AI – Based Approach for Mobile User Interface Adaptation

Hajer Dammak
University of Tunis El Manar
Faculty of Sciences of Tunis
Laboratory LIPAH-LR11ES14
Tunis, Tunisia
E-mail: hajer.dammak@fst.utm.tn

Meriem Riahi
University of Tunis
High National School of Engineers of Tunis (ENSIT)
Tunis, Tunisia
E-mail: meriem.riahi@ensit.rnu.tn

Faouzi Moussa
University of Tunis El Manar
Faculty of Sciences of Tunis
Laboratory LIPAH-LR11ES14
Tunis, Tunisia
E-mail: faouzi.moussa@fst.rnu.tn

Abstract— **The technological evolution and the advent of smart mobile devices have profoundly changed the daily lives of users. Indeed, users are particularly focused on their smartphones in order to manage their activities, check their e-mails, follow the news, connect to social networks, etc. Despite the impressive technological evolution of smartphones, the user interface remains below expectations of users, especially in terms of adaptation. Parallel to the technological evolution, Artificial Intelligence (AI) has also progressed very significantly. This discipline can improve user-smartphone interaction. Indeed, Machine Learning (ML), for example, offers effective means to adapt the interface according to the habits and changes in the behavior of the user. The goal here is to dynamically reorganize the mobile interfaces by grouping the frequently used applications so they will be more efficiently accessible by users. In this sense, the smartphone's log files are used to make better data-driven decisions. These logs will be exploited in a ML approach to model the user's behavior and to propose adaptations. In this paper, we discuss the user feedback for the first results of grouping icons.**

Keywords-Adaptation; Mobile User Interface; User Behavior; Log File; Artificial Intelligence; Machine Learning.

I. INTRODUCTION

In mobile and ubiquitous computing, the context changes frequently, which can affect the functionality of the system. Therefore, the need for adaptation becomes a fundamental requirement. Many researchers tried to define adaptation. Kakousis et al. [1] defined adaptation as '*any kind of structural, functional or behavioral modification of a software component, with the aim of better fitting to a changing environment and satisfying a high-level overall objective*'. More formally, Capra et al. [2] defined it as '*the ability of the application to alter and reconfigure itself as a result of context changes to deliver the same service in different ways when requested in different contexts and at different points in time'*. In brief, adaptation is seen as a discipline that addresses the necessity to adjust information systems behavior in order to meet the specific user characteristics and the current context.

The literature shows a variety of definitions of the adaptive system. In a broader scope, it is a system helps the user in satisfying the need for information by adapting the system and/or the displayed information to the user's specific requirements.

It is important to mention that adaptation in the Web environment differs from adaptation in the ubiquitous environment. The first helps to reduce the information overload problem in order to satisfy the user information need by adapting the system and/or the displayed information to user' specific requirements. The latter helps to deal with the frequently changing context (known as context-aware system) and thus to adapt the system according to the user's current context or/and the user's behavior.

Furthermore, according to Brusilovsky [3], Kobsa [4], Torre [5] and Plumbaum [6], the adaptive system in the Web environment can be divided into three different tasks. The first task is the data acquisition and consists of collecting information about users. The second task is the representation and data mining task and it supports processing information and creating a user model. The last one is the adaptation task. It serves to adapt the application to the user's profile.

Generally, in the ubiquitous environment, the adaptation cycle of a context-aware system is based on Dey's approach [7]. It includes observations of the environment, the selection of adaptations and their executions. Several authors adopt the same adaptation cycle. The approach proposed by Da et al. [8] and Cheng et al. [9] contains four steps, namely: information Collecting, Analysis, Decision and Action (CADA). Joining the same spirit, Dobson et al. [10] and Kakousis et al. [1] define adaptation as a closed loop that includes three phases: (i) context detection and processing (ii) reasoning and adaptation planning (iii) action for adaptation. Based on these two approaches, we can conclude that the adaptation system in the ubiquitous environment has four tasks: context manager, planner, decision-maker and middleware.

The Human-Computer Interaction (HCI) discipline helps to improve the usability of User Interface (UI) and provides better interaction between users and systems. In other words, UI should be easily adapted to perform various tasks. The use of mobile applications is relatively new compared with the use of desktops and Websites. Desktop computers do not suffer from much interference from the external environment compared to mobile devices. Consequently, applying the traditional HCI adaptation methods for mobile applications is not efficient [11]. With the continuous growth of the number of mobile devices and applications, it becomes crucial to understand how the users interact with their devices and applications. The user can change effortlessly the purpose of a mobile device through the used applications. The smartphone can be transformed into Global Positioning System (GPS), musical instruments, credit cards, among others [12].

