Developing Heuristic Based Mobile Apps

Pooja Sinha, Samrat Dutta, Sundaresan Krishnaswami

sinhapooja2004@gmail.com, samrat.dutta501@gmail.com, sundaresank@gmail.com

Abstract

Phones have evolved exponentially in every aspect in the last few years, from a traditional landline to cordless to mobile to smartphone, it has improved in terms of capability, storage, features, components, but above all smartness.

The question arises are smartphones really smart by themselves? Can they solve their own problems? Can they monitor, analyze, repair or archive things without human intervention?

A smartphone is no longer just a calling device, but is used today for almost everything, and this is made possible by, first, the phone itself – with its high-performance yet optimized hardware, and its Operating System – which has opened an avenue for smartphone applications to make the best use of the phone's features while greatly enhancing user experience.

They are several layers of data available on a smartphone that can help both device manufacturers and application developers to leverage the inherent smartness for a myriad of applications such as security, usability or performance.

Our paper is a collaborative effort to analyze such heuristics already applied or researched in the field and present it as a quick reference for product architects to apply it in identity protection, threat protection, optimize resource usage on a smartphone, which otherwise applications may take it for granted.

Biography

Pooja Sinha is a Software Engineer at IBM Software Group, currently working for MaaS360 device management products in Bangalore, India. She is a domain expert in cloud based applications, who has extensive experience in testing large-scale database transactions and analytics. She has also lead the persona driven mobile device management solution for MaaS360 and continues to drive the 3rd party integration points for MaaS360Pooja has a Master's degree in Information technology from IIIT-Bangalore.

Samrat Dutta is a Software Engineer at IBM Software Group's, storage division. He is a software technology enthusiast and interested in opensource projects and has submitted multiple works for Pattern Language of Processing conference, Apache Tomcat along with several whitepapers for IBM's storage portfolio, mainly in the fields of Cloud, Mobile and Software Quality..

Samrat has a Master's degree in Information technology from IIIT-Bangalore.

Sundaresan Krishnaswami is a Staff Software Engineer at IBM Software currently leading the technical testing effort for Android ecosystem at MaaS360. He has over 10+ years of experience in the field of software testing, test automation and mobile application testing.

1 Introduction

Every individual is unique; they think differently, their perspectives differ, and they react differently to the same situation.

This holds true for smartphones, too. Users customize their smartphones based on their priorities or to suit better usability.

For example, a child will use his/her phone mostly for educational, messaging or gaming applications, the kid may also overclock the phone's CPU for gaming. A professional on the other hand may not tweak the hardware while he/she will interested in using productivity applications such as emails, creating documents, customizing notifications or simply browse the Internet. A developer may use the phone for developing or testing applications with minimum applications running the background while several developer options enabled on the device.

Let us illustrate this using an habitual example: After waking up in the morning, almost everyone has the habit of checking their phones, some may access email application and spend some time checking their social networking statuses, plan their day using a to-do app or key-in their notes for the day. The behavior in usage may change depending on the day and time, but tasks become habitual over a period of time.

1.1 Making use of usage patterns

The major concern over buying a smart phone is security. Several questions arise when the phone falls into the hands of an unintended user.

Europe's Union Agency for Network and Information security cite's Data Leakage resulting from loss of device or theft as the major cause of concern (Top Ten Smartphone Risks, 2010). The study is still relevant and one of the reasons several players are getting into the enterprise Mobile Device Management space, to reduce corporate security risks.

1.2 Smartphones are usability tools

With advancement in smartphone electronics, it is easy to make applications that are context aware

Let us take for example an email application. What if the smartphone is left stationary at a location every night and accessed only in the morning.

With location and accessibility data available on the smartphone, the email application can be made aware to stop syncing emails during the night and automatically sync during a time period in the morning.

1.3 Smartphones are performance intensive

One of the significant drawbacks of smartphone, albeit the intensive performance it can offer, is the battery life it offers. All activities consume battery power, making a smartphone usable only while its battery lasts.

What if the smartphone could control its apps' dormancy or what if the apps could actually make use of the data available through the user behavior to modify its performance?

