

RESEARCH

Open Access



# RecencyMiner: mining recency-based personalized behavior from contextual smartphone data

Iqbal H. Sarker<sup>1,2\*</sup>, Alan Colman<sup>1</sup> and Jun Han<sup>1</sup>

\*Correspondence:  
msarker@swin.edu.au

<sup>1</sup> Swinburne University  
of Technology, Melbourne,  
VIC 3122, Australia  
Full list of author information  
is available at the end of the  
article

## Abstract

Due to the advanced features in recent smartphones and context-awareness in mobile technologies, users' diverse behavioral activities with their phones and associated contexts are recorded through the device logs. Behavioral patterns of smartphone users may vary greatly between individuals in different contexts—for example, temporal, spatial, or social contexts. However, an individual's phone usage behavior may not be *static* in the real-world *changing* over time. The volatility of usage behavior will also vary from user-to-user. Thus, an individual's *recent behavioral patterns* and corresponding *machine learning rules* are more likely to be interesting and significant than older ones for modeling and predicting their phone usage behavior. Based on this concept of recency, in this paper, we present an approach for *mining recency-based personalized behavior*, and name it "RecencyMiner" for short, utilizing individual's contextual smartphone data, in order to build a context-aware personalized behavior prediction model. The effectiveness of RecencyMiner is examined by considering individual smartphone user's real-life contextual datasets. The experimental results show that our proposed recency-based approach better predicts individual's phone usage behavior than existing baseline models, by minimizing the error rate in various context-aware test cases.

**Keywords:** Mobile data mining, Machine learning, Association rule learning, Data science, User behavior modeling, Context-awareness, Recency, Personalization, Predictive modeling, IoT analytics, Smartphone, Intelligent service

## Introduction

Nowadays, smartphones are considered as essential devices in our daily life. Due to the recent advanced features in smartphones and the popularity of context-awareness in mobile technologies, individual's behavioral activities with their phones, such as phone call activities, mobile applications usage, mobile notification responses, social networking, and corresponding contextual information are recorded through the device logs. An individual smartphone's ability to store user's such diverse activities and associated contexts with their phones enables the study on data-driven smartphone usage behavior modeling and prediction [1]. In this paper, we aim to mine a set of personalized recent behavioral patterns, i.e., recency, based rules utilizing individual's contextual phone log data, for the purpose of building an effective context-aware personalized usage behavior prediction model. To illustrate the efficacy of our proposed approach, in this paper, we

analyze the device's phone call logs that record individual phone call behaviors in different contexts. These phone logs contain individual's diverse phone call activities, e.g., accept, reject, missed, or making an outgoing call, and corresponding contextual information, such as when the call was made (temporal context), where the user was (spatial context), and who the call was from (social relationship context).

To analyze individual's diverse behavioral patterns in such multi-dimensional contexts, and design an effective behavior prediction model utilizing phone log data, can be used for building various data-driven context-aware personalized systems, such as smart interruption management system, intelligent mobile reminder system, smart mobile searching, and context-aware recommendation system etc. that intelligently assist the end mobile phone users in their daily activities in a context-aware computing environment. In order to provide such personalized services, *extracting* a set of *recency based behavioral rules* of individual users based on relevant contextual information utilizing their phone log data, is the key. As individuals' mobile phone usage behavior may change over time from user-to-user, the *recent behavioral patterns*, i.e., recency, and corresponding *machine learning rules* of individuals are more likely to be interesting and significant than older ones for modeling and predicting their behavior, in which we are interested. A *recency based behavioral rule* of an individual mobile phone user based on multi-dimensional contexts is defined as  $[A \Rightarrow C]$ , where  $A$  (antecedent) represents relevant contextual information such as temporal, spatial or social contexts, and  $C$  (consequent) represents individual's recent behavior (phone call activity) for that contexts.

To extract such behavioral rules based on *recency* is challenging as mobile phone log data is not static; it is progressively added to day-by-day according to individual's current behavior with their phones [2]. Currently, researchers use a *static period* (e.g., 6 months) of phone log data in order to build a rule-based user behavior model [3–5]. However, the problem utilizing a static period of log data to model individual's behavior is that behavioral rules may not reflect the recent behavior of a user. Let's consider an example of a mobile phone user Alice. Assume that as per log data the user has a call 'reject' behavioral pattern on Monday [10:00 A.M.–12:00 P.M.] as she used to have a regular meeting at that time. Recently, she has no meeting at that time period on Monday and she typically 'accepts' incoming phone calls. So for this example, the past 'reject' behavioral pattern, even with high evidence (support value) according to log data, is not meaningful to predict her future behavior. Therefore, we need to dynamically determine the *behavior changes* of individual users' so that more currently relevant rules based on recency can be formulated to build an effective model.

In order to achieve our goal, a data-driven recency analysis of individual's log data rather than assumptions is important. For instance, if we assume only a short period (e.g., last week's data) as indicative of recent behavior of an individual, sufficient data instances may not be found to infer a set of meaningful behavioral rules. Individual's behavioral rules based on observations with so little "support" is unlikely to be effective [6]. On the other hand, if we take into account comparatively longer period (e.g., last 6 months data) as indicative of recent behavior, we could get greater *support* but it might result a greater *behavioral variations*. Such variations in behavior for a particular context decrease the confidence value and we may loose these rules because of not satisfying the confidence preference. Thus, the main challenge in this work, is to identify an *optimal period* of recent log data that reflects the recent behavior of individuals' and to *extract*

corresponding recency based rules according to their own behavioral patterns. For extracting rules both association rule learning and classification rule learning are common and popular in the area of machine learning and data mining [7]. In our work, we take into account behavioral *association rule learning* [8] rather than classification rules for the purpose of rule-based modeling. The reason is that rule-set produced by classification techniques, such as rule-based machine learning classifier, e.g., decision tree [9], does not consider user preference that may vary from user-to-user, leading to rigid decision making. As a result, in many context-aware cases, it does not reflect the expected behavioral patterns according to individual users' preferences and may decrease the overall prediction accuracy. According to [10, 11], classification rules mostly have low reliability and cannot ensure that an extracted classification rule will have a high predictive accuracy, which is briefly discussed in "[Background and related work](#)" section.

To address the above mentioned issues in behavior modeling, in this paper, we present RecencyMiner, a *recency-based* approach to model individual's mobile usage behavior, which significantly extends our earlier work [2]. In our approach, we first dynamically determine an *optimal period* by identifying the behavior changes of individuals, for which a recent behavioral pattern has been dominant by analyzing the behavioral characteristics of individual mobile phone users utilizing their phone log data. If the behavioral changing point may not found for a particular user, then we assume that her behavior is consistent over time and the entire dataset of her smartphone log can be used to discover her recent behavioral patterns. In our approach, once we have determined the recent log data, we then identify and remove the outdated rules that do not represent the present behavior of an individual. Thus, our recency-based approach outputs a complete set of updated behavioral rules of individuals utilizing their phone log data. The rule-set produced by our recency-based approach can be used to minimize the error rate in various context-aware test cases while predicting their behavior. As individuals' behavior can vary widely in the real world, such optimal period of recent log data and corresponding discovered recency-based rules may differ from user-to-user, depending on their unique behavioral patterns.