Smartphones are equipped with various applications. Some of the applications exist by default and some of them are installed by the user. However, many applications remain unused, or rarely used, while others are regularly used. In fact, some applications are frequently used and may not be considered as important applications while others are less used and may be considered as important ones such as social media versus e-mail application. . In this research work, we aim to create an adaptive Mobile User Interface (MUI) by adopting the grouping approach. We wanted to test the efficiency and the practicality of this method. The idea behind adopting this hypothesis is that we noticed that the grouping of applications is static and fixed by the device manufacturer. Consequently, we tried in this study to group applications in a dynamic and modifiable way.

In this work, we raise the following questions: How can we measure the importance of an application for a user: is it measured by the frequency of use, by the time spent on the application or by other criteria? How can we group the “frequently used applications” efficiently? By application’s category or by user’s category? How to monitor the interface with this new grouping? Where are we going to put the created grouping in the interface? Where is the most suitable icon’s position? Do we need more than one grouping?

To answer all these questions, we propose, in this paper, a solution for adapting MUI based on ML through log files. The remainder of the paper is organized as follows: Section 2 gives an overview of the related works. Section 3 describes the proposed approach. Section 4 presents the experiment carried out in a case study while Section 5 discusses the feedback from users’ evaluation. Finally, Section 6 concludes the paper and outlines future perspectives.

II. RELATED WORK

To understand the adaptation process in adaptive systems that occurs in runtime stages, many research works tried to classify the adaptation methods by referring to these questions: who, why, what and how the adaptation occurs [13]-[15]. Particularly, Almutairi and Alharbi [16] offer a graphical illustration of the adaptation taxonomy where they explain the 4 dimensions. The "WHY" dimension explains the reasons for launching an adaptation: the purpose of adaptation can be corrective, adaptive, perfective, extending or preventive. The “WHO” dimension describes the problem of adaptation from various actors (human and software) that are involved. In the “WHAT” dimension, the adaptation aims and objectives are classified. The “HOW” dimension supports applying adaptation by specifying the particular strategic approaches, decision mechanisms and implementation approaches.

In our case, we are aiming for an adaptive adaptation. However, we are especially interested in the "HOW" dimension. In the literature, different UI adaptation approaches exist. They are generally based on the user model. Except that this model is generally static and is previously defined. Consequently, it doesn’t take into account the user's behavior changes and its evolution while using his mobile device. Thus, the idea of relying on the user's behavior via the log files seems interesting for the success of the adaptation process.

In the first part of this section, we point out the importance of log files in studying user behavior. In the second part, we enumerate research works that focus on collecting data from mobile devices using log files. The last part is dedicated to the contribution of AI and ML in the adaptation process.

A. Log File

When it comes to studying user interaction with mobile applications, we can distinguish three approaches in the literature for the usability studies: laboratory experiments, field tests and logs studies. In HCI research, behavioral logs arise from the activities recorded when the user interacts with devices. Dumais et al. [17] highlight approaches that are in contrast with log studies presented in two dimensions. The first one indicates whether the studies are observational or experimental. The second one indicates the naturalness, depth and scale of the resulting data (Table I).

TABLE I. DIFFERENT USER STUDIES IN HCI RESEARCH

	Observational	Experimental
Lab Studies		
Controlled interpretation of behavior with detailed instrumentation	In-lab behavior observations	In-lab controlled tasks, comparison of systems
Field Studies		
In the wild, ability to probe for detail	Ethnography, case studies, panels (e.g., Nielsen)	Clinical trials and field tests
Log Studies		
In the wild, little explicit feedback but lots of implicit signals	Logs from a single system	A/B testing of alternative systems or algorithms

Laboratory experiments represent the most controlled approach. The participants are brought into the laboratory and asked to perform pre-fixed tasks. Such experiments imply perceiving the participant performing the tasks and the evaluation of usability is realized during the interaction. The observed behavior happens in a controlled and artificial setting and may not represent the behavior that would be observed "in the wild" [18][19]. Consequently, collecting data in laboratory studies is expensive in terms of the time which limits the number of participants and systems that can be studied.

