Pers Ubiquit Comput (2011) 15:857–870
DOI 10.1007/s00779-011-0401-5

ORIGINAL ARTICLE

Adapting the mobile phone for task efficiency: the case of predicting outgoing calls using frequency and regularity of historical calls

Osama O. Barzaiq · Seng W. Loke

Received: 29 November 2010 / Accepted: 10 May 2011 / Published online: 11 June 2011
© Springer-Verlag London Limited 2011

Abstract The set of functionalities provided by advanced mobile phones is significantly increasing. However, the small size of mobile phone user interfaces makes it difficult for the user to deal with this large number of functionalities, which could reflect negatively on user performance and the efficiency of mobile phone functionalities. In this paper, we designed and developed an adaptive task-based functionality called ATF on mobile phones, where the task we focused on was to predict the next contact that the user is most likely to call. Furthermore, we conducted comprehensive evaluation of our approach. We show that our approach can successfully predict contacts that a user will most likely call next. Our results uncover the frequency and pattern of regularity in making mobile phone calls and suggest promising avenues for future work for optimising tasks (beyond phone calls) performed with the mobile phone.

Keywords Adaptive user interface · Mobile phone · Phone usage · Call prediction function

1 Introduction

In recent years, the presence of the mobile phone in an individual's life has become essential and one of the requirements of contemporary life [1]. Furthermore, mobile phones provide a large number of functions and services, which allow end-users to access information anytime and anywhere, as well as the different forms of communication they offer. Also, mobile phones are used by a large number of people with different attributes (e.g. gender, age and profession), and for different purposes [2]. However, some usability issues have been raised regarding the small size of the mobile phone user interface. One of the issues is the difficulty in displaying all the mobile phone's functions on one screen, which results in a large number of different screens that the user has to deal with [3]. Whereas the main purpose behind the increasing number of mobile phone functions is to increase the usability of the mobile phone, the size restriction may hinder this goal and cause an increase in the complexity of the interface.

Several approaches were proposed to improve mobile phone user interfaces by developing a new type of menu structure based on the concept of task computing [4]. The approach in [4] includes designing and developing a static task-based user interface in order to facilitate access to the most frequently used functionalities and to suit the requirements of mobile phone users. Other work presented ideas to improve the usability of the small screens of mobile phones by developing new prototypes of adaptive user interfaces [5]. The concept of an adaptive user interface has been defined by [6] as follows: 'An adaptive user interface is a software artefact that improves its ability to interact with a user by constructing a user model based on partial experience with that user'.

Accordingly, our aim is to enhance the usability of mobile phone user interfaces through the integration of the concept of task computing and the concept of adaptive user interfaces. We designed and developed an adaptive task-based functionality on mobile phones, where the task we focused on was to predict the next contact that the user is

O. O. Barzaiq · S. W. Loke (✉)
Department of Computer Science and Computer Engineering, La Trobe University, Bundoora, VIC 3086, Australia
e-mail: s.loke@latrobe.edu.au

O. O. Barzaiq
e-mail: oobarzaiq@students.latrobe.edu.au

most likely to call. The designed adaptive task-based functionality is called ATF. ATF contains a user interface that displays a list of the most frequently and regularly dialled numbers, and then, the mobile phone user can select any number from that list and dial the selected number. The dynamic contact list is generated after a deep analysis of outgoing call records, which are stored in the Log application of the mobile phone to determine the frequency and the regularity of outgoing calls for each contact stored in the Phone Book application. The main benefit of the dynamic contact list is to reduce time wasted in searching for a contact number in the Phone Book application and to reduce the number of key presses that are required to perform a phone call to only two key presses; one key press to display the dynamic contact list and another key press to dial the selected number. The rest of the paper is organised as follows. Section 2 reviews the related work. Section 3 presents our approach. Section 4 describes the implementation and the evaluation of our approach. Section 5 presents the conclusion.

2 Related work

Several approaches have been discussed in order to address the complexity of mobile phone user interfaces. Some of these approaches focused on the difficulties that the mobile phone user may find in dealing with the current generation of mobile phones. Various suggestions to develop an enhanced version of a static user interface based on the different interests of mobile phone users were put forward. Furthermore, researchers discussed how the appropriate category classification and unambiguous item labelling can facilitate the interaction between the user and the mobile phone user interface [3, 7, 8].

Bridle and McCreath [9] noted that the majority of mobile phone users tend to use certain functions most of the time. These researchers conducted an experiment to find the most frequently used functions in order to create shortcuts to those functions on the main menu, which may result in saving time and improving the user performance in using mobile phones. However, they focused only on voice-calls and text-message applications. Creating shortcuts to only two mobile phone functionalities does not resolve the problem of handling the large number of mobile phone functionalities that the user has to deal with.

Loke and Almagrabi [4] introduced a prototype of a task-oriented menu, which was designed to meet the requirements of mobile phone users who can be divided into different groups in terms of age, job, etc. The main goal of their prototype is to ease access to the most frequently used applications and to suit the requirements of mobile phone users [4]. However, they divided the mobile phone users into seven different groups, and because of this, seven different task hierarchies had to be designed to suit the requirements of each group of users, which raises the following question.

The last group of approaches suggested improving the previous idea of using a static task-based user interface by developing a dynamic user interface, which can adapt and modify its structure according to the user's interests in order to enhance the usability of mobile phone user interfaces. Their dynamic user interface observes and analyses user behaviour in order to generate a list of the most frequently used functions [1, 2, 10]. Therefore, learning from the above work, in our approach, we apply the concept of adaptive user interfaces to automating contact selection in phone calls.

3 Solution design

Our main objective is to improve the efficiency when making mobile phone calls by highlighting the frequency and the regularity of outgoing calls for each contact and then predicting the next contact that the mobile phone user is most likely to call.

