International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 05 | May 2019 www.irjet.net p-ISSN: 2395-0072

Cloud-Based Naive Bayes Classifier for Dynamic Design to Support Usability for Smart Homes Apps

Amirah Alshammari[1], Haneen Alhadeaf[2], Sabreen Alrasheedi[3]

[1,2,3]Department of Informatics, College of Computer, Qassim University, Kingdom of Saudi Arabia.

Abstract - *Many UI/UX design tool kits recently used for the mobile apps industry, some of them are available for free. These tools are considered usability principles for convenient, reliable and interactive design. Unfortunately, these tools ignore to support dynamic features that recommended on the context of smart homes mobile apps. Users that use IoT appliances for everyday life need special designs such as dynamic design to support many behavior patterns. Appliances and software are working together as a cycle in the same progress to handle several daily tasks. Machine learning such as Naïve Bayes now a day used for solving many significant issues for various fields of business. We present a novel model as cloud-based for a dynamic design for supporting users demand to control appliances on smart homes. This adaptive context of UI consist of plastic icons can re-ranked near finger to realize customer need as fast responsive for ambiguous daily tasks. Our approach can support many patterns of behaviors that used for trigger and complete daily tasks on smart homes. Naïve Bayes as a lightweight machine learning algorithm that trained by daily used patterns to suggest specific task that more used at this time of weeks and seasons. As a result of work, such an approach will support usability design for any kind of users even who not familiar with mobile apps or facing memorial problems.*

Key Words: **Naïve Base, HCI, Usability design, dynamic design, behaviors patterns.**

1. INTRODUCTION

Nowadays there are a lot of tools kit for UI/UX design for mobile apps even emotional icons available for smart homes. These tools are tested enough with many UI interface experiences to heuristic the usability, some of these are provided for free. For example, the following designs are fetched from (https://www.pinterest.com/) which provide many interface design for smart home apps.

Fig -1: Home design Interface

These tools consider most of the principles of usability according to HCI that we studied in our previous paper [1]. Unfortunately, these tools neglect adaptively that needed for dynamic interface design which has significant and recommends for ambiguous tasks on context such as smart home apps [2]. The user need on smart home apps needs more facilities and features that support much behavior that realize everyday life requests such as switching on TV for a specific channel, collaborating AC, electricity outlets, refrigerators and etc. [3]. Some of these home appliance a period of time to finish its task such as garage door, microwave and etc. The mobile apps design should consider this period of time to complete progress with the devices this technology available nowadays as IoT devices, service is installed are running all time on these machines to handle communications with the internet or any other local network. This service has the ability to revoke and composed with any other application such as mobile apps to tackle such progress [4].

Adaptive and dynamic UIs are responsive of their used context and are able of provided that a response to deviations in this context or layout, by adapting one or more of their features according to of rules that either predefined or inferred by machine learning model training over time [5]. For example, dynamic UI can automatically scale the text size for the customer in visual enhancements. On the other hand, the dynamic UI can able users to by hand adapt desired features such as adding and removing re-rank emotional buttons. such UIs semi and fully automated adaptation, it can be more practical for complex and ambiguous apps such as smart homes than adaptable ones who need manual adaptation mostly due to weeks and reasons [6]. Such design does not know in design time even known it needs more context layouts which will increase the ambiguity of application.

The increased need for user adaptively, convoyed by an increase in the possibility of comprehending adaptive apps due to advances in [7]. The dynamic design can provide benefits in definite cases such as humanizing user enactment. Several research has stated specific settings that influenced the characteristic of dynamic UIs. providing plastic capabilities to the app depending on multiple of behavior such as a pattern of the task for the user. For example, users in the smart home app who need to perform complex tasks can get the benefit of automated adaption and suggested a short cut for a series of steps can be done at one click. Another significant feature for a dynamic UIs that can keep the area of the layout for recommended tasks by their times [8]. We present mobile app design that realizes the usability for convenient, reliable and interactive design this approach depending on machine learning model such as naïve base that trained by the behavior of the user on the app daily with time stamp, so such model can predict which the adaption layout needed later. The causes of use naïve Bayes is the lightweight of resources and lazy trained which means can be produced a good result in early uses of the mobile app. Naïve Bayes is a classification a term in Bayesian statistics dealing with a simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions.