Data collection in field studies tends to be less artificial and less controlled than lab studies. The participants are in their real usage environments conducting their own activities and in general, they are periodically asked for additional information. Observation of users in their own environments allows gathering information including interruptions and distractions. As affirmed by Barbosa [20], this is an investigation of the reality of the users and not of assumptions. Field studies bring the benefit of understanding user's interaction and the influence of external factors on such interaction. Yet, collecting data in this approach is not an easy task. In fact, it may change the user behavior, as capturing user's interaction in the field is an intrusive method [21].

Contrary to lab studies and field studies, log studies appear as the most natural observation as the systems are used with no influence by experimenters or observers. Log studies provide a portrait of uncensored behavior. They give a more complete, accurate picture of all behaviors. Furthermore, logs have the advantage of being easy to capture at scale. They can easily include data from tens or hundreds of millions of people while laboratory and field studies typically include tens or hundreds of people. Logs are more about WHAT users are doing rather than WHY. In other words, there is less information about the user's motivation and user satisfaction. Furthermore, log data can be used to test hypotheses that researchers develop about user's behavior. In the case of mobile devices, we distinguish 2 types of log files:

- Log files-oriented application: traces generally the user's interaction with a particular application.
- Log files-oriented device: traces usage information, the context of use and data across arbitrary apps.

B. Data collection

We aim to adapt the MUI for Android devices based on the most frequently used applications. Thus, the data collection task is an important task to acquire the looked-for adaptation. To enhance our knowledge about collecting data, we tried to answer the following questions through our readings: What is the main purpose behind collecting data? What data is collected? How it is collected (approaches/methods)? Where is it stored? What are the types of logs (extension)?

For most of the research works, the purpose behind using log files is to develop methodologies and techniques to evaluate smartphones or application usability. Indeed, analyzing the interaction log file allows a better understanding of how the user behaves with his mobile device and applications.

Marczal and Junior [22] catalog a series of variables that are examined while studying user behavior. They classified the variables into interaction, context and device variables. The first category "determines the user behavior while the user interacts with the application". The second one "concerns the physical, social, temporal and technical environment where the interaction took place". The last category "represents the device characteristics with which the user interacted".

The log file can help to capture user interaction with applications accurately and efficiently. Otherwise, the challenge of restoring to such a file rests on the whole process of preparing the system from collecting data to extracting and interpreting the logged data. Thus, it would be easier to have a tool that can process a large amount of data. Table II summarizes the work of some researches around log files and data collection.

Fernandez and Hussmann [23] developed a tool EvaHelper that helps developers in the usability analysis of the mobile application. The authors simplify the developer's task of evaluating and processing of the automated data collection. The proposed methodology is based on 4 phases: preparation, collection, extraction and analysis.

Ma et al. [24] propose a toolkit that embeds into mobile applications the ability to automatically collect UI events as the user interacts with the applications. The events are fine-grained and useful for quantified usability analysis. The authors implement the toolkit on Android devices and evaluate it with a real deployed Android application by comparing event analysis with traditional laboratory testing.

Kluth et al. [25] modify the four-phased model of Fernandez and Hussmann [23] by adding an automatic critique phase. This phase allows the developer to get feedback in the form of a suggested improvement of usability issues analyzed.

For the same purpose of usability evaluation in a mobile application, [26] presents a solution for the need of applying cost-effective methods to such evaluation. The authors extend Google's API basic service named Google Analytics for Mobile Applications (GAMA) to collect specific low-level user actions. The solution allows lab usability testing, automating quantitative data gathering on one hand and logging real use after application release on the other hand. The mentioned work needs instrumentation of specific code to collect data.

Alternatively, Holzmann et al. [27] [28] present an open-source toolkit for Android that does not require any instrumentation of the application source. The toolkit allows automated logging of the mobile device to evaluate the efficiency of the MUI. It traces user interactions, the context of use and works across arbitrary applications on Android devices.

The commonly used log format for mobile systems is comma-separated-values, CSV. Researchers chose this format instead of XML to minimize the file size, knowing that XML needs additional tags to stock information.

We conclude that most of the works that use log files are application-oriented. They focus on evaluating the usability of a particular application. Rare are the works that are interested in evaluating the mobile device and that focus on the adaptation part.

C. Artificial Intelligence in Adaptation

AI can bring added value to the adaptation process. Initially, we need to clarify what AI really is. It can be several things such as doing smart things with computers or doing smart things with computers the way people do them. In the field of AI, imitating human approaches has been a long-standing effort as a mechanism to confirm our understanding.