We defined a prediction function that can generate a dynamic contact list of the most frequently and regularly dialled numbers, after a comprehensive analysis of the outgoing call records, in order to find the frequency and the regularity of phone calls with each contact stored in the Phone Book application. In addition, the prediction function uses Multi-Attribute Utility Theory (MAUT) [11] to predict the next phone number that is most likely to be dialled by the user. Consequently, MAUT calculates the weight of each contact based on the following questions:

(a) How many records does he/she have in the data set?
(b) Has his/her number been dialled at least one time every month?
(c) Has his/her number been dialled at least one time every day?

We call these the three dimensions of our prediction function. Then, after calculating the weight of each contact, MAUT will rank contacts in descending order, based on their weights. Accordingly, the next contact that is most likely to be called is the contact that has the highest weight.

The prediction function has the following three stages:

1. The year condition:
   The outgoing call records that were recorded in the current year and the previous year are the only records that will be analysed. The reason for including the 'year' condition is to reduce the number of records that will be examined and to exclude any contact that has not been called by the mobile phone user for 2 years or

more. Reducing the number of analysed records will ease the load on the mobile phone processor when calculating the weight of each contact, which will increase the efficiency of the prediction function.

2. The Contact Dimensions:
In order to calculate the weight of each contact, the following dimensions have been defined:

(a) *The Counter Dimension*: In this dimension, the number of records stored in the 'Phone Calls Log' data set will be counted in order to calculate the weight of the Counter Dimension for each contact. The following formula will be used to find the weight of the Counter Dimension for each contact:

$$\text{The weight of the Counter Dimension of contact}(X) = (C(X)R \times \text{CDW})/\text{TNR}$$

where $C(X)R$ = the number of records belonging to contact (X) in the 'Phone Calls Log' data set, which have been recorded in the current year and the previous year, CDW = the Counter Dimension relative weight, which is suggesting the relative importance of the Counter Dimension between dimensions, and is a constant value = 0.2 (the weight of the Counter Dimension is small because the Counter Dimension is only counting the outgoing call records that have been made to a contact regardless of when or how regular), and TNR = the total number of records in the 'Phone Calls Log' data set, which have been recorded in the current and previous year.

(b) *The Every Month Dimension*: In this dimension, the outgoing call records will be analysed in order to find the answer to the following question: 'Has his/her number been dialled at least one time every month?' Based on the answer to the previous question, the weight of the Every Month Dimension for each contact will be calculated by using the following formula:

$$\text{The weight of the Every Month Dimension of contact }(X) = (C(X)\text{MC} \times \text{EMDW})/\text{TNM}$$

where $C(X)$MC = the contact(X) Monthly Calls, which is the number of different months where the mobile phone user made at least one phone call to contact(X) in the current year and the previous year, EMDW = the Every Month Dimension Relative Weight, which is suggesting the relative importance of the Every Month Dimension between dimensions, and is a constant value = 0.3 (the weight of the Every Month Dimension is slightly bigger than the weight of the Counter Dimension because the Every Month Dimension is trying to capture the frequency and the regularity of calls for each contact on a monthly basis), and

$$\begin{aligned}\text{TNM} &= \text{the Total Number of Months}\\ &= 2 \text{ years} \times 12 \text{ months/year}\\ &= 24 \text{ months (a constant value)}\end{aligned}$$

(c) *The Every Day Dimension*: In this dimension, the outgoing call records will be analysed in order to find the answer to the following question: 'Has his/her number been dialled at least one time every day?' Based on the answer to the previous question, the weight of the Every Day Dimension for each contact will be calculated using the following formula:

$$\text{The weight of the Every Day Dimension of contact}(X) = (C(X)\text{DC} \times \text{EDDW})/\text{TND}$$

where $C(X)$DC = the contact(X) Daily Calls, which is the number of different days where the mobile phone user performed at least one phone call with contact(X) in the current year and the previous year, EDDW = the Every Day Dimension Relative Weight, which is suggesting the relative importance of the Every Day Dimension between dimensions, and is a constant value = 0.5 (the Every Day Dimension has the biggest weight because it is trying to capture the frequency and the regularity of calls for each contact on a daily basis), and

$$\begin{aligned}\text{TND} &= \text{the Total Number of Days}\\ &= 2 \text{ years} \times 365 \text{ days/year}\\ &= 730 \text{ days (a constant t value)}\end{aligned}$$

3. *From 1 and 2*, using a Multi-Attribute Utility Theory (MAUT) approach, we can now calculate the weight of contact(X) as follows, based on calls during the current and previous years:

$$\begin{aligned}&\text{The weight of contact}(X)\\ &= \text{the weight of the Counter Dimension of contact}(X)\\ &+ \text{the weight of the Every Month Dimension of contact}(X)\\ &+ \text{the weight of the Every Day Dimension of contact}(X)\end{aligned}$$

The weight of each contact can be calculated by using the above formula and then ranked.
Deployment: We developed an adaptive task-based functionality called 'ATF' on mobile phones. The integrated development environment of NetBeans and J2ME (Java 2 Micro Edition) was used to develop ATF. The main screen of ATF includes the following choices:

(a) '*Make a Call*': If the user selects this choice, another screen will appear, offering three methods from which the user may choose to generate a list of contacts. The three methods are as follows:

   i. Using a Counter: In this method, the number of outgoing call records stored in the 'Phone Calls Log' data set for each contact will be counted. Then, a dynamic contact list will be generated in order to display the names and phone numbers of the five contacts who have the highest number of records.
   ii. Using MAUT: In this method, the weight of each contact will be calculated by using the MAUT formula in order to predict the next contact that the user is most likely to call. Then, MAUT will rank the contacts in descending order based on their weights, and it will generate a dynamic contact list. The generated dynamic list will have the names and phone numbers of the five contacts who have the highest weights.
   iii. Displaying all contacts: In this method, all stored contacts in the Phone Book application will be displayed. This method is useful in case the previous methods fail to predict the next phone number that the user is most likely to dial.

(b) '*Phone Usage History*': If the user selects this choice, another screen will be displayed containing weekly and monthly reports about the different tasks and operations conducted by the mobile phone user, such as phone calls, text messages, etc.