Keeping the above points in mind, we came up with this question -

What if the smartphone with a given set of data could do the following?

a. Block an unintended user from using the phone
b. Leverage the usage pattern to optimize application performance
c. Enhance usability by displaying relevant notifications

To achieve this, we came up with a solution that forms the following framework:

1. Gathering data
2. Data collection methods
3. Creating the activity map
4. Application

2 Gathering data

Data can gathered from logs, settings, events, triggers, alarms, back ground and foreground activity timeouts among others can be helpful in creating the activity map using heuristics.

While we would like to generalize as many data points as possible for any given operating system, we may tend to gravitate towards Android given its openness to the application developer.

Table of reference for user interaction points

Interaction points	**Common data points**
Swipe	• Frequency of use • Maximum usage in a given time period • Minimum usage in a give time period • Applications sensing or using a particular behavior (example, use of the accelerometer or location services) • Applications sharing data • User behavior after using an application or a service • Profiling steep deviations from regular usage
Single tap	
Multi-touch	
Headphone use	
Location,change in location	
Sharing	
Connectivity	
Typing, typing speed	
Volume	
Voice Quality (Prof. Emily Mower Provost)	
Call	
SMS	
Network usage	
Accelerometer (Weiss, 2013)	
Navigation	
Phone on/off status	
Screen lock	
Charging status	
Accessibility	
Brightness	
Accessing notifications	
App preference or settings	
Rooting or Jailbreak status	
Stylus use	

3 Data collection methods

There are several sources for gathering this data, the data collection methods can vary between operating systems.We would like to quote a few examples in the table below.

Data collection methods	Specfic examples
Logs	Analyzing usage pattern from logs is discussed at length on page 5
Application broadcasts	In this paper by Erika Chin et al (Erika Chin), they discuss about how securing inter-application communication on Android. This example could be used in ways application interaction can be captured via Application Broadcasts and analyzed.
Accelerometer / Gyroscope sensor data	In this paper by Brunato et al (M. Brunato), how machine leaning can be used to detect early Parkinson's disease, the authors have used accelerometer to study patient behavior. We understand this can be used in understand any user behavior too)
GPS coordinates or Location detection using Geo-IP	Service providers such as Trusteer lookup location or IP change on a device to understand user behavior[1] to prevent fraud while using Banking application.
Touch sensitivity	Longer unusual user presses can be captured and analyzed for patterns. This stackexchange discussion (Authors) explains how this can be achieved on any operating system.
Compass co-ordinates	In this paper by Nirupam Roy et al (Nirupam Roy), the authors discuss about analyzing user's everyday walking direction based on smartphone's compass co-ordinates. This can be used in building a heuristic to track unusual activity on days the smartphone is carried in a different direction.
Network switching pattern	In this paper by Kevin S Kung et al, (Exploring universal patterns in human home/work commuting from mobile phone) the authors have successfully used big data analytics on cell phone data to understand user commutes on a given day.

[1] *http://www.trusteer.com/blog/real-time-credential-theft-your-risk-engine-won%E2%80%99t-catch-this-one*