The contributions of this work can be summarized as follows.

- We propose an approach to dynamically identify an *optimal period* of recent log data based on changes in patterns of an individual's behavior.
- We mine a set of *recency-based user behavioral rules* of individuals from their contextual smartphone data, in order to model and predict their smartphone usage behavior.
- We have conducted experiments on individual's real life smartphone datasets to evaluate our recency based approach and compare with existing base models to show the effectiveness of RecencyMiner in predictions.

The rest of this paper is organized as follows. We review the related works in "[Background and related work](#)" section. "[Requirements analysis](#)" section summarizes the key requirements of our recency-based approach. In "[RecencyMiner: our approach](#)" section, we present our recency-based approach step-by-step in order to extract the recency based behavioral rules for individual mobile phone users. We report the experimental results in "[Evaluation and experimental results](#)" section. Some key observations of our

recency-based approach are summarized in “[Discussion](#)” section, and finally “[Conclusions](#)” section concludes this paper and highlights the future work.

## Background and related work

In the area of data mining and machine learning, both association rule learning and classification rule learning, are the most common techniques for discovering rules from a given dataset [7, 12]. Decision tree [9] is the most popular classification algorithm for generating classification rules. However, rules produced by the decision tree, based on contexts mostly have low reliability [10]. According to [11], decision trees cannot ensure that a discovered classification rule will have a high predictive accuracy. Moreover, this technique provides no flexibility to set user preferences (e.g., confidence level) that may vary from user-to-user according to the consistency in behaviors, leading to rigid decision making [9]. On the other hand, association rule learning [13] is the discovery of associations or relationships among a set of available items in a given dataset. It discovers association rules that satisfy the pre-defined minimum support and confidence constraints preferred by an individual, which ensures the reliability of rules [14]. Apriori proposed by Agrawal et al. [14] is the most popular algorithm for mining association rules. In addition to these techniques, a number of techniques have been proposed for mining rules in a dynamic database. These are pattern based [15, 16], tree based [17], three-way decision based [18, 19], probability-based [20, 21]. These techniques take into account the faster processing, e.g., efficiency of mining process by reducing the scan of dataset instead of processing the merged dataset that includes the original dataset and the incremental part of the dataset. However, these techniques do not take into account the *freshness of rules*, i.e., rules that represent recent patterns, in which we are interested to output a complete set of updated behavioral rules based on recency for individual mobile phone users utilizing their contextual smartphone datasets.

In order to mine users’ contextual mobile phone data to model their behavior, a number of authors use a static period of phone log data, such as phone call logs [22–25], SMS Log [26], mobile application (apps) usages logs [3, 27, 28], mobile phone notification logs [10], web logs [29–31], game Log [32], context logs [4], and smartphone life log [33, 34] etc. for various purposes. In particular, Pielot et al. [25], use a static period of log data starting from 2012 to May 2014 to predict whether a user would pick up a call or not. In [23], the authors use phone call log data starting from August 2014 to September 2015 as a context source to model individual mobile phone user behavior. In [35], Sarker et al. have proposed a machine learning based robust user behavior model by doing experiments on individual’s real-life mobile phone data over the period of 9 months. Mafrur et al. [34] use smartphone sensing life-log data of 2 months time period for modeling and discovering human behavior for identification purpose. Zhu et al. [4] use a static period of contextual data of several months for mining mobile user preferences for personalized context-aware recommendation. In [3], Srinivasan et al. also use a static period of phone log data of 3 months for mining the contextual behavioral rules of individual mobile phone users, for predicting which app is preferred by a particular user under a certain context. To extract contextual behavioral rules according to individual’s preferences, Mehrotra et al. [10] use a static period of mobile notification log data consisting of 11,185 notifications for the purpose of building intelligent mobile notification management systems. All these approaches use entire log data for a static period of time,

and consider the overall behavioral patterns in the given datasets, to model users' phone usage behavior. However, they do not privilege recent behavioral patterns, in which we are interested in, to model individual user behavior utilizing their phone log data.

In order to produce rules according to the recent behavior of an individual, a number of researchers use the behavioral patterns of recent mobile phone log data to predict the future behavior than the patterns derived from the entire historical logs. However, they consider a static period to define "recent" behavior. For instance, Lee et al. [5] extract call logs data for previous 3 months as a recent period for designing a call recommendation algorithm for an adaptive speed-call list. In [22], the authors assume the latest 2 months as a recent period of time call records for predicting incoming and outgoing calls for the next 24-h based on the user's past communication history in order to get better prediction accuracy. Phithakkitnukoon et al. [36] discuss about the adequate amount of historical data than considering the entire historical for constructing a predictive model for caller behavior. Besides these approaches, a number of authors [37–39] deal with the problem of managing personal information in their mobile phones based on their different usage patterns for a static period of log data. Although, the most recent pattern is more significant than older ones, these approaches use an arbitrary period of recent log data from the entire data set. The problem utilizing such arbitrary period of log data to produce rules is that those rules may not reflect the present behavior of a user, as an individual's behavior changes over time in their real world life.

Unlike these works, in this paper, we present a *recency-based* approach that dynamically determines an *optimal period* of recent log data for individuals according to their recent behavioral patterns. Using this recent log data, this approach not only identifies and removes the *outdated rules* but also outputs a complete set of *recency-based updated rules* for individuals according to their recent behavioral patterns utilizing their own phone log data.

## Requirements analysis

In this section, we discuss and summarize the key requirements of our recency-based approach. These are:

- Req1** *Identifying changes in individual's behavioral patterns and determining recent log*  
 As we aim to extract individual's recency-based behavioral rules, a key requirement is to identify changes in individual's behavioral patterns, and determining corresponding dynamic recent log data. An *optimal period* of recent log data that reflects the recent behavior of an individual can be determined by analyzing their behavioral patterns in relevant contexts. The concept of *recent log data* is formally stated as—*Let,  $s_1$  be the number of instances (records) in the entire mobile phone dataset  $DS$ , which is temporally ordered. A recent mobile phone dataset  $DS_{recent}$  is a subset of  $DS$ , which contains the most recent records of  $DS$  based on timestamps of size  $s_2$ , where  $s_2 \leq s_1$ .* This dynamic optimal period of data can be used to discover the recency-based rules of individuals. Therefore, the approach should have the ability to *identify the changes* in individual's behavior from entire phone log data without making any predefined assumptions.