The paper is organized as follows. Section 2 describes the dynamic design. Section 3 presents a training model. Sections 4 describes some of related works .5 consist of conclusion and future works.

2. DYNAMIC DESIGN

As shown in figure 2. We propose the main layout of the smart home app context consist of two parts. The upper part will contain main emotional icons for main features such as a close app, electrical appliance, music devices, locks, waters, adjusting AC ..etc. for example, when the user touch button of electrical appliance it will open sub activity which contains all electronic device on home each one has own activity contains the appliance features and its configuration. The progress bar will start when trigger device running till finish the task. The emotional buttons in sub activity of lamb features don't need progress because it will appear as it's physically any change of these buttons will change the logical button on activity.

All emotional buttons weather in the main layout or in sub activity will re-ranked and adaptive according to the training model. Re-ranked means the place of the button will be changed automatically near the used finger of the user. This adoption of the button should not be done on design time, the layout of buttons can be set on TableView and this table will be adapted each time on running time according to the training model. This adapted TableView will responsive according to the sticky service that working all time on the background to communicate with the cloud to get optimal order of buttons as well as interaction with appliance service to display the task progress.

The second part of the main layout consists of scrolled listView that contains recommended task needed at this time. This suggestion will be ordered on time and according to the training model. Each item of suggested list has its own progress down display the status of appliance and does not need to be accessed to sub activity because it shortcuts for a series of steps that predicted form dynamic model which depend on a machine learning model that residence on cloud and directly feed the apps in real time.

Figure 3,4 show a part of submenus appliance setting each menu consist of progress a setting for configuration (e.g. door lock) in figure 3 display the door raises up and down at the same time when real door status is changed (Augmented Reality) this technique works as a communication establish between appliance service and Mobile app both of programs working to gathers to give the user a real state of appliances changes.

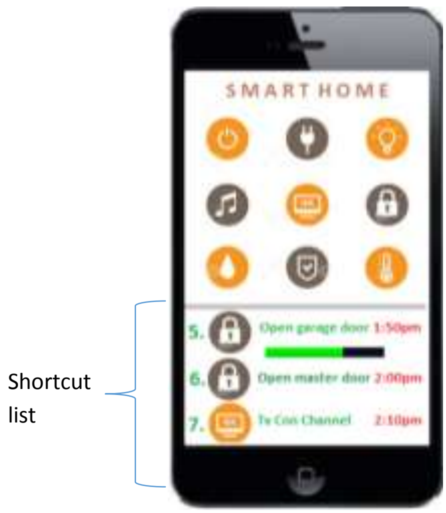
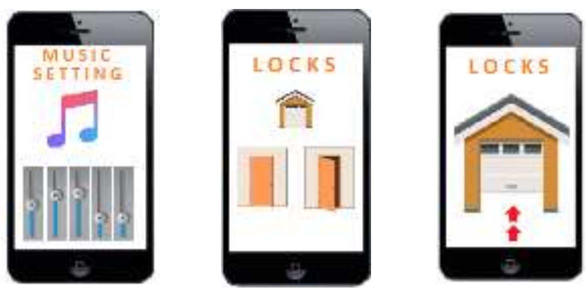


Fig -2: Sample design of main Layout



(1) (2) (3)

Fig -3: Design collections submenus

3. TRAINING MODEL

As shown in figure 5. Training model consists of naïve Bayes algorithm that trained by dataset, this dataset is contains selected features in which activity, for which appliance and the progress time that takes to finish the task and time stamp of the requested feature.

Table 1 show apart of dataset proposed to train the adoptive the design consist of 8 attributes of user behaviors on the interface, rank is the data class attribute, while setting attributes represent the configuration of that appliance needed from the user at the time stamp.