We also designed and developed a light version of ATF called ATF Mini. ATF Mini is designed particularly for one goal, which is to display a dynamic contact list generated by MAUT. Moreover, ATF Mini can be added to the main menu of the mobile phone as a shortcut in order to reduce the number of key presses that are required to make a phone call to only two key presses; one key press to display the dynamic contact list and another key press to dial the selected number. Figure 1 shows the structure of ATF (implemented using the JavaME platform).

4 Implementation and evaluation

We installed ATF and ATF Mini on a Nokia N95 8 GB mobile phone in order to test the operational efficiency of our approach and to evaluate the accuracy of the results of the prediction function.

Figures 2 and 3 show ATF and ATF Mini running on the mobile phone.

The following figures show the implementation phases of ATF:

As shown in Figs. 4 and 5, the main screen of ATF contains the following choices:

i. 'Make a Call'.
ii. 'Send SMS Message'—deactivated.
iii. 'Phone Usage History'.

If the user selected the first choice, another screen will appear containing a number of different methods to generate a list of contacts as shown in Fig. 6.

The following figures show two dynamic contact lists, the first one (Fig. 7) was generated by using a counter, and the second one (Fig. 8) was generated by using MAUT.

Figure 9 shows a list of all stored contacts in the Phone Book application, and Fig. 10 shows a screen performing a phone call with the selected contact.

The screen shown in Fig. 11 has 2 options.

(a) *Calls History*: This option displays weekly and monthly reports of the phone calls that were made by the mobile phone user.
(b) *SMS History*: This option displays weekly and monthly reports of the text messages that were sent by the mobile phone user.

ATF can be used to provide simple phone call stats by tracking calls made with ATF.

As shown in Fig. 12, the first choice is Outgoing Calls History, which shows the number of phone calls that were made (Fig. 13). It also displays weekly and monthly reports of all phone calls made by the user as shown in Figs. 14 and 15. The second choice is Contact Number History which displays a list of all contacts stored in the Phone Book application as shown in Figs. 16, 17,18,19,20, 21, 22.

4.1 Solution evaluation—1

We created a syntactic data set that contains a number of records, which are similar to the outgoing call records stored in the Log application of the mobile phone. The reason for creating a syntactic data set is that there is no direct access from a JavaME application to retrieve the registered records in the Log application due to the privacy policies of mobile phones. The syntactic data set is called Phone Calls Log, and it was created by analysing the list of recent outgoing calls. However, because of the small size of the recent calls' list, we used our personal experience with the stored contacts in our mobile phones to estimate the number of phone calls we had made with each contact in order to create a number of records and to add them to the 'Phone Calls Log' data set. Each record stored in the Phone Calls Log data set contains the following

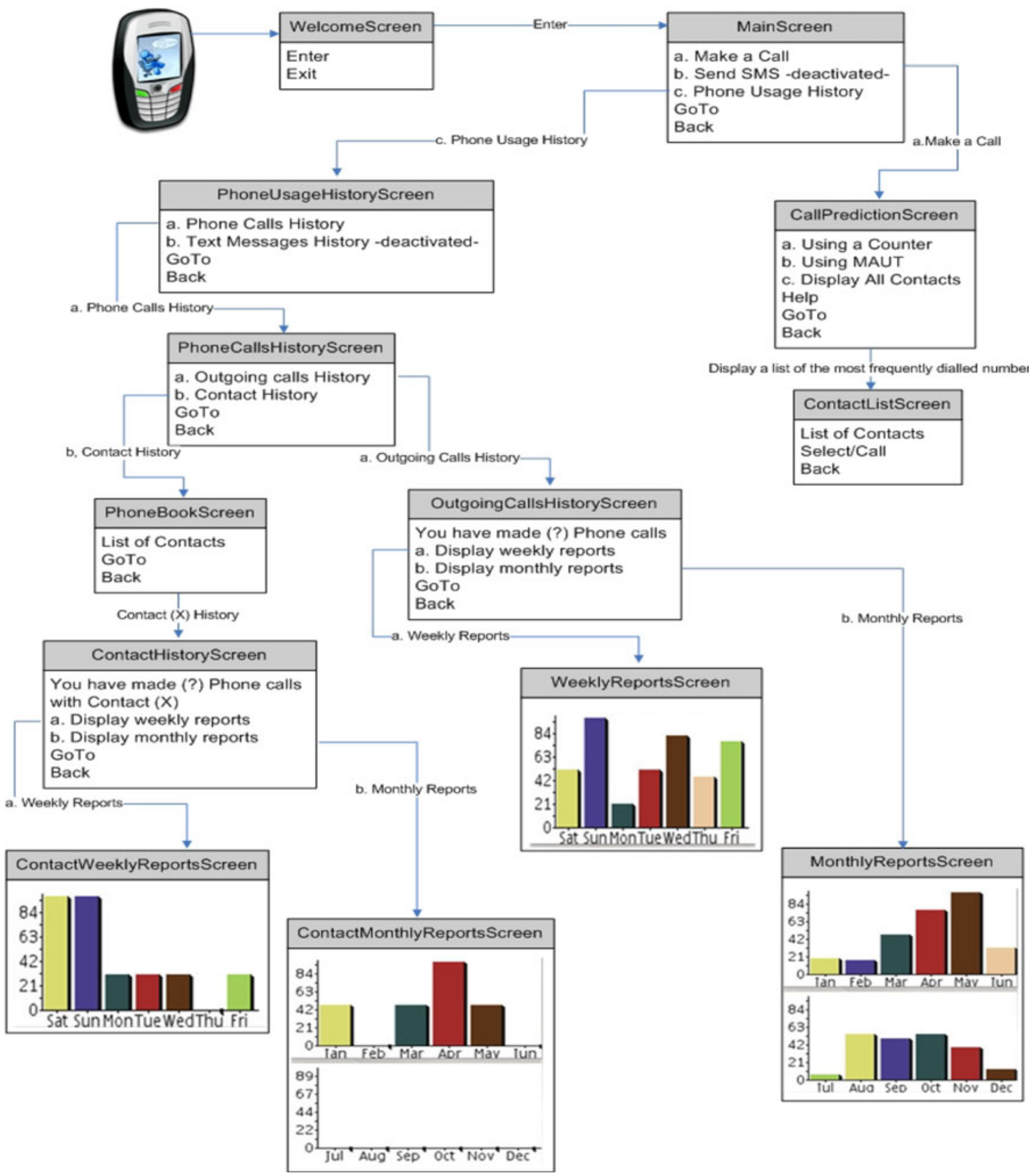

Fig. 1 Structure of the advanced task-based functionality user interface

information: the contact name, the contact number, the call date, the call time and the call duration.