- Req2** *Detecting and removing outdated rules* An outdated (out-of-date) behavioral rule is a valid rule in terms of rule's constraints (e.g., support and confidence) but does not represent the recent behavior of an individual user. The definition of an outdated rule of individual mobile phone users is formally stated as—*Let, a rule  $R_1 : A_1 \Rightarrow C_1$  that is discovered from entire mobile phone dataset  $DS$ , where  $A_1$  represents the contextual information and  $C_1$  is the mobile phone usage behavior. The rule  $R_1$  is considered as an outdated rule  $R_{outdated}$ , if and only if  $C_1$  is identified as conflict (different behavior) for that context  $A_1$  utilizing recent phone log data  $DS_{recent}$ , i.e.,  $A_1 \Rightarrow C_2$  and  $C_1 \neq C_2$ , where  $C_1$  and  $C_2$  represent the past and recent behavior respectively for  $A_1$ . In general, this type of rules are produced based on past behavioral patterns of individual's utilizing the entire phone log data. As the most recent pattern is more significant than older ones, the outdated rules even with high support value increases the error-rate for predicting individual's future behavior. Therefore, the approach should have the ability to detect and remove the outdated rules from the rule-set extracted from entire phone log data.*
- Req3** *Discovering new recent behavioral rules* A new recent behavioral rule is a rule that is not produced when utilizing the entire phone log data  $DS$  but is produced when utilizing the recent period of log data  $DS_{recent}$ . The definition of a new recent behavioral rule of individual mobile phone users is formally stated as - *Let, a rule  $R : A \Rightarrow C$  that is produced utilizing recent log data  $DS_{recent}$ , where  $A$  represents the contextual information and  $C$  is the mobile phone usage behavior. The rule  $R$  is considered as a new recent behavioral rule  $R_{new}$ , if and only if, there is no such rule discovered from the entire log dataset  $DS$ . Although  $DS_{recent}$  is a subset of  $DS$ , such kind of rules are not discovered utilizing the entire log data  $DS$  because of their low confidence value and not satisfying the user preferred confidence threshold (say, 80%). The reason is that individual's behavior changes over time for a particular context and a number of variations in user's behavior or conflicts for that context decrease the confidence of the associated behavior. However, a strong behavioral pattern with high confidence may be found in the recent phone log  $DS_{recent}$ , which satisfies the user preferred confidence threshold. Such new rules make the behavior model more significant in order to predict individual's future behavior. Therefore, the approach should have the ability to produce such new recent behavioral rules of individuals.*
- Req4** *Dynamic management of rules* As the recency-based approach is responsible not only to identify the dynamic optimal period of recent log data but also identifying and removing the outdated rules, and discovering new recent behavioral rules, a dynamic management of rules is needed to get a complete set of updated rules without making any assumptions about when individual's behavior changed to a new pattern. *Let,  $RS_{initial}$  be a set of rules discovered from entire mobile phone data  $DS$ , and  $RS_{recent}$  be another set of rules discovered from recent log data  $DS_{recent}$ . A complete set of recency based updated rules  $RS_{updated}$  will be the merging output of these two rule-sets, e.g.,  $RS_{updated} = merge(RS_{initial}, RS_{recent})$ . This complete updated rule-set  $RS_{updated}$  not only contains all the significant rules of an individual mobile phone user, but also expresses recent behavioral patterns that will be applicable for modeling mobile phone usage behavior in the real world applications.*

### RecencyMiner: our approach

In this section, we discuss our recency-based approach step-by-step for modeling individual mobile phone users' behavior utilizing their phone log data.

#### Approach overview

Our approach accepts as input a real mobile phone log dataset  $DS$ . From this log data, our approach is able to output a complete set of recency based updated rules for individuals by going through several processing steps. First, we identify changes in individual's behavioral patterns to dynamically determine an *optimal period* of recent log data from the entire phone log. The optimal data period is determined by *measuring behavioral similarity* of an individual for relevant contexts between the adjacent weeks started from the most recent week to the previous weeks. Second, from this recent log data, we produce a set of recent behavioral rules  $RS_{recent}$ . We also produce a set of rules using the entire log dataset, which is known as initial rule set  $RS_{initial}$  for the purpose of rule comparison. Third, once we have produced behavioral rules from the determined recent log data, we identify and remove the *outdated rules* that do not represent the present behavior of an individual, from the initial rule set. We also remove rules from  $RS_{recent}$  that exist in the initial rule set  $RS_{initial}$ . Finally, we merge these two rule-sets in order to output a complete set of recency-based updated rules  $RS_{updated}$  for each individual user. This complete updated rule set not only contains all the significant rules of an individual mobile phone user from the initial week to the most recent week but also expresses their recent behavioral patterns.

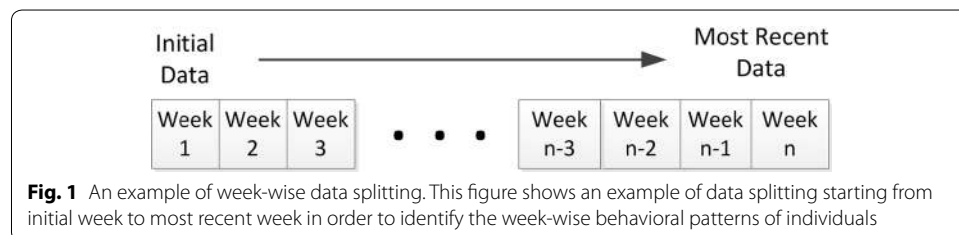
#### Identifying optimal period of recent log data

##### Data splitting

In this first step, we split the entire log into week-wise data as the time-of-the-week is the most important aspect impacting on user behavior [30]. We choose weekly basis splitting because of an individuals' behavior is unlikely to be identical for all days in a week (Monday, Tuesday,..., Sunday). Thus we assume that weekly patterns of behavior will repeat (e.g., a user has the same days off work each week). Figure 1 shows an example of week-wise data splitting, where week  $W_1$  represents the initial week data and  $W_n$  represents the most recent week data in the mobile phone log of an individual mobile phone user.

##### Association generation

Once the data splitting has been completed, we generate context-association for each set of week-wise data  $DS_{week}$  starting from the most recent week  $W_n$ . Context association is simply the combination of contexts, where





- i. The association may contain single (user social activity, e.g., meeting) or multi-dimensional contexts (user social activity, e.g., meeting, user location, e.g., office).
- ii. Contexts are added incrementally according to the precedence of contexts to create an association based on multi-dimensional contexts.
- iii. Each context may occur at most once in an association.
- iv. The number of contexts in an association is less or equal to the total number of contexts in a given dataset  $DS_{week}$ .

In order to identify the precedence of contexts in a dataset, we calculate information gain [9] which is a statistical property that calculates entropy and measures how well a given context-value separates the training datasets into targeted behavior classes available in the dataset. The context with the highest information gain value is considered as the highest precedence context.

The process for generating context associations is set out in Algorithm 1. Input data includes week wise data:  $DS_{week} = X_1, X_2, \dots, X_n$ , which contains a set of instances with categorical contexts and output data is the association list *assoc\_list*. We first initialize *assoc* as empty. After that, we calculate the entropy and information gain for each context and identify the precedence of contexts. Once we have determined the highest precedence context, for each context value we generate a subset  $DS_{sub}$  that contains that context value. If the subset  $DS_{sub}$  is not empty, we recursively do this for all contexts and generate the associations by taking into account all contexts according to their precedence. When the context list becomes empty, the algorithm returns the generated association list *assoc\_list*. {office, meeting} is an example of context association containing 2-contexts.