The second field of study of AI is ML. It provides computers with the ability to learn autonomously, using a combination of methodologies developed by the statisticians and computer scientists, to learn relationships from data while also placing emphasis on efficient computing algorithms. Here, we do not try to model what is happening. Instead, we simply provide inputs and feedback on the outputs. With a learning algorithm, there is some procedure whereby the computer changes its approach to better match the desired output. Eventually, the machine learns what to do. Also, the resulting ‘rules’ may be opaque to semantic inspection: we can not necessarily intuit what rules are being used, even if the output is good [29].

The automated learning power of ML helps data scientists gain knowledge in a variety of applications such as computer vision, speech processing, natural language understanding, neuroscience, health and Internet of Things (IoT). The major challenge of using ML in big data is to perform the analysis in a reasonable time [30].

ML techniques are used in many fields for different purposes [31]: analyze and diagnose medical images in radiological medicine [32][33], predict the susceptibility of soil liquefaction [34] and forecast models of consumer credit risk [35]. The used techniques fall under one of two categories of supervised or unsupervised learning. In the first category, algorithms are trained using labeled data, while in the second category, algorithms are used against data which are unlabeled.

The following section explains how we use the ML in our approach to adapt the MUI based on user interaction.

TABLE II. RESEARCH WORKS USING LOG FILES FOR DATA COLLECTION

	[23]	[24]	[25]	[22]	[27]	[26]	[28]
Platform	Android	Android	iOS	Android + iOS	Android	Android	Android
Log format	CSV	-	-	-	CSV	-	-
Storage	Mobile device	Central server	Central server	Server	Mobile device	GAMA Server	Web Server
Instrumentation	√ (manually added code)	√ (requires code modificationl)	√	-	×	√	-
Type de collection	Triggered by the user		Auto	Auto (service)	Triggered by the user	Auto	Triggered by the user
Scaling	×	×	-	√	×	√	×
Collected data	Interaction data	Interaction info: UI events	Interaction info	• Interaction info • Mobility (GPS, data sensor)	• Context of use • Interaction info	Interaction info	• Visited apps' screenshot • Interaction info
Test / evaluation	Real world / lab	Lab	Lab	Real world	Real world	Lab + Real usage	Lab
Purpose	Usability analysis	Usability eval	Usability eval	Behavior analysis	UI eval	Usability eval	UI Evaluation
Object of study	Mobile apps	Mobile apps	Mobile apps	Mobile apps	Mobile device	Mobile apps	Mobile UI

III. PROPOSED APPROACH

Our research focuses on adapting MUI based on user behavior: the user interaction with mobile applications. Thus, we can manage the used apps by grouping them as “frequently used apps” in a dynamic and changeable way.

To reach our goal, we propose an approach based on 3 phases. As mentioned in Figure 1, the first phase is considered as field studies. It consists of training the model. The second phase is oriented log studies, i.e., real-world interactions. It consists of recommending to the user a set of apps mostly used and pre-judged as "important" by the model. The latter phase uses a fine-tuning, for continuous improvement and sends the noticed changes to the model so it can cope with them.

To train the model in the first phase, several steps should be followed. The first step focuses on collecting data from diverse mobile users. To accomplish this task, we use an open-source Android toolkit called Automate that supports field studies on mobile devices proposed by Holzmann et al. [27]. It allows the automated logging of mobile device usage in the background. It captures data related to user interactions, the context of use and works across arbitrary apps. This software does not require any instrumentation of the application source code. We are trying to gather an important amount of log files from volunteers, mainly students, from our university over a period of time. These files are stored in the external storage directory of each smartphone and are sent voluntarily by participants. We gather the log files and then we import them into our data store where we perform an offline analysis.

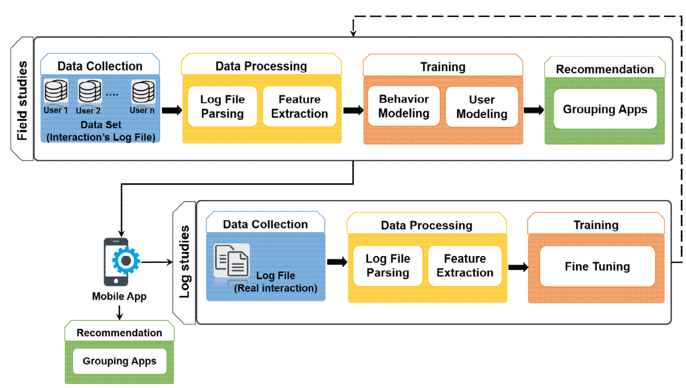

Figure 1. Proposed approach for the adaptation process.

