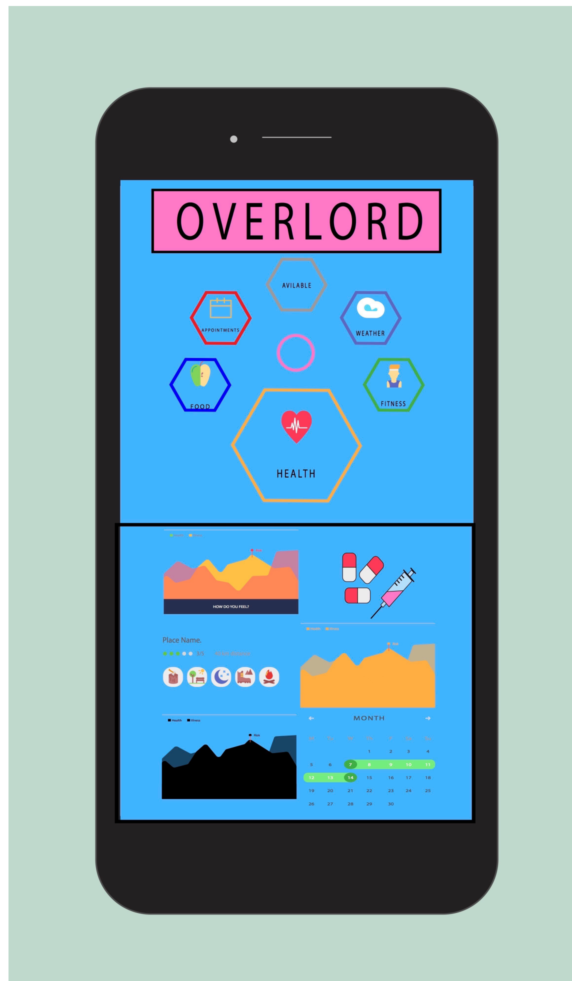

Figure 1 Overlord GUI

Frontend. The GUI design will be a dual split-screen approach (split horizontally), with a rotary style selection of icons on the top half of the screen that allows selection of functionality/topic, and the bottom half presenting the relevant information. User options and configurations are intentionally limited in order to reduce cognitive load, however there will be user configuration in some cases. For example, the screen will show a maximum number of categories at any one time, but the user can select which categories to display. In addition to the category limit, there will of course be a limit to the number of data points presented within each category. The idea is to enforce constraints to reduce cognitive load on users, with much decided in the background to allow focus/cognitive load onthe areas that matter.

The number of categories presented at any one time is five, with a sixth icon a specially labeled ‘available’ icon.
The ‘available’ icon, when selected, will allow any communication or messaging; all other category icons will blockmessaging by default, allowing viewing of messages upon any category switch. The purpose of the block-by-defaultfunctionality is to enable focus on the category being viewed and to prevent context switching; research has indicated that cognitive overload is more acute within tasks than between them, and more for similar than dissimilar tasks (greater care and attention is required to differentiate important information) [14]. The novel use of the ‘available’ option is intended to allow completion/focus on a task, and stage delivery of interruptions for context switches.

The GUI will have a single screen, with user interaction on the top half to select the active category, and the bottom half reserved for data presentation for the selected category. There will be an off/on button in the center of the rotarymenu on the upper half to enable/disable Overlord. We have also envisioned a second screen, which will be presented as part of a separate project.

The primary benefits of the system are:

- Reducing screen/context switching; different categories available on one screen (one at at time), reduces theneed to go to individual apps unless a specific transaction is needed
- Reducing amount of information flow: aggregating data to summarize it
- Improving efficiency of information flow: removing duplicated information
- Limiting cognitive tasks - limit by default allows user to preserve intention when on the overload screen and view the data that is relevant; messages/interruptions occur at category changes, or the 'available' setting, ordisabling the system

Usability will be assessed during our evaluation stage, but is also a consideration during design. As all of the teammembers are mobile phone and application users, usability and user perspectives are being conducted by the four- member team. Given the broad nature of the issue we are attempting to address, anyone with a mobile phone that supports applications is a 'user', meaning nearly everyone in the United States.