---

**Algorithm 1:** Context Association Generation

---

**Data:** Week-wise dataset:  $DS_{week} = X_1, X_2, \dots, X_n$  // each instance  $X_i$  contains a number of categorical contexts

**Result:** association list *assoc\_list*

```

1 initial_assoc  $\leftarrow \phi$ ; // initialization
2 Procedure CAG ( $DS_{week}, context\_list$ );
3 if context_list is empty then
4   | return assoc_list;
5 end
6 //calculate entropy and information gain for each context and identify the precedence of
  contexts
7  $A_{best} \leftarrow$  the context that best classifies examples;
8 foreach context value con  $\in A_{best}$  do
9   | assoc  $\leftarrow$  create_association(con, initial_assoc);
10  | added assoc to the assoc_list;
11  | // partition into subsets and grow the corresponding subtrees for each subset
12  |  $DS_{sub} \leftarrow$  subset of  $DS_{week}$  that have value con;
13  | if  $DS_{sub}$  is empty then
14  |   | initial_assoc  $\leftarrow \phi$ ;
15  | else
16  |   | //remove  $A_{best}$  context from the context list
17  |   | context_list  $\leftarrow$  context_list -  $A_{best}$ ;
18  |   | // recursively generated association with remaining attributes
19  |   | CAG( $DS_{sub}, context\_list$ )
20  | end
21 end

```

---

**Score calculation**

Once we have generated the context associations, we then calculate the conflict score based on the conflict behavior for each association between two adjacent weeks. For this, we first identify the *dominant behavior* (maximum number of occurrences) [40], as we do not



expect always 100% like behavior of a user for a particular association. For instance, say a user 85% rejects, 10% accepts and 5% misses the incoming calls for a particular association of context (e.g., meeting, office), then ‘reject’ will be the dominant behavior for that association. Another example, say a user 65% accepts and 35% rejects the incoming calls for a particular association of context (e.g., seminar, office, colleague), then ‘accept’ will be the dominant behavior for that association. We start scanning from the most recent week  $W_n$  and continue to all previous weeks  $W_{n-1}, W_{n-2}, W_{n-3}, \dots, W_1$  one by one to identify the conflict behavior for each context association in the adjacent weeks.

Once we have determined whether there is a conflict or not for each context association generated in the earlier section, we calculate the conflict score according to Eq. 1. If  $assoc_{total}$  represents the total number of associations generated in week  $W_n$  and  $conflict_{total}$  is the total number of conflicts found comparing with the generated associations in week  $W_n$  and the adjacent week  $W_{n-1}$ , then the percentage (%) of conflict score with respect to the most recent week  $W_n$  is defined as below:

$$Score (\%) = \frac{conflict_{total}}{assoc_{total}} \times 100 \quad (1)$$

The process for calculating this conflict score is set out in Algorithm 2. Input data includes adjacent weeks data:  $DS_{week1}$  for week  $W_n$  and  $DS_{week2}$  for week  $W_{n-1}$ , each of which contains a set of training instances  $X_1, X_2, \dots, X_n$  and output data is the conflict score in percentage. We first generate context associations for  $DS_{week1}$  and  $DS_{week2}$  using Algorithm 1. After that for each association, we check whether the dominant behavior is same or not. If different dominant found then the number of conflict increases. After that, we calculate the percentage (%) of conflict behaviors. Finally, this algorithm returns the calculated score.

---

#### Algorithm 2: Conflict Score Calculation

---

**Data:** adjacent weeks data:  $DS_{week1}$  for week  $W_n$  and  $DS_{week2}$  for week  $W_{n-1}$ .

**Result:** conflict score: *score*

---

```

1  assoc_list1 ← generate_association(DSweek1);
2  (using algorithm 1)
3  assoc_list2 ← generate_association(DSweek2);
4  (using algorithm 1)
5  //initializing score count
6  initialize count ← 1
7  foreach association assoc in assoc_list1 do
8      if same assoc found in assoc_list2 then
9          //identify the dominant behavior for assoc
10         BH1 ← identify_dominant(assoc, DSweek1);
11         BH2 ← identify_dominant(assoc, DSweek2);
12         //check the dominant behavior
13         if BH1 ≠ BH2 then
14             increment count
15         end
16     end
17 end
18 //calculate the percentage (%) of conflicts score
19 score ← calculatePercentage(count, assoc_list1)
20 return score

```

---

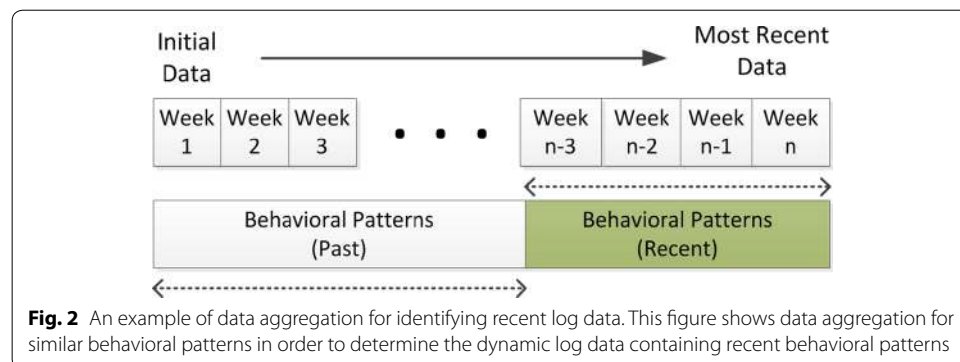
### Data aggregation

Data aggregation is the last step for determining *an optimal period* of recent log data. For this, we aggregate the week-wise data based on similar behavioral patterns identified by conflict score. For identifying behavioral similarity, we use the conflict score (discussed above) between 2 adjacent weeks rather than likelihood (the fact of somethings being likely), as we do not expect similar contextual information in each week. For example, say, in a particular week, the user attends in a seminar, but may not attend in seminar in all weeks. However, the conflict score identifies the behavioral variations between 2 adjacent weeks. If the conflict score of 2 adjacent weeks is 0% (no conflict), the behavioral patterns are highly similar in these 2 weeks [41]. We aggregate from the most recent week  $W_n$  to the previous weeks  $[W_{n-1}, W_{n-2}, \dots]$  so on until getting a significant variation in the conflict scores of 2 adjacent weeks. We then set a boundary line for recent similar behavioral patterns. A significant variation is encountered when it exceeds the average result of the variations by considering the overall behavior in the entire dataset. If  $S_{total}$  represents the total conflict score and  $N_{weeks}$  is the number of total weeks in a dataset, then the average score is defined as:

$$\text{Average score} = \frac{S_{total}}{N_{weeks}} \quad (2)$$

This helps to identify the dynamic threshold rather than assuming a static threshold to determine an optimal period of recent log data. Such threshold may differ from user-to-user according to their behavioral consistency. Thus, for some users, recent behavioral patterns are found by aggregating large number of weeks and for some users a smaller number of weeks depending on how the user's behavior changes over time-of-the-week in different contexts.