We used the 'Phone Calls Log' data set to test and evaluate the efficiency of ATF and ATF Mini.

4.1.1 The ranking speed of MAUT

In order to predict the next contact that the mobile phone user is most likely to call, MAUT calculates the weight of each contact and then ranks the contacts in descending order based on their weights. Consequently, the contact that has the highest weight is the next contact that the user is most likely to call. Because of limited resources on the mobile phone, a question is whether this process of calculating the weight of the contacts and ranking them would be too slow. Hence, the ranking speed of MAUT must be tested on a real mobile phone. We used the following steps to test the ranking speed of MAUT:

1. We developed a program that can automatically generate a group of records in order to test the ranking speed of MAUT with a different number of records. The speed test program was developed based on the

Fig. 2 The applications folder of the mobile phone

Fig. 3 ATF application running on the mobile phone

premise that the average number of calls per day is 3; and hence, the total number of records of phone calls, which were performed during the period of 2 years can be calculated as follows:

$$3(\text{calls/day}) \times 365(\text{days/year}) \times 2(\text{years}) = 2,190 \text{ records.}$$

2. We installed the speed test program on Nokia N95 8 GB mobile phone, which has the following specifications:

Fig. 4 The welcome screen of ATF

Fig. 5 The main screen of ATF

Fig. 6 The prediction methods of ATF

Operating System: Symbian OS 9.2, S60 rel. 3.1
Memory: 128 MB RAM
CPU: Dual ARM 11, Clock Rate 332 MHz, 3D Graphics HW Accelerator
Figures 23, 24 and 25 show the speed test program running on the mobile phone.

Fig. 7 A dynamic contact list generated by using a counter

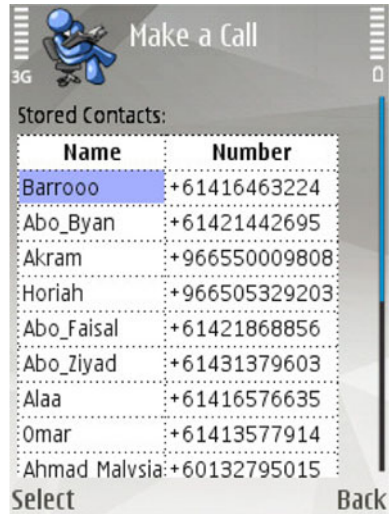

Fig. 9 A list of all stored contacts in the phone book application

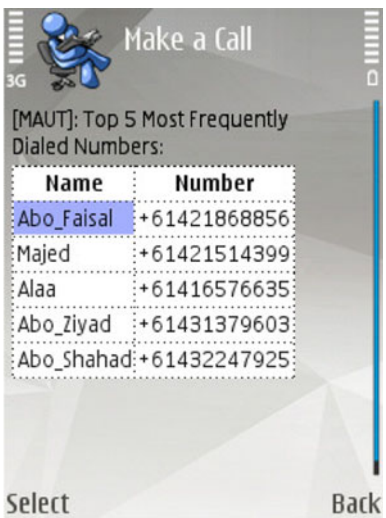

Fig. 8 A dynamic contact list generated by using MAUT

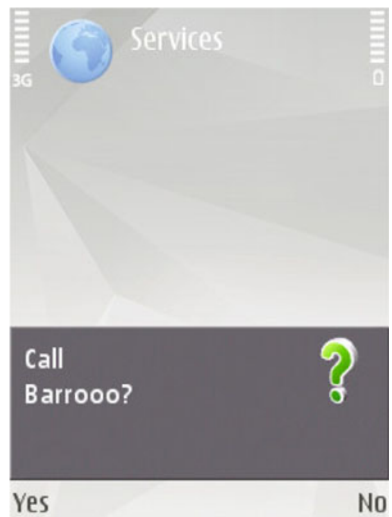

Fig. 10 Performing a phone call with the selected contact

3. We also examined the ranking speed of MAUT with 3,000 records for 3,000 phone calls in order to test the ranking speed of MAUT with the worst possibility for the maximum number of records. Figure 25 shows the option of generating 3,000 records
4. We calculated the ranking speed of MAUT by finding the average of five different reads for each number of records.
5. Figures 26 and 27 show the results of testing the ranking speed of MAUT.

It can be noted from Fig. 26 that the ranking speed of MAUT with 300 records was better than its ranking speed with 200 records. The reason for this is that with 200 records, the mobile phone processor was handling other programs at the same time that MAUT was ranking contacts. Accordingly, the ranking speed of MAUT could be better in the absence of other programs that consume processor capacity.