The second step consists of processing the previously collected data by parsing the log files and extracting the most appropriate features. We can group the features to two main aspects: time and visit. The former takes into account

 ISBN: 978-1-61208-761-0

information, such as the time spent per application (time linger), time of the last visit and time of the day (morning, afternoon). The latter takes into account information, such as the frequency of visits (number of visits per day) and the sequence of visited apps out of one session. A session, in this context, starts when the screen is turned on and the device returns from sleep mode and it lasts until the screen display is off again (returns to sleep mode) or when the device is turned off completely. The third step represents the AI module to train the data by applying ML techniques. At this stage, we apply an offline learning where we first use non-supervised algorithms to train the model and to identify the different clusters of user behavior. Second, we use supervised algorithms to classify the new entry.

In this study, we examine the following unsupervised ML algorithms:

- *Agglomerative clustering*: It is a subgroup of K-means clustering which is an iterative clustering algorithm that helps find the highest value for every iteration. Agglomerative clustering starts with a fixed number of clusters. It allocates all data into the exact number of clusters. This clustering method does not require the number of clusters K as an input. The agglomeration process starts by forming each data as a single cluster. This method uses some distance measure and reduces the number of clusters (one in each iteration) by merging process.
- *Hierarchical Clustering*: It is an algorithm which builds a hierarchy of clusters. The Hierarchical clustering Technique can be visualized using a Dendrogram which is a tree-like diagram that records the sequences of merges or splits. Hierarchical clustering methods summarize the data hierarchy, i.e., they construct a number of local data partitions that are eventually nested. The clustering outcome depends on the selected linkage strategy (single, complete, average, centroid or Ward's linkage) and the similarity measure being considered.

We also examine the following supervised ML algorithms:

- *Logistic Regression:* Linear regression attempts to fit a line to data that has only two levels or outcomes, whereas, logistic regression models the chance of an outcome based on a transformation known as a logit [36].
- *Support Vector Machine (SVM):* The Support Vector Machine algorithm uses training examples to create a hyperplane that separates the dataset into classes. The complexity of classes may vary, but the simplest form of the SVM algorithm has only two possible labels to choose from. To reduce misclassifications, a decision boundary is obtained while training the SVM algorithm. This decision boundary is known as the optimal separation hyperplane.

The last step recommends a grouping of the frequently used apps and places it on the most suitable place for the user as mentioned in Figure 2.

After training the model, we suggest installing the framework into the mobile device. This framework supports log studies, i.e., real-world interactions. Similarly, as the first phase, several steps should be followed. The first step consists of collecting the user's real time interactions. The second step consists of processing the gathered data, which will be forwarded to a real-time data analysis system for learning. The third step consists of fine-tuning the ML model. This step can be described as rotating TV switches and knobs to get a clearer signal. In fact, with fine-tuning, the learning of new tasks relies on the previously learned tasks. The fine-tuning ML predictive model is a crucial step to improve the accuracy of the forecasted results. It is essential to mention that, if we want to improve the accuracy of our forecasting model, we ought to enrich data in the feature set first.

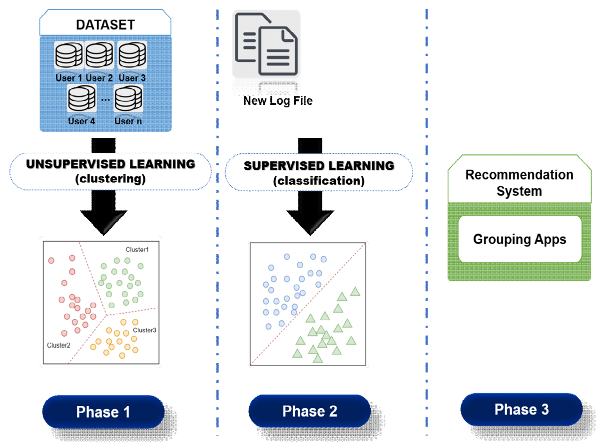

Figure 2. Leraning process of the proposed approach.