Figure 2 shows an example of recent log data by aggregating the most recent four weeks data (from Week  $W_{n-3}$  up to Week  $W_n$ ), which reflect the recent behavioral patterns of an individual user. According to Fig. 2, week  $W_n$  is the most recent week and week  $W_{n-3}$  is the boundary of recent behavioral patterns, that is, the behavioral patterns based on related contexts before week  $W_{n-3}$  (from week  $W_1$  up to week  $W_{n-3}$ ), are considered as past behavior and the behavioral patterns after week  $W_{n-3}$  up to the most recent week  $W_n$  (from week  $W_{n-3}$  up to week  $W_n$ ), are considered as recent behavior of that user. If there is no change in behavioral patterns from week  $W_1$  (beginning of log data) up to week  $W_n$ , then the behavioral patterns in the entire log data are considered as recent patterns.



**Fig. 2** An example of data aggregation for identifying recent log data. This figure shows data aggregation for similar behavioral patterns in order to determine the dynamic data containing recent behavioral patterns

Rather than arbitrarily determine the number of period in advance, our algorithm dynamically derives an optimal period of recent log data from an individual's mobile phone data. Thus, the number of weeks and time boundaries for recent log data will differ from user-to-user depending on how the user's behavior changes over time-of-the-week in different contexts. We utilize such variable length of recent log data for producing individual's recency-based rules.

### Machine learning based behavioral rule generation and management

Once the recent log data  $DS_{recent}$  has been determined, we produce rules utilizing this data. To produce rules, we apply our earlier rule-based machine learning technique, *association generation tree* [8] on recent log data. The reason for choosing this tree-based learning is that in a tree-based approach, the nodes closer to the root are more general, that can be used in mining *general* behavioral rules. In order to generate the behavioral rules, this approach first generates a tree according to the precedence of contexts, where each node represents the behavior class and corresponding confidence value. After designing the tree, rules are extracted by traversing the tree from root node to each decision node, identified by node's value. This approach produces a set of human understandable behavioral rules ( $contexts \Rightarrow behavior$ ) based on multi-dimensional contexts in order to model individual mobile phone user behavior. The produced rules not only capture individual's generalized behavior at a particular level of confidence with a minimal number of contexts, but also express specific exceptions to the general rules when more context-dimensions are taken into account. For instance, typically a user rejects most of the incoming calls (83%), when she is in a meeting; However, she always (100%) accepts if the incoming call is from her boss. Thus the produced general and specific exception rule are represented as  $R_{general} : meeting \Rightarrow reject$  (conf = 83%) and  $R_{exception} : meeting, boss \Rightarrow accept$  (conf = 100%) respectively. Such produced rules are *non-redundant* and *reliable* according to individual's preferred confidence.

In our approach, once we have produced rules utilizing a dynamic length of recent log data  $DS_{recent}$ , we merge this rule-set with initial rule-set  $RS_{initial}$  that is produced utilizing the entire phone log data  $DS$ . To extract the initial rule-set  $RS_{initial}$ , we also use the same rule discovery approach [8] discussed above, in order to output a *complete set of updated rules*  $RS_{updated}$  for each individual user. While merging, we identify and remove the *outdated rules* from the initial rule-set  $RS_{initial}$ , as these rules do not represent the recent behavior of an individual. We also remove rules from  $RS_{recent}$  that exist in the initial rule set  $RS_{initial}$ . Thus, we output a complete set of recency-based updated rules by taking into account the behavioral patterns in both the rule-sets  $RS_{initial}$  and  $RS_{recent}$  using a rule merging operation, e.g.,  $RS_{updated} = merge(RS_{initial}, RS_{recent})$ .

### Evaluation and experimental results

To validate our proposed recency-based approach, we have conducted a range of experiments on the real mobile phone datasets of individual mobile phone users. For this purpose, first we set a number of questions that we aim to answer by the experiments and describe the experimental setup. Then, we discuss our findings in answering these questions.

### Experimental setup

To validate our proposed recency-based approach, we aim to answer the following questions:

- Question 1: Does the produced recency-based behavioral rule-set of an individual mobile phone user differ with the initial rule-set discovered from the entire phone log data?
- Question 2: Is our recency-based approach personalized and how is the conflict score used to identify an optimal period of recent log for individuals?
- Question 3: How effective is our proposed recency-based approach, RecencyMiner, in minimizing the error rate in context-aware predictions relative to existing base models?

In answering these questions, we have conducted a range of experiments on the real mobile phone datasets of individual mobile phone users. In the following subsections, we briefly describe the datasets, and present the experimental results and discussion.

### Smartphone datasets

We have conducted experiments on ten phone log datasets of individuals to evaluate our approach. These call log datasets consist of 55,105 phone call records, and represented as *CDS01*, *CDS02*, ..., *CDS10* for ten individual mobile phone users respectively for the purpose of experimental evaluation. These datasets are collected from individual mobile phone users over the period of 9 months by Massachusetts Institute of Technology (MIT) for their Reality Mining project [26]. An example of such data is represented as ‘device ID (e.g., 000e6d2a3564— Amy’s Phone), time series (e.g., 2016-09-19 10:03:15—time format YYYY-MM-DD hh:mm:ss), cell area (e.g., 24127), cell tower ID (e.g., 111 - MIT), contact phone number (e.g., 6175559821—Amy’s number), call directions (incoming, missed, and outgoing), call duration (e.g., 23 s). These raw datasets are used to conduct our experimental analysis for the purpose of validating our recency-based approach “RecencyMiner”.

### Preparing contextual raw data

In our experiments, initially we pre-process all the raw contextual data of the given datasets described above. In this process, we first convert the time-series data into nominal values as the raw temporal data represents continuous time-series with numeric timestamps values. In order to generate nominal values of raw time-series data, we use our earlier BOTS technique [40] that dynamically generates a number of behavior-oriented time segments, according to their behavioral activity patterns. An example of such segment is Friday [09:00–11:00] that represents similar behavioral activities in that time period. As social context, we use individual’s unique contact number available in the datasets in our experiment. For this, we also generate data-centric social context [42] that represents individual’s one-to-one social relationship based on their unique mobile phone numbers in the dataset. For example, mother’s phone number (03..0543) is used as one relationship ‘Rel01’, while friend’s phone number (03...0342) is used as another

relationship ‘Rel02’. For spatial context, we use individual’s physical location on the earth, such as home, office, market, MIT, Harvard etc. recorded from the given cell tower information that exist in the dataset. We also pre-process individual’s phone call behavior as we are interested in user’s accepting and rejecting behaviors that are recorded as incoming call activity in the dataset. Thus, we derive these behaviors using call duration that represents one’s talking period with another over phone. If the call duration for an incoming call is zero then the call has been rejected (not answered), otherwise (call duration > zero) the call has been accepted [6]. Overall in our experiment analysis, we use the above mentioned three dimensions of contexts, and corresponding diverse phone call behaviors, accept, reject, missed and making outgoing calls in these contexts, to evaluate our recency-based approach.