The Nokia N95 can be considered as a middle level phone at this time. However, we expect with Moore's law that phones will only get more powerful. Hence, the above results show that up to 1,000 records (Fig. 26) our approach is still feasible and fast. Up to 1,000 records in a phone log over 2 years correspond to around 1.5 calls a day on average, which is reasonable for an average user. Figure 27 shows the performance with up to 3,000 records, which is really rather high for a typical user. The results show that our approach is not prohibitively inefficient. Also, since mobile phones can only get more powerful,

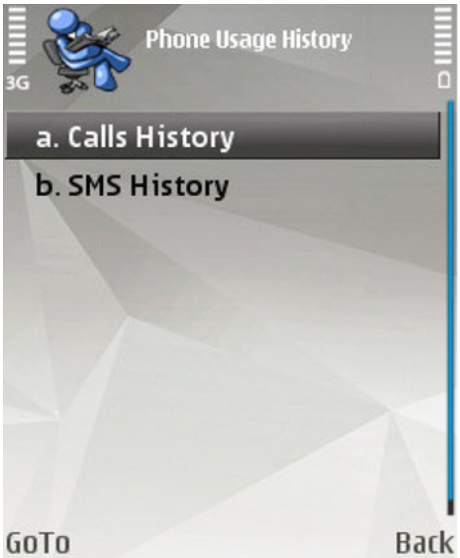

Fig. 11 The screen of phone usage history

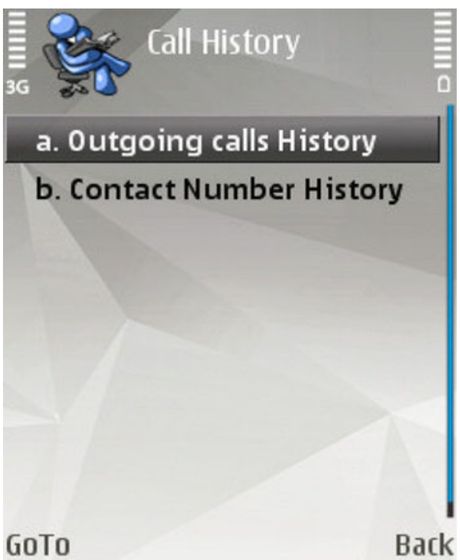

Fig. 12 The screen of phone calls history

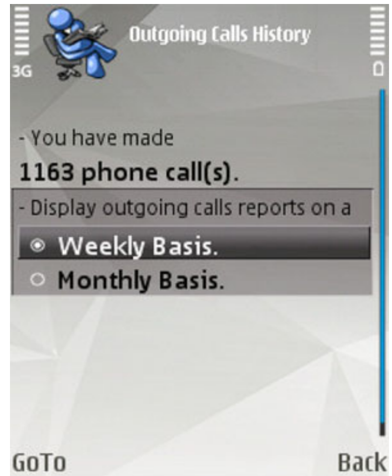

Fig. 13 A detailed report of outgoing calls

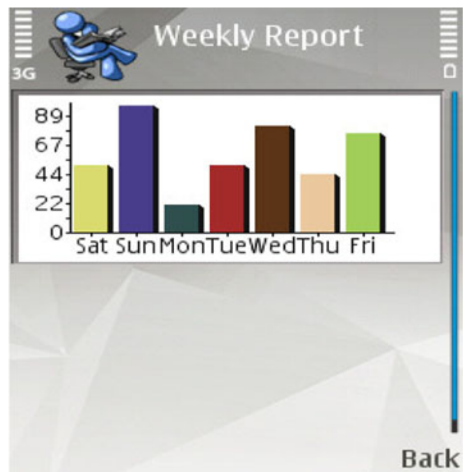

Fig. 14 A weekly report of outgoing calls

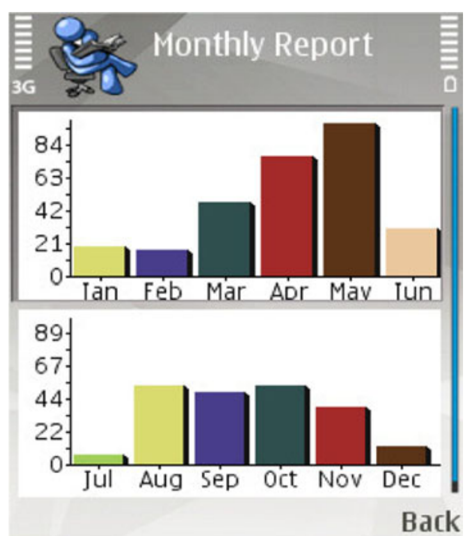

Fig. 15 A monthly report of outgoing calls

these results can be viewed as an upper bound on efficiency or worst case scenarios.

4.1.2 The improvement of MAUT over time

In order to test the improvement of MAUT in the process of predicting the next contact that the mobile phone user is most likely to call, we used the following steps:

1. We installed ATF and ATF Mini on Nokia N95 8 GB mobile phone, testing them for 5 weeks from 17 April 2010 to 21 May 2010.
2. Figure 28 shows the results of testing the improvement of MAUT over time:

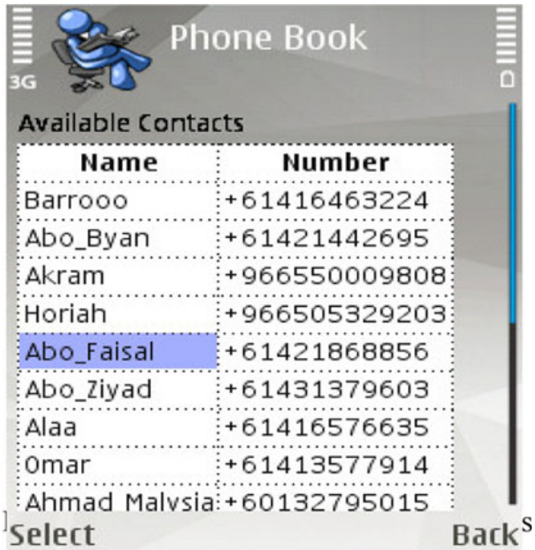

Fig. 16 The contact list of the phone book application

Fig. 17 A detailed report of outgoing calls for a specific contact

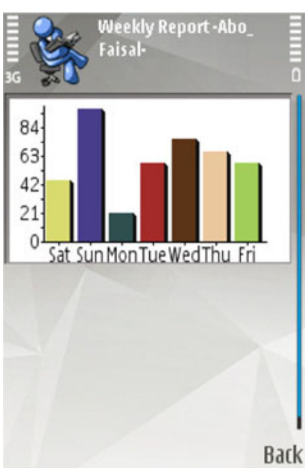

Fig. 18 A weekly report of a specific contact

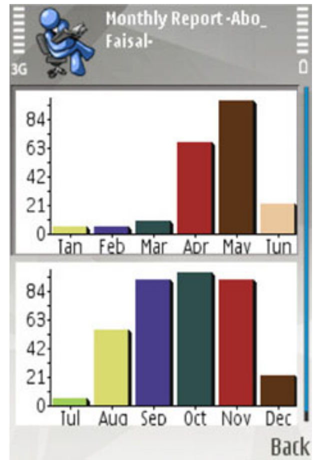

Fig. 19 A monthly report of a specific contact

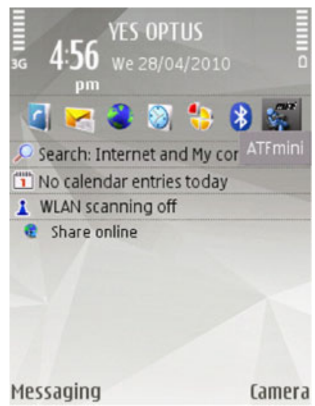

Fig. 20 ATF Mini as a shortcut in the main menu of the mobile phone

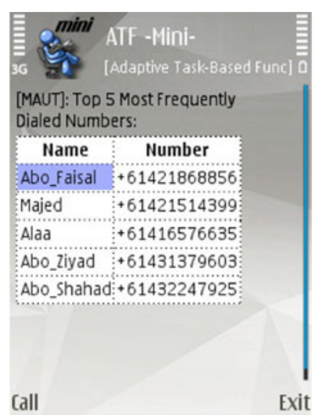

Fig. 21 The MAUT list in ATF mini

We used the following formula to determine the improvement of MAUT each week:

$$\text{MAUT Weekly Improvement} = (n/m) \times 100$$

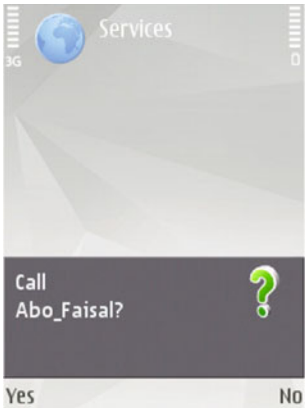

Fig. 22 Performing a phone call with the selected contact

Fig. 23 The main screen of the speed test program

Fig. 24 The screen of generating any number of records

where $n =$ the total number of phone calls that were performed from the dynamic contact list generated by MAUT each week, and

$m =$ the total number of phone calls that were performed each week by using both the MAUT generated list and the original list of the Phone Book application.

Fig. 25 The screen of generating up to 3,000 records

Fig. 26 The ranking speed of MAUT with 100–1,000 records

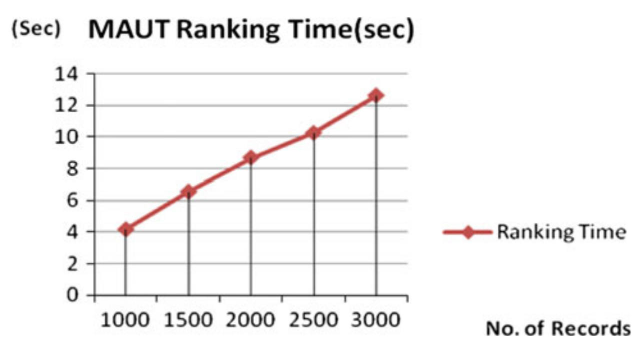

Fig. 27 The ranking speed of MAUT with 1,000–3,000 records

It can be noted from Figs. 28, 29, 30 and 31 that the user used the dynamic contact list of MAUT to perform most of his/her phone calls in the last week of the test and did not need to use the phonebook; this suggests the goodness or accurate prediction of the ranked contact list generated using the MAUT formula. The reason for this is that the process of calculating the weight of each contact is conducted after analysing the regularity and the frequency of phone calls with each contact, and such a process requires enough time in order to obtain accurate results. The results shown in Fig. 28 suggest that the predictions can get better over time; however, further testing could be required as the last section of this paper explains.

4.1.3 The advantage of using more than one factor in the process of generating a dynamic contact list

We used the following steps to test whether the number of dimensions that are used to calculate the weight of each contact can increase the accuracy of the results of the prediction function or not. We used three different methods to predict the next contact that the user is most likely to call. The first method uses only one dimension, the Counter dimension, which counts the number of outgoing call records stored in the Phone Calls Log data set for each contact. Then, a dynamic contact list will be generated to display the names and phone numbers of the five contacts who have the highest number of records and who are expected to be called by the user.

The second method uses two dimensions: the first is the Counter dimension and the second is the Every Month dimension. In this method, the weight of each contact will be the sum of the weight of the first dimension and the weight of the second dimension. Then, a dynamic contact list will be generated in order to display the names and phone numbers of the five contacts who have the highest weights. The last method uses the MAUT formula, which has all the three dimensions shown earlier to determine the weight of each contact. The three dimensions are the Counter dimension, the Every Month dimension and the Every Day dimension.

Then, we calculated the standard deviation in the weights of contacts of each method as shown in the following figures:

From the previous figures, we can note the following:

(a) *In the first method* that uses only one dimension, the curve is the most broad. The reason for this is that the first method uses only the count of calls to a contact regardless of when or how regular, and so, it experiences difficulty in capturing the regularity of calls for each contact; and hence, a significant number of contacts have the same or similar weights.