3.1 Analyzing the usage pattern from logs

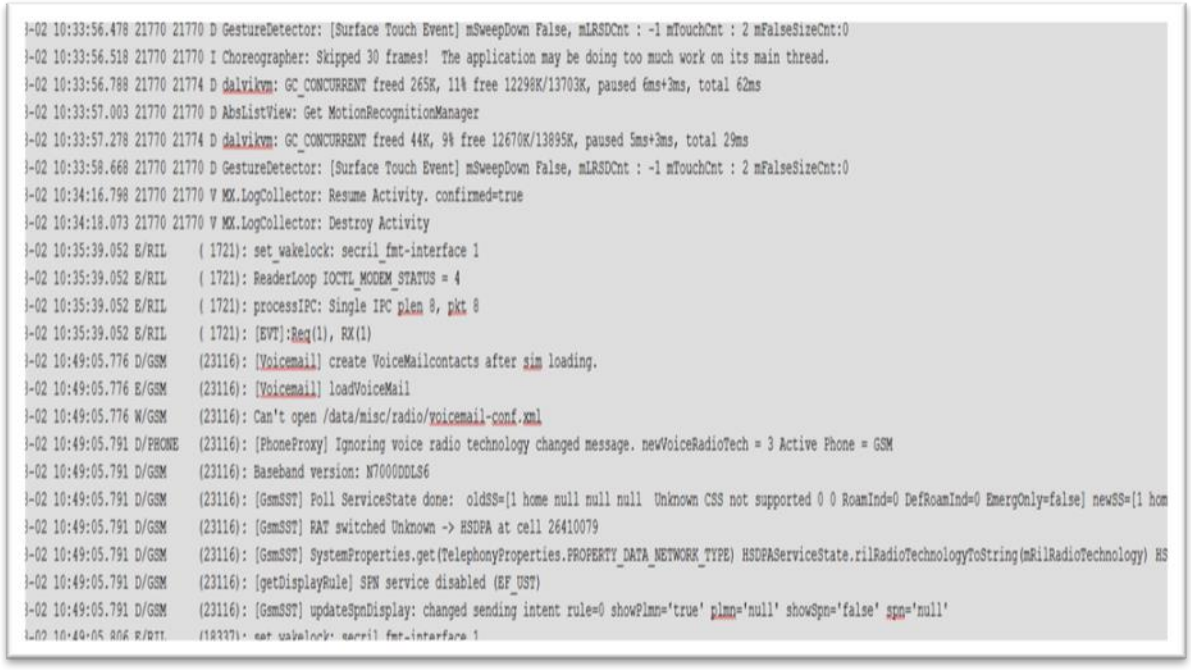

Figure 1

Line 1 in Figure 1 log explains the user gesture activity *mSweepDown* which is set to False, based on this data the 'List view' performs up or down motion on the device. This can give us the user's gesture pattern.

4 Creating the activity map

There are several data interaction points that a smartphone's operating system provides that allow us to create an activity map.

The motivation behind activity map:

1. Activity map created using decision trees are proven methods for machine learning
2. It is simple to understand and implement

The smartphone can heuristically derive the usage pattern and create an activity map. This map can be a decision tree.

To optimize creating the decision tree, based on whether this is implemented on the smartphone itself or by an app, it is best that the process should be only executed when the device is idle.

All data logged in the decision tree will have some value; these values are based on the usage behavior of the user.

For example, when a user buys a new phone with this heuristic applied, every action and sequence of operation the user does get stored in the decision tree and the initial weighted value. This will be a decision tree node.

4.1 Creating and updating the Decision Tree

Again on the next day if the user performs a sequence of operation that may or may not be matched the previous day's data. Now with further usage on upcoming days if the data matches with its initial nodes, the weight for each node will increase.