### Evaluation metric

In order to measure the effectiveness of the discovered recency-based rules, we compare the predicted response with the actual response (i.e., the ground truth) and compute the effectiveness in terms of:

- *Error rate* a performance measure often express as a percentage in predictions. It measures the percentage of incorrect predictions over the total number of test cases, which is determined by the best matching rules that are discovered. Let, the number of incorrect predictions is  $n_{incorrect}$ , and the total number of test cases is  $|N|$ , then the formal definition of error rate is:

$$Error\ rate\ (\%) = \frac{n_{incorrect}}{|N|} * 100\% \quad (3)$$

- *Prediction coverage* measures how many of the test cases are predicted by the discovered rules for a particular confidence threshold, preferred by an individual mobile phone user. Let, the number of test cases predicted by the rules is  $n_{covers}$ , and the total number of test cases is  $|N|$ , then the formal definition of the coverage is:

$$Coverage\ (\%) = \frac{n_{covers}}{|N|} * 100\% \quad (4)$$

In order to calculate the effectiveness of our recency-based approach in terms of the above defined error rate (%) and prediction coverage (%), we take into account the most recent two weeks data as test cases, and the remaining as train dataset, in order to build the model. For instance, if the data of the weeks  $W_n$  and  $W_{n-1}$  is considered as test data, then the data of all the previous weeks  $\{W_{n-2}, W_{n-3}, \dots, W_1\}$  are considered as train data, where  $W_n$  represents the most recent week data. The *higher value of prediction coverage* with *lower error-rate* represents the effectiveness of our recency-based approach.

### Experimental results

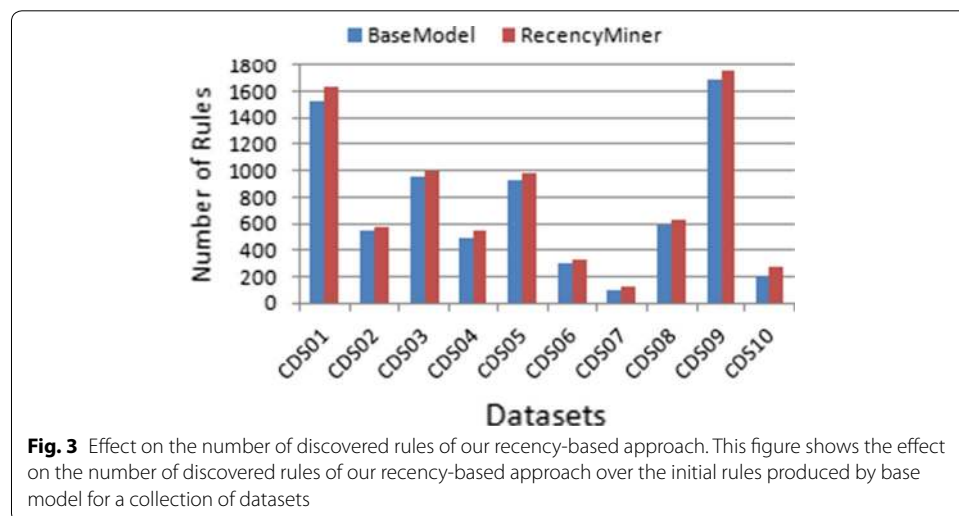
As our recency based approach is individualized, we illustrate with the detailed of experimental results utilizing all the datasets, mentioned above. In addition to the

individualized results, we also report the average prediction results of our experiments on all the above mobile phone datasets of individual users. The experimental results of our approach are also compared with the existing baseline approaches that use a static period of log data in order to model the phone usage behavior of individuals. We call these models as “BaseModel” in terms of using the static period of log data rather than recency for modeling phone usage behavior. The details of these base models are discussed briefly in “[Background and related work](#)” section. As we determine a dynamic period of recent log data by taking into account individual’s recent behavioral patterns, in our recency based approach “RecencyMiner”, we compare the experimental results with the base model mentioned earlier, in the below subsections.

#### **Effect on the discovered rules**

In order to answer the first question, in this experiment, we show the effect on the number of rules discovered by our recency-based approach. For this, Fig. 3 shows the relative comparison of the produced number of rules for all ten datasets CDS01, CDS02, ..., CDS10 for a particular confidence preference 80%. For the purpose of comparison, we apply both the base model that considers the entire static log data and our recency-based approach that takes into account the dynamic recent log data for an optimal time period, for all the datasets mentioned above.

If we observe Fig. 3, we see that the produced number of rules using our recency-based approach increases compared with the number of rules produced by the base model for these datasets. The base model considers the overall patterns in the dataset meaning that if there is a change in behavior in the dataset the older conflicting rules will nullify the more recent rules. As a result, a number of recent patterns can not be discovered in the rule-set produced by base model. On the other-hand, in our recency-based approach, we discover a number of new behavioral rules according to the recent behavioral patterns in the datasets and output a complete set of recency based updated rules. Thus the number of rules increases, depends on the number of new discovered rules based on recent patterns.



### Effect of conflict score for identifying individual's recent log

In order to answer the second question, in this experiment, we show the effect of conflict scores on adjacent weeks in order to identify the recent log period. For this, Table 1 shows the conflict scores of a sample user for each adjacent week, where  $W_n$  represents the most recent week in the dataset.

If we observe Table 1, we see that the behavior of an individual mobile phone user is not conflict-free over time. For some adjacent weeks (week  $W_n$  and week  $W_{n-1}$ ), the conflict score is zero, i.e., the behavior is identical for the similar contexts in these weeks. On the other hand, for some adjacent weeks (week  $W_{n-15}$  and week  $W_{n-16}$ ), the conflict score is more than zero, i.e., not identical for all the similar contexts. As can be seen in Table 1, the conflict score is not always zero of an individual (dataset CDS06), we calculate the average score (2.22%) using Eq. 2 of an individual to use as a threshold rather than assuming an arbitrary threshold value.

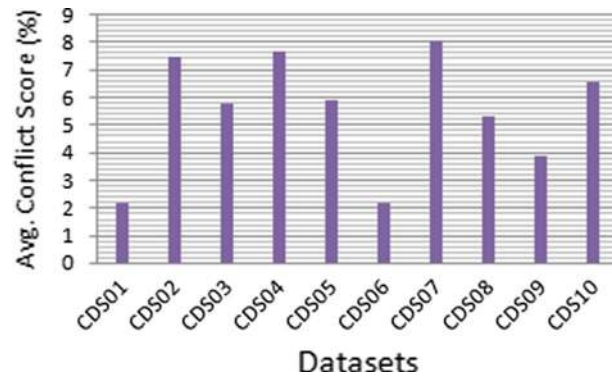
From Table 1, we found that the behavioral patterns are similar from week  $W_n$  to week  $W_{n-5}$  and a significant variation ( $\geq 2.22\%$ ) has been encountered between week  $W_{n-5}$  and week  $W_{n-6}$  for this user. In other words, the last 6 weeks data is the recent log period that represents the recent behavioral patterns of this user.