EVALUATION.

RQ1: Can application data be de-duplicated and aggregated effectively {to help reduce the cognitive activity onmobile phones}

RQ2: Can content aggregation and de-duplication (a) reduce cognitive load on the part of mobile phone users (specifically quick, repeatable data checks), and (b) reduce interruption overload via an 'unavailable by default'model?

We plan to isolate key functionality from the designed solution for testing with individuals, automated testing and metrics such as success at categorization, as well as user feedback/testing. We plan to recruit students and/or otherindividuals (relatives, friends, etc.) for additional feedback, after obtaining approval from IRB.
For the categorization, we will look at % classified, and % classified correctly.
To assess the cognitive load, we will have users complete a task with our block-by-default system as well as atypical mobile system that allows messages at all times and compare the task completion rates.
We will also utilize in-person assessments of key functionality, including the data aggregation, presentation, and itsimpact (perceived) on cognitive load. Assessments regarding the perception of interruptions will also be used.

Determining whether the solution worked will be done in parts as explained above. Overall technical performancewill be evaluated base on success of categorization, as well as the utility, usability, and improvement of the approach.

Evaluating the social-technical impact will be done via user-testing, having them complete a task and measure groups that have standard interruptions and the system to be tested to allow for comparison. Measures will include number of distractions/interruptions resulting in a screen/context switch, the effect of these interruptions/distractionson task performance. Qualitative feedback will also be collected to assess the impact on cognitive load of both the block-by-default-model and the aggregation as presented.

CONTRIBUTIONS

Theory. The current work applies Sweller's Cognitive Load theory [15] to mobile app overload, and adds interruptions in order to assess the potential impacts of reducing not only information overload at any given point intime but also view how interruptions affect the cognitive load.

Additionally, demonstration of a system that effectively reduces cognitive load and interruption overload basedon Jackson and Farzaneh's model would validate the model and demonstrate how it can be applied to certain categories of information.

Practice. If successful, the Overlord system will demonstrate that systems do not have to become increasingly complex in order to convey information effectively, and that single-screen apps can effectively present information. Overlord's subscription setup will promote apps that provide useful and novel information in an attempt to counter the increasing fragmentation of the mobile app environment. The effective categorization of data streams will promote user benefit as opposed to developer benefit, and illustrate that the world does not need more apps, it needs more effective apps and/or more effective methods of presenting information to users. Having thousands of apps forany given purpose or domain is not really user choice, it is user paralysis.

DISCUSSION AND FUTURE WORK

The approach used in the development and testing of Overlord is of interest to anyone developing mobile applications as well as researching cognitive load. Drawing upon work done across a number of approaches, weanticipate the current research will add to the re-evaluation of the mobile environment currently underway, in thehopes that a more user-friendly environment will evolve from the current approach of simply flooding users withinformation.

Overlord is currently under development as an application, however this is not the ideal solution; complaining about app overload and subsequently developing an app is certainly not desired. The current approach was selecteddue in large part to time and resource constraints, and is intended more as a proof of concept and path to follow to continue development. Thus, the current study can be viewed as a roadmap and pilot study, with continued development to take place after the current evaluation which will take place in multiple parts.

Once the functionality is integrated within a mobile OS it can be tested in its entirety, based upon lessons learned from the current project. Additionally, there are concepts for a second screen to be developed to aid invisualization of data subscribed to by Overlord as well as allowing finer-grained control over what is filtered.

CONCLUSIONS

The mobile computing environment is quickly becoming the primary mode of interaction between people and information systems, and the developing trends have generally been a solidification of the application model where apps compete for user attention. In this environment, it is no surprise that individuals are often unable to effectivelyprocess the incredible amounts of information at their fingertips, and the problem is only worsening as more and more apps are added daily.