The idea of grouping the applications raises many questions. How many groups of applications should we create? Should we group according to the application's category or according to the user's category? How many applications per group? What is considered as the most suitable place for the user (bottom, up, left, right, in the middle)? Does the user prefer a group of applications or does he prefer them to be placed in the main widget? In the case of many widgets, in which widget should we place the recommended group? And if the widget is overloaded, what is the best decision to make?

We can notice that the user's feedback is important to evaluate the adapted interface. Thus, taking into account the user's interaction with the created grouping can improve the model. In fact, modifying the place of the group or re-adjusting it must be considered in the next generation of grouping.

IV. CASE STUDY

As we mentioned previously, we used the toolkit named Automate proposed by Holzmann et al. [27] for collecting data. The resulting log file is in CSV format. As seen in Figure 3, the extracted log file contains overall interesting information like the sequence of opened apps, app usage duration, phone orientation, where the user clicked, etc. The given sample of log file shows the used application: Google Quick Search Box and WPS office.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<session>
    <appUsage
    packageName="com.google.android.googlequicksearchbox"
    name="Google"
    startTime="1573929957560">
        <state
        name="[Tap to update]"
        className="android.widget.FrameLayout"
        duration="211"
        interactionCount="1"
        orientation="1"
        />
    </appUsage>
    <appUsage
    packageName="cn.wps.moffice_eng"
    name="WPS Office"
    startTime="1573929960194">
        <state
        name="[ WPS Office]"
        className="cn.wps.moffice.documentmanager.PreStartActivity"
        duration="2231"
        interactionCount="1"
        orientation="1"
        />
    </appUsage>
</session>
```


Figure 3. Excerpt from a log file.

These data can be used in many ways. Some use cases of log files processing can be:

- *User Behavior Analysis*: By studying the underlying patterns and styles of use of a user, we can detect personality traits and classify a user into a category, which can be used later for various recommendations or as input for other use cases.
- *Apps recommendations*: We can cluster users and log each type's most-used apps, then recommend the apps to a new user that has been classified as one of them but lacks some apps.
- *Sequence detection*: Some people have regular app patterns (exp: Mail then Social Media). If we can identify a user's pattern, we could make it so that apps can be loaded in advance into memory or get their notifications refreshed before the user actually clicks on the app, thus making it easier for him.
- *App grouping*: Like the apps recommendations, we could look into how the user groups their apps and propose a similar grouping to a new user, which could interest him.

We tested our approach on 3 users having different backgrounds and different attitudes. User#1 is an entrepreneur having an unpredictable lifestyle and actively toggling between work and fun every day. He has only 1 widget screen, where he put all his apps into multiple groups (professional, social, entertainment). User#2 is a Ph.D. student who has many widget screens and doesn't group her apps. User#3 is a startup CEO and has multiple widgets screen, but uses solely the home widget where he puts only productivity apps to focus on his work. Figure 4 shows screenshots of the main widget of the 3 users before and after grouping the used apps.

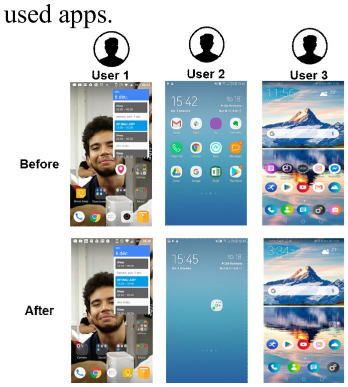

Figure 4. Screenshots before and after grouping the used apps.

These users have been using their configuration for a long time and they announced that they are satisfied with the way apps are arranged. After proposing a new layout, we asked them for their feedback. User#1 said that the grouping didn't go well with his needs as he initially grouped his apps based on his frequency of use and routine. User#2 completely refused the proposition as she just doesn't like to have groups. She prefers to set the most important apps in the main widget. User#3 said that, while the grouping made sense, it's ineffective to have one group when there are a lot of empty spaces in the home widget. We discuss the feedback from user evaluation in the next section.