Figures 4 and 5 summarize the average conflict score (%) and corresponding recent log period (in weeks) for all datasets. From Fig. 4, we found that average conflict score may vary from user-to-user depends on their behavioral consistency over time. As the behaviors of different individuals are not identical in the real word, the dynamic log period may also differ from user-to-user according to their unique behavioral patterns, shown in Fig. 5. Thus, we can conclude that a static period of log data may not be meaningful to model individual's phone usage behavior, should be personalized, as shown in Figs. 4 and 5.

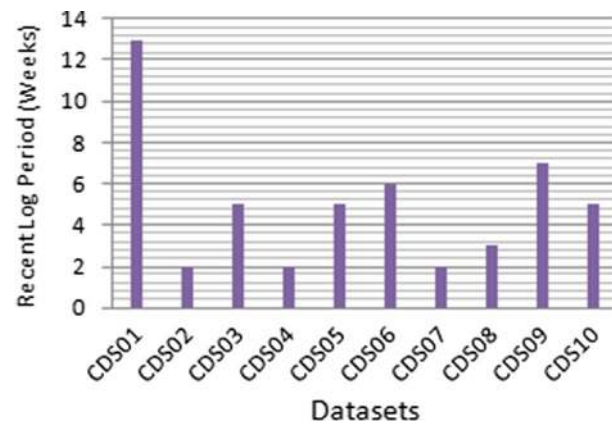
**Table 1 Conflict score count for a sample user utilizing the dataset (CDS06)**

Adjacent weeks	Conflict score (%)
Week [ $W_n$ ] and week [ $W_{n-1}$ ]	0
Week [ $W_{n-1}$ ] and week [ $W_{n-2}$ ]	0
Week [ $W_{n-2}$ ] and week [ $W_{n-3}$ ]	0
Week [ $W_{n-3}$ ] and week [ $W_{n-4}$ ]	0
Week [ $W_{n-4}$ ] and week [ $W_{n-5}$ ]	1.43
Week [ $W_{n-5}$ ] and week [ $W_{n-6}$ ]	2.95
Week [ $W_{n-6}$ ] and week [ $W_{n-7}$ ]	0.83
Week [ $W_{n-7}$ ] and week [ $W_{n-8}$ ]	0.70
Week [ $W_{n-8}$ ] and week [ $W_{n-9}$ ]	3.37
Week [ $W_{n-9}$ ] and week [ $W_{n-10}$ ]	3.35
Week [ $W_{n-10}$ ] and week [ $W_{n-11}$ ]	2.77
Week [ $W_{n-11}$ ] and week [ $W_{n-12}$ ]	1.10
Week [ $W_{n-12}$ ] and week [ $W_{n-13}$ ]	1.67
Week [ $W_{n-13}$ ] and week [ $W_{n-14}$ ]	1.80
Week [ $W_{n-14}$ ] and week [ $W_{n-15}$ ]	2.58
Week [ $W_{n-15}$ ] and week [ $W_{n-16}$ ]	5.01
Week [ $W_{n-16}$ ] and week [ $W_{n-17}$ ]	3.41
Week [ $W_{n-17}$ ] and week [ $W_{n-18}$ ]	4.51
Week [ $W_{n-18}$ ] and week [ $W_{n-19}$ ]	3.46
Week [ $W_{n-19}$ ] and week [ $W_{n-20}$ ]	5.32
Week [ $W_{n-20}$ ] and week [ $W_{n-21}$ ]	0.59





**Fig. 4** Effect on the conflict score of individuals. This figure shows the effect on the conflict score according to individual's behavioral patterns utilizing their datasets, which may vary from user-to-user

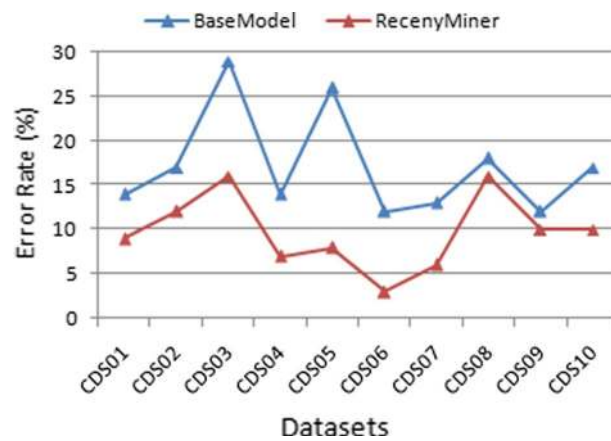


**Fig. 5** Effect on the recent log period of individuals. This figure shows the effect on individual's recent log period in weeks utilizing their datasets, which may vary from user-to-user

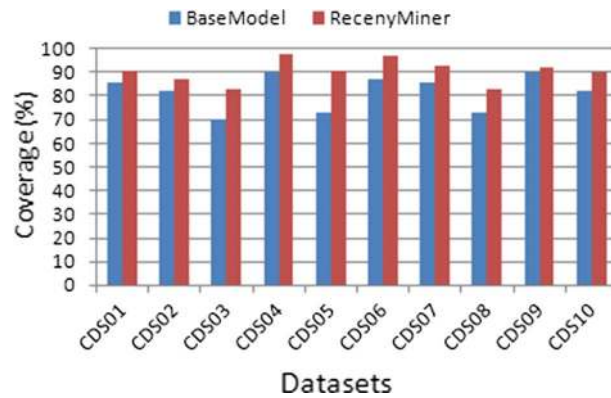
#### **Effectiveness comparison and analysis**

In order to answer the third question, in this experiment, we show the effectiveness of our recency based approach in terms of prediction coverage (%) and error rate (%) comparing with the base model for providing context-aware mobile services. Figures 6 and 7 show the prediction results for different individuals utilizing their own datasets, mentioned above. The results are shown for a particular confidence preference 80% for both approaches.

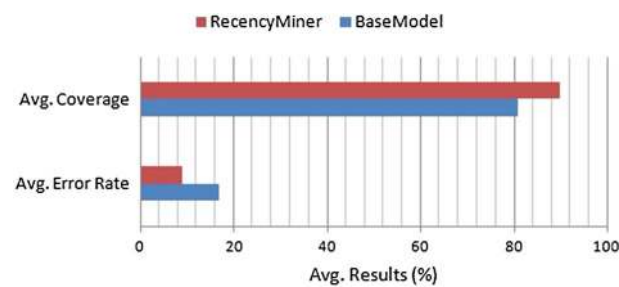
From Figs. 6 and 7, we find that our recency-based approach consistently outperforms the base model for predicting individuals mobile phone usage behavior. The main reason is that rules produced by the base model do not reflect the rule's freshness according to the recent behavior of individuals. As a result, it gives higher error rate in predictions, as we use the recent dataset as the test cases. On the other-hand, our recency-based approach resolves this issue by producing rules according to individual's recent behavioral patterns, thus makes the approach more effective by maximizing prediction coverage with minimum error rate.