Period per (hh: mm) this attribute clarify for how many time this appliance is used, while no of Guests describe how many guests or people now at home. The time stamp is a vital attribute for predict class in our dataset. The class attribute which is rank transfer from cloud regularly to the app when according to the training process in this time. The shortcut list re-ranked according to the choice of the same user in the past, these choices transferred to the cloud for training purpose. Our proposed design should be adapted to this list according to customer need. So this list of shortcuts will change each period of time according to the existence of the user in the home in each period of time the status of the user depending on his place will be updated to re-ranked according to the user may need at this time.

(4) (5) (6) (7)

Fig -4: Submenus Appliance Setting

These features composed a series of steps on apps which predefined as a class. The main variable of the dataset that used as input is progress time and timestamp, while the output is the composed series of steps or buttons that lead to the task which is rank.

Table -1: Apart of dataset used in Training

Appliance	Setting	Period per (hh: mm)	NO. of Guests	Place of user	Login	Time stamp	Rank
TV	A	2:20	3	Living room	User1	20:05	1
light	C	5:22	5	office	User1	19:10	1
Micro Wave	D	1:00	2	kitchen	User1	10:00	1
light	C	4:00	4	Bed room 1	User1	19:10	2
AC	A	10:30	5	Bed room1	User1	8:00	3
Mater door	A	00:05	3	garden	User1	06:00	1
Main door	A	00:05	3	entrance	User1	05:00	1
AC	B	10:30	5	Living room	User1	8:00	3
light	C	4:22	4	Bed room 1	User1	19:30	1
TV	A	2:20	3	Living room	User1	20:05	1
TV	A	2:20	3	Living room	User2	20:05	1
light	C	5:22	5	office	User2	19:10	1
Micro Wave	D	1:00	2	kitchen	User3	10:00	1
light	C	4:00	4	Bed room 3	User3	19:10	2
AC	A	08:30	5	Bed room2	User2	8:00	3
Mater door	A	00:05	3	garden	User1	06:00	1
Main door	A	00:05	3	entrance	User2	05:00	1
AC	B	7:12	5	Living room	User1	8:00	3
light	C	5:22	4	Bed room 2	User2	19:30	1
TV	A	4:20	3	Living room	User1	20:05	1

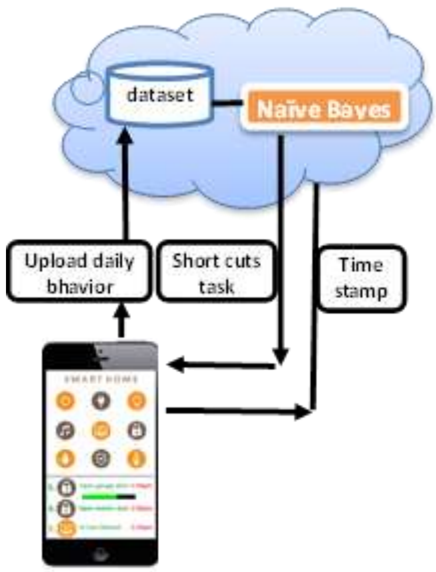


Fig -5: Training Model

The Naïve Bayes model constructed on cloud and connected to the mobile app using JSON technology APIs. This API's allow Mobile app access the dataset for adding new behavior pattern, request to update the suggested list and request for re-ranked the buttons to be in the nearest place of the finger. The dataset on the training model is limited to growth once its reach to the threshold will overwrite the oldest occurred instances. The occurred instance only overwrite as not to miss significant seasonal patterns. Otherwise, The dataset continues growth by new behavior patterns and their progress time and timestamp.

In each time user interact with design the app send complete task steps as behavior pattern to the Naïve Bayes on the cloud and send the current time for training to infer the recommended pattern, the suggested list will be updated according to the output of Naïve Bayes model.

4. RELATED WORKS

[9] Present a Model-Driven method to generate dynamic and adaption context of UIs according to the user profile. This model uses the KNN algorithm to predict the class of next behavior for this user on the web application, the system will be adopted related to the result from the algorithm. The proposed model doesn't separate the training model about the engine of the web application which will influence the performance of the system. The second issue that the user profile continue growth without limitation of dataset size.