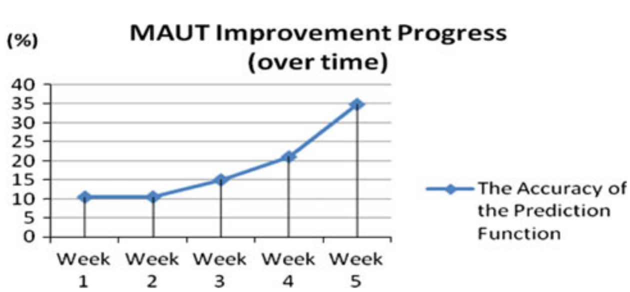

Fig. 28 MAUT improvement over time with synthetic dataset

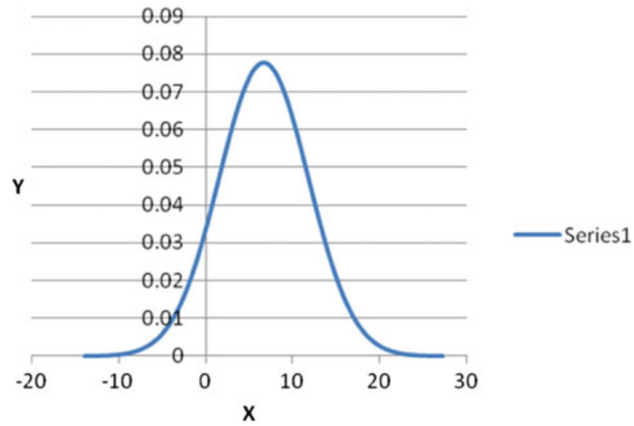

Fig. 29 The standard deviation of weights of contacts in the first method (with one dimension)

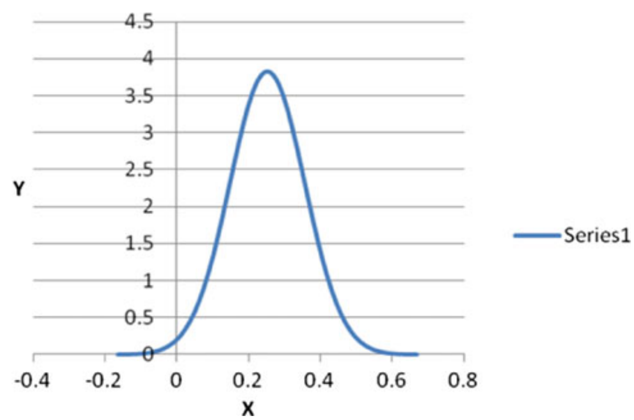

Fig. 30 The standard deviation of weights of contacts in the second method (with two dimensions)

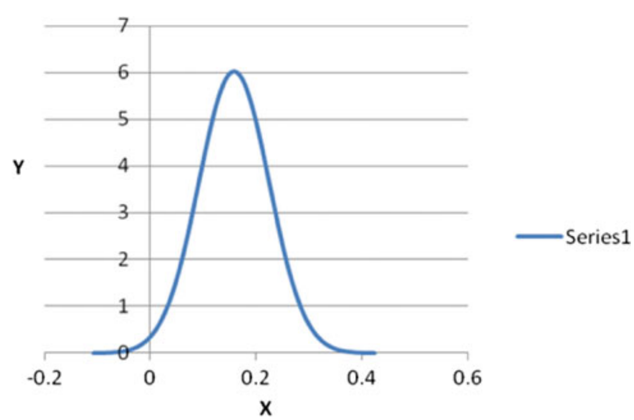

Fig. 31 The standard deviation of weights of contacts in the third method (with three dimensions)

(b) *The second method* that uses two dimensions was better able to distinguish differences in importance of each contact for the mobile phone user. Therefore, the curve of the second method is not as broad as the curve of the first method.

(c) *The third method* that uses MAUT with three dimensions has the narrowest curve compared with the curves of the previous methods. The reason for this is that the third method was the best in distinguishing among contacts, which contributed to improving the process of finding the regularity of calls for each contact which, in turn, contributed to increasing the usefulness of the calculated weight of each contact. Hence, the results show that using all three dimensions is best for distinguishing between contacts and takes into account not just frequency of calls to a person, but when and how regular.

4.2 Solution evaluation—2

We also conducted an objective evaluation of our approach using a real data set of a phone log available from MIT's reality mining project.[1] After removing duplicates, we used a data set with 5,484 records (including incoming and outgoing calls) and 1091 outgoing phone call records over 9 months. Our previous synthetic data set had 1143 outgoing phone call records and show strong similarity with this real data set (increasing our confidence in the synthetic data set results). With a data set spanning over 9 months, we used our MAUT approach to predict the phone calls made in the first day of the first month, on the first day of the second month, on the first of the third month and so on, till the first day of the ninth month. We found that the contacts called in the actual data set on the first day of each month corresponded to the contacts called as predicted by our MAUT approach with increasing accuracy (measured as a percentage of the total number of correct MAUT predicted calls over the total number of actual call made), as given in Fig. 32, which shows the MAUT technique improves in prediction accuracy over time, similar to Fig. 28. There is a main difference in the first 3 months when the MAUT approach deteriorated due to calls being made to new contacts being added to the contact list. Expected improvements then followed as calls were made to existing contacts in a frequent and regular way, as we predicted using the MAUT approach.

4.3 Analysis and discussion

We demonstrated the successful implementation of our programs (ATF and ATF Mini) by using a detailed explanation for each service that ATF and ATF Mini provide. Furthermore, we presented the tests that we conducted to evaluate the efficiency of our approach based on the following points:

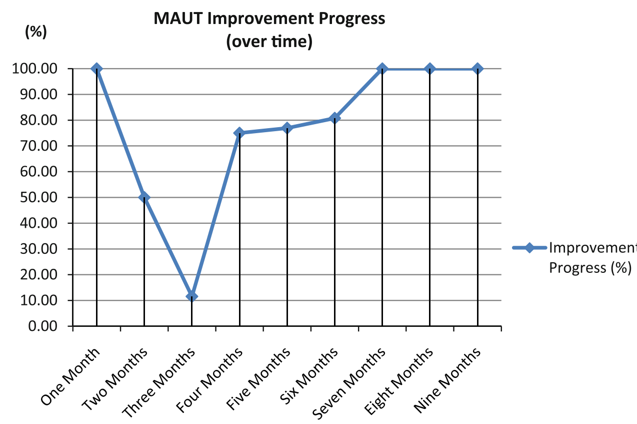

Fig. 32 MAUT improvement over time with MIT's real dataset

(a) The ranking speed of MAUT
(b) The improvement of MAUT over time (with real and synthetic data sets)
(c) The advantage of using more than one factor in the process of generating a dynamic contact list

After analysing the findings of the previous tests and evaluations, we found that the significant success of the prediction function in the process of predicting the next contact that the mobile phone user is most likely to call allows different aspects to be considered in the prediction in order to improve the efficiency of mobile phone functionalities.