The goals of Overlord are to aggregate information and present it in ways that both reduce information overloadas well as provide an environment that limits interruptions. By building on many of the existing approaches to tackle this problem, Overlord implements a novel block-by-default GUI interface that performs automated data classification and aggregation in the background. The subscription functionality will hopefully evolve into more of an attention-based revenue model where apps are rewarded for providing user utility as opposed to maximizing user time on app.

REFERENCES

[1] Bret Victor, November 8, 2011. “A Brief Rant on the Future of Interaction Design”: http://worrydream.com/ABriefRantOnTheFutureOfInteractionDesign/

[2] MINT app. https://www.mint.com/

[3] Apple Health. https://www.apple.com/ios/health/

[4] Timothy Sohn, Vidya Setlur, Koichi Mori, Joseph 'Jofish' Kaye, Horishi Horii, Agathe Battestini, Rafael Ballagas, Christopher Paretti, and Mirjana Spasojevic. 2010. Addressing mobile information overload in the universal inbox through lenses. In *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services* (MobileHCI '10). ACM, New York, NY, USA, 361-364. DOI: https://doi-org.ccl.idm.oclc.org/10.1145/1851600.1851666

[5] Tadashi Okoshi, Kota Tsubouchi, and Hideyuki Tokuda. 2019. Real-World Product Deployment of Adaptive Push Notification Scheduling on Smartphones. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (KDD '19). ACM, New York, NY, USA, 2792-2800. DOI: https://doi-org.ccl.idm.oclc.org/10.1145/3292500.3330732

[6] Shamsi T. Iqbal and Brian P. Bailey. 2005. 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* (CHI EA '05). ACM, New York, NY, USA, 1489-1492. DOI=http://dx.doi.org.ccl.idm.oclc.org/10.1145/1056808.1056948

[7] Mary Czerwinski, Ran Gilad-Bachrach, Shamsi Iqbal, and Gloria Mark. 2016. Challenges for designing notifications for affective computing systems. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct* (UbiComp '16). ACM, New York, NY, USA, 1554-1559. DOI: https://doi-org.ccl.idm.oclc.org/10.1145/2968219.2968548

[8] Abbas Razaghpanah, Narseo Vallina-Rodriguez, Srikanth Sundaresan, Christian Kreibich, Phillipa Gill, Mark Allman, Vern Paxson. 2015. Haystack: A Multi-Purpose Vantage Point in User Space. ArXiv. https://arxiv.org/abs/1510.01419

[9] iOS 13. Apple Inc. https://support.apple.com/guide/shortcuts/intro-to-personal-automation-apd690170742/ios

[10] PureOS, Librem 5 phone. https://puri.sm/products/librem-5/

[11] Thomas W. Jackson, Pourya Farzaneh. 2012. *Theory-based model of factors affecting information overload*, International Journal of Information Management, Volume 32, Issue 6, 2012, Pages 523-532, ISSN0268-4012, https://doi.org/10.1016/j.ijinfomgt.2012.04.006.

[12] Andrew J. Kennedy. 1997. *Relational Parametricity and Units of Measure*. In Proceedings of the 24thAnnual ACM Symposium on Principles of Programming Languages, Paris, France, January 1997. https:// people.mpi-sws.org/~dreyer/tor/papers/kennedy.pdf

[13] Varkor. https://varkor.github.io/blog/2018/07/30/types-for-units-of-measure.html

[14] Ingrid Mulder, Henk de Poot, Carla Verwij, Ruud Janssen, and Marcel Bijlsma. 2006. An information overload study: using design methods for understanding. In *Proceedings of the 18th Australia conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments* (OZCHI '06), Jesper Kjeldskov andJeni Paay (Eds.). ACM, New York, NY, USA, 245-252. DOI=http://dx.doi.org.ccl.idm.oclc.org/ 10.1145/1228175.1228218

[15] Sweller, J. 1994. Cognitive load theory, learning difficulty, and instructional design. *Learning andInstruction, 4*(4), 295-312. http://dx.doi.org/10.1016/0959-4752(94)90003-