The following efforts have been interesting in finding standard solutions adapting UIs to different arrangements of contexts [10, 11] [12] without using machine learning. These efforts focus on the dynamic in design time and ignore running time.

[13] present the importance of extensibility for generic dynamic interface frameworks. Though, just a little awareness has been targeted to open frameworks which obviously intent at attractive other external experts to cooperate in the modification and extension of used adaptation methods and rules.

5. CONCLUSION AND FUTURE WORK

To realize usability in ambiguous behavior environment such as smart home, that need dynamic user interface design which is adaptive according to the occurrence of user tasks in the timestamp. In this paper, we provide a novel approach that consist of dynamic design for mobile apps adoptive on running time depending on machine learning model constructed on the cloud that trained by dataset which is collected by user behavior on apps. We use Naïve Bayes classifier which is a lightweight model and can be trained by little instances, so the mobile app will response suggested list for the user in early time after the app where deployed. We believe such a model will support usability, convenient, reliable and interactive design. The part of our future work is evaluated such design in a real environment.

ACKNOWLEDGEMENT

This work was carried out at the College of Computer, Qassim University.

REFERENCES

[1] The Usability of HCI and Design Principle in Smart Home. 2019.

[2] Tonkin, E., et al., Talk, Text, Tag? Understanding Self-Annotation of Smart Home Data from a User's Perspective. Sensors, 2018. **18**(7): p. 2365.

[3] Ghajargar, M., M. Wiberg, and E. Stolterman, Designing IoT systems that support reflective thinking: A relational approach. International Journal of Design, 2018. **12**(1): p. 21-35.

[4] Caivano, D., et al., Supporting end users to control their smart home: design implications from a literature review and an empirical investigation. Journal of Systems and Software, 2018. **144**: p. 295-313.

[5] Märtin, C., S. Rashid, and C. Herdin. Designing responsive interactive applications by emotion-tracking and pattern-based dynamic user interface adaptation. in International Conference on Human-Computer Interaction. 2016. Springer.

[6] Troyer, M., et al., Systems and methods dynamic localization of a client device. 2016, Google Patents.

[7] Bouchelligua, W., et al. An MDE approach for user interface adaptation to the context of use. in International Conference on Human-Centred Software Engineering. 2010. Springer.

[8] Zouhaier, L., Y.B. Hlaoui, and L.J.B. Ayed. A model driven approach for improving the generation of accessible user interfaces. in 2015 10th International Joint Conference on Software Technologies (ICSOFT). 2015. IEEE.

[9] Ghaibi, N., O. Dâassi, and L.J.B. Ayed, User Interface Adaptation based on a Business Rules Management System and Machine Learning. 2018.

[10] Thevenin, D. and J. Coutaz. Plasticity of User Interfaces: Framework and Research Agenda. in Interact. 1999.

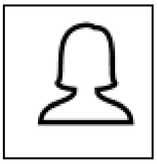
[11] Calvary, G., et al. Plasticity of user interfaces: A revisited reference framework. in In Task Models and Diagrams for User Interface Design. 2002. Citeseer.

[12] Bacha, F., K. Oliveira, and M. Abed, Using context modeling and domain ontology in the design of personalized user interface. International Journal on Computer Science and Information Systems (IJCSIS), 2011. **6**: p. 69-94.

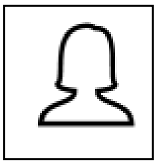
[13] Genaro Motti, V. A computational framework for multi-dimensional context-aware adaptation. in Proceedings

of the 3rd ACM SIGCHI symposium on Engineering interactive computing systems. 2011. ACM.

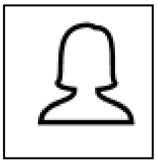
BIOGRAPHIES



Amirah Alshammari: a master's student in the Informatics department at Qassim University, Saudi Arabia.



Haneen Alhadeaf: a master's student in the Informatics department at Qassim University, Saudi Arabia.



Sabreen Alrasheedi: a master's student in the Informatics department at Qassim University, Saudi Arabia.