V. Discussion

The given feedbacks highlight an important point: as much as the solution can technically be good, is it really useful? Although the users' evaluation feedback is negative toward the grouping method, nevertheless, this does not indicate that the conceptual model of the prototype is wrong or needs revision. It denotes that it is natural that people do not like significant changes in a very short time. The case study here drastically changed routine usage. Therefore, we are looking to make the approach more friendly to mobile users and thus by taking into account the periodicity and the frequency of adaptation.

VI. Conclusion

In this paper, we pointed out the importance of log files in the adaptation process in order to make the mobile user's experience better. We also presented use cases on how log files could be used and went in depth with suggesting grouping mobile apps. We presented a novel process-based AI where we use ML to understand user's behavior. It consists of recommending to the user a set of apps mostly used and pre-judged as "important" by the model. We used a case study to show a sample of adapted interfaces from different users with different attitudes. The user's feedback did not show a big interest in the grouping which brought us questioning usability versus utility. Furthermore, the given feedback points out that users do not like major changes in their devices. Thus, in future work, we will study and adjust the periodicity and the frequency of adaptation so the user can benefit from an ongoing interaction. Besides, we aim to consider the user mood for a smooth user experience. We will examine further the performance of many other ML algorithms. We intend to expand our experimentation to a wider, but specific, audience and we will be exploring utility-first solutions whose sole purpose is to improve the mobile user's interaction. In addition, we plan to explore users' implicit feedback to fine-tuning the model to get a more accurate adaptation.

References

[1] K. Kakousis, N. Paspallis and G. A. Papadopoulos, "A survey of software adaptation in mobile and ubiquitous computing", Enterprise Information Systems, vol. 4, no. 4, pp. 355-389, 2010.

[2] L. Capra, W. Emmerich and C. Mascolo, "Reflective middleware solutions for context-aware applications", International Conference on Metalevel Architectures and Reflection, Springer, Berlin, Heidelberg, pp. 126-133, September 2001.

[3] P. Brusilovsky, “Methods and techniques of adaptive hypermedia”, User modeling and user-adapted interaction, vol. 6, no. 2-3, pp. 87-129, 1996.

[4] A. Kobsa, “Generic user modeling systems”. User modeling and user-adapted interaction, vol. 11, no. 1-2, pp. 49-63, 2001.

[5] I. Torre, “Adaptive systems in the era of the semantic and social Web, a survey”, User Modeling and User-Adapted Interaction, vol. 19, no. 5, pp. 433-486, 2009.

[6] T. Plumbaum, T. Stelter and A. Korth, “Semantic Web usage mining: Using semantics to understand user intentions”, International Conference on User Modeling, Adaptation and Personalization, Springer, Berlin, Heidelberg, pp. 391-396, June 2009.

[7] A. K. Dey, G. D. Abowd and D. Salber, “A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications”, Human–Computer Interaction, vol. 16, no. 2-4, pp. 97-166, 2001.

[8] K. Da, M. Dalmau and P. Roose, “A Survey of adaptation systems”, International Journal on Internet and Distributed Computing Systems, vol. 2, no. 1, pp. 1-18, 2011.

[9] B. Cheng et al., “Software engineering for self-adaptive systems : A research roadmap”, Software Engineering for Self-Adaptive Systems, Springer Berlin Heidelberg, vol. 5525, pp. 1–26, 2009.

[10] S. Dobson et al., “A survey of autonomic communications”, ACM Trans. Auton. Adapt. Syst., vol. 1, no. 2, pp. 223–259, 2006.

[11] A. H. Kronbauer and C. A. S. Santos, “Um modelo de avaliação da usabilidade baseado na captura automática de dados de interação do usuário em ambientes reais”, In Proceedings of the 10th Brazilian Symposium on Human Factors in Computing Systems and the 5th Latin American Conference on Human-Computer Interaction, Brazilian Computer Society, pp. 114-123, October, 2011. [In English: “A usability evaluation model based on the automatic capture of user interaction data in real environments”].