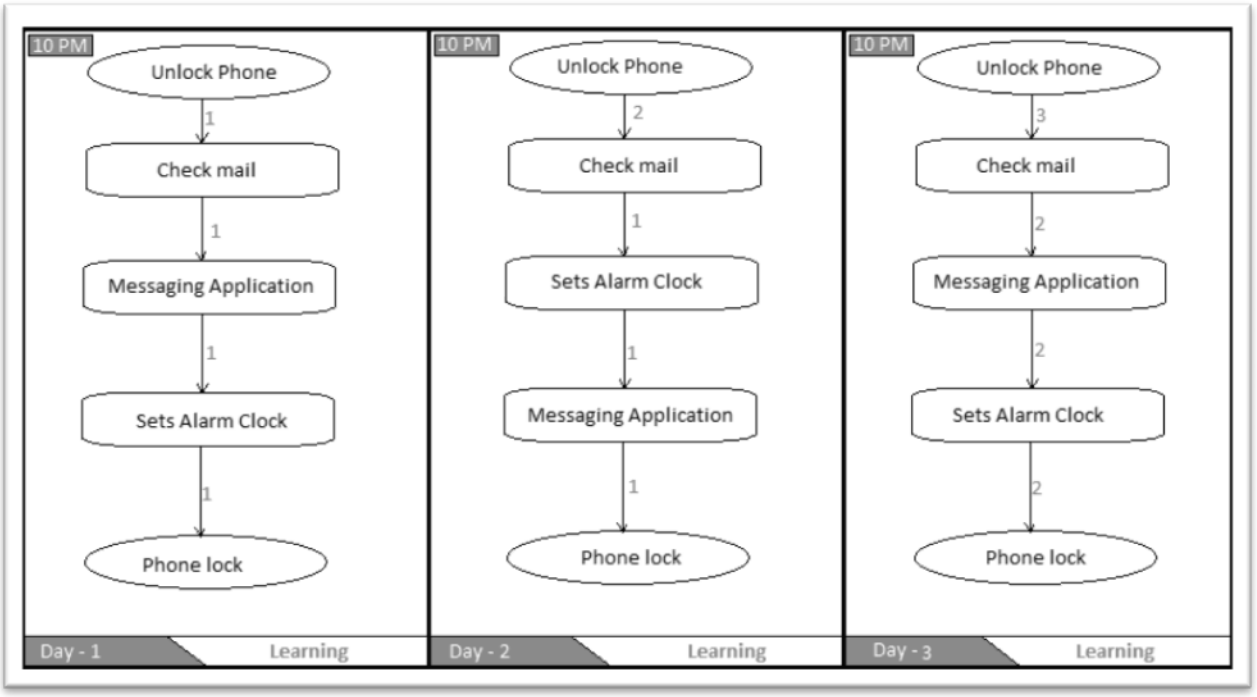

Figure 2

Figure 2 explains that the operating system will derive from the decision the node that has the highest value for a given behavior. Any deviation from this node would mean that it may not be the intended user. Based on a set of conditions the operating system or the application can take action.

4.2 Building the heurisitics

Let us try to put the ‘Building the heuristics’ aka, building artificial intelligence in a perspective with Figure 3 given below:

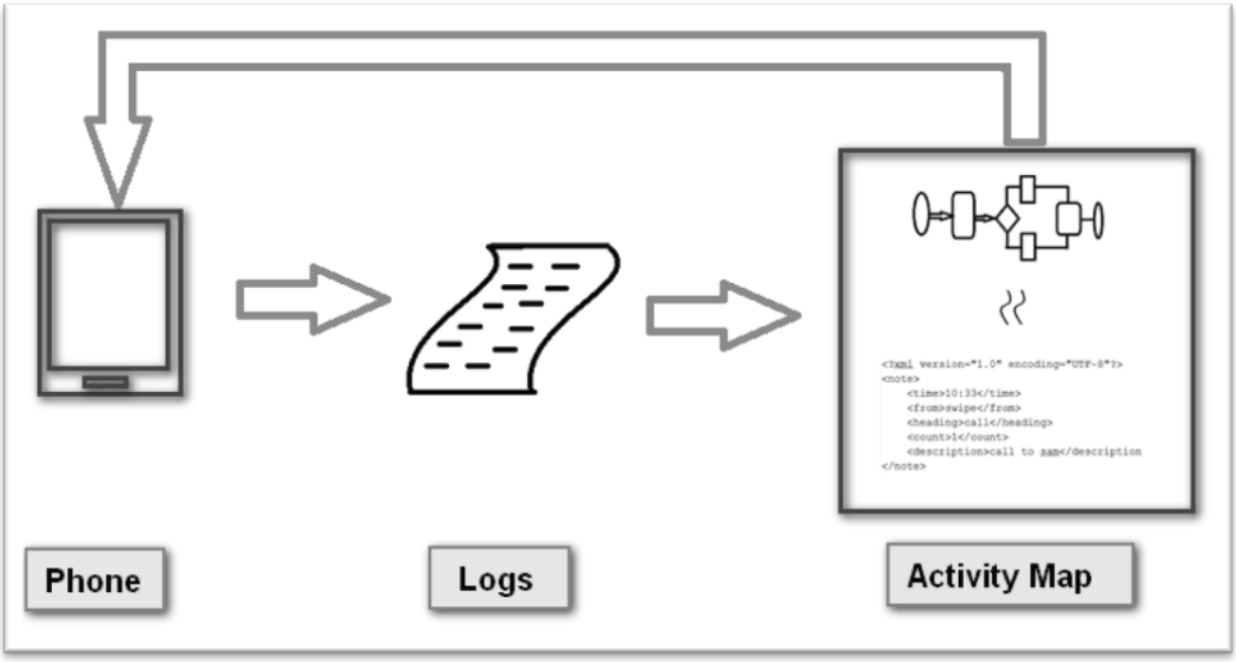

Figure 3

5 Application

5.1 Sample application ideas to apply the Heuristics

5.1.1 Security

- An enterprise application that manages sensitive data, which can also be prone to attacks, can actually derive the usage patterns to block access to the application if required
- The operating system can gather this data, learn from it and enhance security on the device. A user may be prompted or blocked from device use if the usage pattern deviates from a set of regular patterns.

5.1.2 Performance

- An email application can sync user data once based on the usage pattern which was discussed in our example for smartphone as a usability tool.

5.1.3 Usability

- A task management application can learn from user behavior, patterns and automatically start suggesting tasks based on the usage
- Data generated from the application from the usage pattern can be used for various analyses that can help in business development. For example, if the e-commerce industry wanted to target the

20-30 age group, and they want to know: what kind of content they are browsing, the most used phone models, and how much time a user spends on an app on a regular basis. This can be derived using the pattern derived from heuristics, based on these pattern e-commerce industry can venture new business ideas.

6 CONCLUSION

Security, usability and performance are the hot keywords in the mobile industry. With the advent of technology, everything now has become smart enough to fulfill the user's desire.

From a security standpoint the authors believe that a set of 'What if' questions can be answered by building and applying heuristics

Considering the limitations of a smartphone, it is imperative that an application has to be intelligent enough to understand and adapt to a user's whim to make the app more usable.

The same limitations may apply for performance as well. We strongly believe that logging the patterns discussed in our paper, deriving heuristics from it, and applying it wherever we can, can only make smartphones smarter.

References

Authors, M. (n.d.). Retrieved Jul 1, 2014, from http://stackoverflow.com/questions/1127682/is-it-possible-to-determine-the-finger-pressure-on-the-screen

Erika Chin, A. P. (n.d.). *Analyzing Inter-Application Communication in Android*. Retrieved Jul 1, 2014, from https://www.eecs.berkeley.edu/~daw/papers/intents-mobisys11.pdf

Exploring universal patterns in human home/work commuting from mobile phone. (n.d.). Retrieved Jul 1, 2014, from Arvix.org: http://arxiv.org/ftp/arxiv/papers/1311/1311.2911.pdf

M. Brunato, R. B. (n.d.). *KDNuggets.* Retrieved 1 2014, Jul, from kaggle2.blob.core.windows.net: https://kaggle2.blob.core.windows.net/prospector-files/1117/958625cf-3514-4e64-b0e7-13ebd3cf9791/kaggle.pdf

Nirupam Roy, H. W. (n.d.). Retrieved Jul 1, 2014, from http://synrg.csl.illinois.edu/papers/walkcompass.pdf

Prof. Emily Mower Provost, P. S. (n.d.). *Listening to Bipolar Disorder: Smartphone App Detects Mood Swings via Voice Analysis*. Retrieved Jul 2014, 1, from umich.edu: http://www.eecs.umich.edu/eecs/about/articles/2014/app_for_mood_swings.html

Top Ten Smartphone Risks. (2010, Dec 10). Retrieved Jul 1, 2014, from European Union Agency for Network and Information Security: https://www.enisa.europa.eu/activities/Resilience-and-CIIP/critical-applications/smartphone-security-1/top-ten-risks

Weiss, G. M. (2013, Jan 4). *Your Smartphone Knows You Better Than You Know Yourself.* Retrieved Jul 1, 2014, from Inside Science: http://www.insidescience.org/content/your-smartphone-knows-you-better-you-know-yourself/904