5 Conclusion and future work

In this paper, we introduced a novel approach that differs from all previous approaches that have been designed to enhance the usability of mobile phone user interfaces. Our approach includes a definition of a prediction function that uses Multi-Attribute Utility Theory (MAUT) [11] to generate a dynamic contact list to display the most frequently and regularly dialled numbers by calculating the weight of each contact based on the following questions:

(a) How many records does he/she have in the data set?
(b) Has his/her number been dialled at least one time every month?
(c) Has his/her number been dialled at least one time every day?

After analysing the findings of the experiments with our approach, we found significant success in our prediction function in improving the efficiency of the task of performing a phone call. The MAUT-based prediction function enables different aspects of the prediction function to be considered in order to improve the efficiency of mobile phone functionalities (even beyond phone calls), which

[1] http://reality.media.mit.edu/download.php.

could enhance the usability of mobile phone user interfaces and user performance. Hence, not just frequency but regularity should be factored in for predicting user behaviour. Moreover, our approach runs efficiently and is fairly lightweight in terms of computational resources. Future directions are as follows.

5.1 Using location dimension

It is possible to use the geographic location of the mobile phone user in the process of calculating the weight of each contact in order to increase the accuracy of the process of predicting the next contact that the user is going to call. For example, suppose the following: there is someone named Paul who lives for 9 months in Australia to study and then returns to his homeland in the summer vacation for 3 months. It can be noticed that during the time Paul is in Australia, most of his phone calls are performed with people who live in Australia. On the other hand, when Paul returns to his homeland, most of his phone calls are performed with people who live in his homeland. *Adding Attributes to Dimensions*. In order to improve the process of calculating the weight of each contact, a number of attributes can be added to the dimensions used in the prediction function. For example, the 'Call Duration' attribute and the 'Number of Daily Calls' attribute can be added to the 'Every Day' dimension, in order to increase the accuracy of the process of capturing the frequency and the regularity of phone calls for each contact.

5.2 Testing with more users and implementation on other types of phones

Further testing can also be conducted to show how the prediction function improves over time, and on other types of mobile phones (e.g. Android).

5.3 Received calls

Received calls should be considered in the process of predicting the next contact that the mobile phone user is going to call. For example, if someone calls the mobile phone user more than once per day, there is a high probability that the user is going to make a phone call with that person as a response to his/her repeated calls.

5.4 The use of multithreading to generate the dynamic contact list

When the mobile phone user makes a phone call, the process of generating the dynamic contact list can be conducted as a background process through the use of multithreading technique. Accordingly, the dynamic contact list will be ready for use immediately when the user wants to make a phone call, and during the new phone call, the dynamic contact list will be regenerated again in the background without interrupting the current phone call.

5.5 The relative weight of dimensions

The relative weights of each dimension used in the prediction function were based on our believe that these weights are reasonable (based on initial anecdotal studies) to represent the role of each dimension in the process of predicting the next contact that the mobile phone user is most likely to call (and our experiments show that this choice of relative weights works well). However, further investigations could explore other relative weights.

5.6 Using the prediction function with other mobile phone functionalities

The prediction function can be used with other mobile phone functionalities in order to facilitate the interaction between the user and the mobile phone, which could enhance the user's performance and improve the efficiency of mobile phone functionalities. For example, the prediction function can be used with the 'Text Message' functionality in order to predict the next contact to whom the user is going to send a text message. As another example, the prediction function can be used with the 'Media Player' functionality in order to predict the next song that the user is going to play.

References

1. Bridle R, McCreath E (2005) Predictive menu selection on a mobile phone. In: Workshop W7 on mining spatio-temporal data, Portugal
2. Kamisaka D, Muramatsu S, Yokoyama H, Iwamoto T (2009) Operation prediction for context-aware user interfaces of mobile phones. In: Applications and the internet, SAINT '09 ninth annual international symposium, pp 16–22, July 2009
3. Andersson E, Isaksson I-M (2007) Exploring alternatives to the hierarchical menu structure used in mobile phones. In: Department of computing science. Master of computing science: UMEA University, April 2007
4. Loke SW, Almagrabi A (2009) Hierarchical role-specific task-based user interfaces for the mobile phone. In: Joint conferences on pervasive computing (JCPC), pp 495–498, December 2009
5. Tonder Bv, Wesson J (2008) Using adaptive interfaces to improve mobile map-based visualisation. In: Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology Wilderness, South Africa, ACM, pp 257–266
6. Langley P (1999) User modeling in adaptive interfaces. In: Proceedings of the seventh international conference on user modeling Banff, Canada, Springer, New York, Inc., pp 357–370

7. Huang S-C, Chou I-F, Bias RG (2006) Empirical evaluation of a popular cellular phone's menu system: theory meets practice. J Usability Stud 1:1–108
8. Bostrom F, Nurmi P, Floreen P, Liu T, Oikarinen T-K, Vetek A, Boda P (2008) Capricorn—an intelligent user interface for mobile widgets. In: Proceedings of the 10th international conference on human computer interaction with mobile devices and services Amsterdam, The Netherlands, ACM, pp 327–330
9. Bridle R, McCreath E (2006) Inducing shortcuts on a mobile phone interface. In: Proceedings of the 11th international conference on intelligent user interfaces Sydney, Australia, ACM, pp 327–329
10. Fukazawa Y, Hara M, Onogi M, Ueno H (2009) Automatic mobile menu customization based on user operation history. In: Proceedings of the 11th international conference on human-computer interaction with mobile devices and services Bonn, Germany, ACM, pp 1–4
11. Butler J, Morrice DJ, Mullarkey PW (2001) A multiple attribute utility theory approach to ranking and selection. J Manag Sci 47:800–816