[12] M. Böhmer, B. Hecht, J. Schöning, A. Krüger and G. Bauer, “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, ACM, pp. 47-56, August, 2011.

[13] D. Weyns and T. Ahmad, “Claims and evidence for architecture-based self-adaptation: a systematic literature review”, In European Conference on Software Architecture, Springer, Berlin, Heidelberg, pp. 249-265, July, 2013.

[14] M. Salehie and L. Tahvildari, “Self-adaptive software: Landscape and research challenges”. ACM transactions on autonomous and adaptive systems (TAAS), vol. 4, no. 2, pp. 1-42, 2009.

[15] L. Tang, H. Liu, J. Zhang, N. Agarwal and J. J. Salerno, “Topic taxonomy adaptation for group profiling”. ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 1, no. 4, pp. 1-28, 2008.

[16] A. Almutairi and M. Alharbi, “A Survey of Adaptation and the Best Approach for Ubiquitous Systems”, International Journal of Computing and Digital Systems, vol. 6, no. 05, pp. 277-284, 2017.

[17] S. Dumais, R. Jeffries, D. M. Russell, D. Tang and J. Teevan, “Understanding user behavior through log data and analysis”, In Ways of Knowing in HCI, Springer, New York, NY, pp. 349-372, 2014.

[18] J. Kawalek, A. Stark and M. Riebeck, “A new approach to analyze human-mobile computer interaction”, Journal of usability studies, vol. 3, no. 2, pp. 90-98, 2008.

[19] C. Mayas, S. Hörold, C. Rosenmöller and H. Krömker, “Evaluating methods and equipment for usability field tests in public transport”, In International Conference on Human-Computer Interaction, Springer, Cham, pp. 545-553, 2014, June.

[20] S. Barbosa and B. Silva, “Interação humano-computador”, Elsevier Brasil, pp. 263-326, 2010.

[In English: “Human-computer interaction”].

[21] B. Brown, M. McGregor and E. Laurier, ”iPhone in vivo: video analysis of mobile device use”, In Proceedings of the SIGCHI conference on Human Factors in computing systems, ACM, pp. 1031-1040, April, 2013.

[22] D. Marczal and P. T. A. Junior, “Behavioural Variables Analysis in Mobile Environments”, In International Conference of Design, User Experience and Usability, Springer, Cham, pp. 118-130, August, 2015.

[23] F. Balagtas-Fernandez and H. Hussmann, “A methodology and framework to simplify usability analysis of mobile applications”, In Proceedings of the 2009 IEEE/ACM International Conference on Automated Software Engineering, IEEE Computer Society, IEEE Computer Society, pp. 520-524, November, 2009.

[24] X. Ma et al., “Design and implementation of a toolkit for usability testing of mobile apps”, Mobile Networks and Applications, vol. 18, no. 1, pp. 81-97, 2013.

[25] W. Kluth, K. H. Krempels and C. Samsel, “Automated Usability Testing for Mobile Applications”. In WEBIST, no. 2, pp. 149-156, 2014.

[26] X. Ferre, E. Villalba, H. Julio and H. Zhu, ”Extending mobile app analytics for usability test logging”, In IFIP Conference on Human-Computer Interaction, Springer, Cham, pp. 114-131, September, 2017.

[27] C. Holzmann, D. Steiner, A. Riegler and C. Grossauer, “An android toolkit for supporting field studies on mobile devices”, In Proceedings of the 16th International Conference on Mobile and Ubiquitous Multimedia, ACM, pp. 473-479, November, 2017.

[28] A. Riegler and C. Holzmann, “Measuring visual user interface complexity of mobile applications with metrics”. Interacting with Computers, vol. 30, no. 3, pp. 207-223, 2018.

[29] https://www.litmos.com/blog/articles/the-realities-of-artificial-intelligence-and-adaptive-learning, last access 03.12.2019

[30] L. Zhou, S. Pan, J. Wang and A. Vasilakos, “Machine Learning on Big Data: Opportunities and Challenges”, pp. 350-361, 2017.

[31] Y. Seyedfaraz, “User Behavior Analysis using Smartphones”. Ph.D. Thesis, 2017. https://thescholarship.ecu.edu/bitstream/handle/10342/6330/YASROBI-MASTERSTHESIS-2017.pdf?sequence=1&isAllowed=y

[32] R. C. Deo, “Machine learning in medicine”, Circulation, vol.132, no. 20, pp. 1920–1930, 2015.

[33] S. Wang and R. M. Summers, “Machine learning and radiology”, Medical Image Analysis, vol.16, no. 5, pp. 933–951, 2012.

[34] P. Samui and T. G. Sitharam, “Machine learning modelling for predicting soil liquefaction susceptibility”, Natural Hazards and Earth System Science, vol. 11, no. 1, pp. 1–9, 2011.

[35] A. E. Khandani, A. J. Kim and A. W. Lo, “Consumer credit-risk models via machine-learning algorithms”, Journal of Banking and Finance, vol. 34, no. 11, pp. 2767–2787, 2010.

[36] S. Kristin, “Logistic Regression”, PM&R, vol.6, no. 12, pp. 1-28, 2006.