**Fig. 6** Effectiveness comparison in terms of error rate for individuals. This figure shows the effectiveness comparison in terms of error rate (%) of our approach with the base model utilizing their datasets



**Fig. 7** Effectiveness comparison in terms of prediction coverage for individuals. This figure shows the effectiveness comparison in terms of prediction coverage (%) of our approach with the base model utilizing their datasets



**Fig. 8** Effectiveness comparison of individuals (average) This figure shows the effectiveness comparison of our approach in terms of average prediction coverage (%) and error rate (%) for a collection of dataset

In addition to individual's comparison, we also show the relative comparison of average prediction coverage and error rate in predictions comparing it with base model for a collection of datasets. The average results (average prediction coverage and the average error rate for all the datasets) are shown in Fig. 8. The average results also show that our

recency-based approach performs better than the base model for a collection of datasets. The reason is that we take into account individual's recent behavioral patterns while producing the behavioral rules for individual users, which improves the effectiveness of our approach by capturing their behavioral patterns more properly.

## Discussion

Overall, our recency-based approach is fully personalized and reflects individual's recent behavioral patterns according to their phone log data. To the best of our knowledge, this is the first *dynamic* recent log-based study that takes into account individual's *recent behavioral patterns* for modeling individual mobile phone user's behavior in order to get a complete set of recency-based updated rules. Compared to the base model that uses a static period of log data (briefly discussed in "[Background and related work](#)" section), the effectiveness of our recency-based approach is improved for predicting individual's phone usage behavior. Our approach not only maximizes the prediction coverage for a number of test cases but also minimizes the error-rate (%) in predictions that have been shown in Figs. 6, 7 and 8. In the following, we highlight a number of key observations of our approach.

To determine an *optimal period of recent log*, identifying changes in individual's behavior is the key of our recency-based method. In our approach, we have dynamically determined a particular period of recent log data for each individual, which gives the optimal result based on the recent behavioral patterns of an individual considering all the relevant context associations of that particular user. Such optimal period may differ from user-to-user as the behavioral patterns of individuals are not identical in the real world. In our experiments, we have determined different periods of recent log data for different users based on their unique behavioral patterns, shown in Fig. 5. As we want not only to update the initial rules based on recency but also to discover new recent behavioral rules for a particular confidence threshold, the determination of such an *optimal period of recent log* can play a primary role to achieve our goal.

Another important finding of our study is that a number of outdated rules can be found for each individual mobile phone user, as their phone usage behaviors are not static in the real world. For a particular context, the user may change her behavior over time, which makes a rule out-of-date and not interested to a particular individual. Besides, these outdated rules, a number of new recent behavioral rules can be found which makes the behavior modeling approach more effective, which has been shown in Figs. 6 and 7.

We have observed a significantly lower prediction coverage, and higher error-rate (%) when using base model compared to our approach. The reason is that rules produced by base model considered the overall behavioral patterns available in the entire dataset. On the other hand, our recency-based method takes into account only the recent behavioral patterns of an individual mobile phone users which is most significant than older ones in the real world. Although our recency-based approach gives better prediction results comparing with base model, this approach is not applicable to an arbitrary dataset. Before applying this recency-based approach, the dataset should ordered as temporal sequences containing the behavioral consistency of individuals. In our approach,

we assume users' behaviors follow a weekly pattern and use a weekly window in order to calculate the conflict score, as time-of-the-week is an important factor impacting on mobile user behavior and the behavior is influenced by time-of-the-week [30]. However, our approach does not depend on any particular time scale, e.g., time-of-the-week, to identify the optimal period of recent log. To model behavior for another time scale, e.g., time-of-the-day, day-of-month, week-of-month, week-of-year or quarter-of-year, corresponding data pre-processing is needed according to these scales before applying the approach.

As the main focus of this work is to output a complete set of recency-based updated rules, i.e., freshness in rules, we process the entire dataset rather than only incremental mining. The reason is that the behavior changing point may not be found in the incremental dataset for a small period, e.g., last 2 weeks, but can be found in the entire dataset for getting the optimal value, in order to produce new recent behavioral rules with high support for individual mobile phone users. In addition to smartphone usage, our recency-based approach can also be applied in other application domains, such as recency-based IoT (Internet of Things) service, recency-based stock market prediction, recency-based healthcare or transport service, recency-based job market analysis, and other relevant areas, where temporal context in time-series and human current interests or preferences are involved.

## Conclusions

In this paper, we have presented a recency-based approach to produce and output a complete set of updated rules according to individual's recent behavioral patterns. For this, we have taken into account four aspects, such as identifying changes in individual's behavior and determining an optimal period of recent log data, identifying and removing the outdated rules that do not represent the recent behavior of an individual, discovering new recent behavioral rules using the determined recent log data, and dynamic management of these rules in order to output a complete set of recency-based updated behavioral rules for individual mobile phone users. The updated rule-set not only contains all the significant rules of individual mobile phone users from their entire phone log data but also expresses their recent behavioral patterns in rules that will be applicable in various real-world mobile applications. Although, we use individual's mobile phone usage and corresponding contextual information as example to illustrate our approach, this recency-based model is also applicable to other application domains in the real world. To assess the usability of this recency-based approach in application level can be a future work.

## Abbreviations

Recency: recent patterns;  $DS$ : entire mobile phone dataset;  $DS_{sub}$ : a subset of data;  $DS_{recent}$ : recent phone log dataset;  $RS_{initial}$ : initial rule set;  $RS_{recent}$ : recent rule set;  $RS_{updated}$ : a complete set of recency-based updated rules;  $R_{new}$ : new recent behavioral rule;  $R_{general}$ : general behavioral rule;  $R_{exception}$ : specific exception behavioral rule;  $W_i$ : initial week;  $W_n$ : most recent week;  $DS_{week}$ : week-wise data;  $assoc_{list}$ : contextual association list;  $assoc_{total}$ : total number of associations;  $conflict_{total}$ : total number of conflicts;  $N_{weeks}$ : total number of weeks.

## Acknowledgements

The authors would like to thank the administrative staff of Swinburne University of Technology, Melbourne, Australia, for their support while doing this work and experiment in their post-graduate research lab. The authors also like to thank Dr. Ashad Kabir, Charles Sturt University, Australia for his relevant support.

**Authors' contributions**

The first and corresponding author IHS carried out the conception, design, and implementation of this research as well as interpretation of experimental results. All the co-authors critically discussed and reviewed the manuscript and provide their helpful suggestions to prepare this manuscript. All authors read and approved the final manuscript.

**Funding**

Not applicable.

**Availability of data and materials**

The datasets generated and/or analysed during the current study are available in the [MIT Human Dynamics Lab] repository.

**Competing interests**

The authors declare that they have no competing interests.

**Author details**

<sup>1</sup> Swinburne University of Technology, Melbourne, VIC 3122, Australia. <sup>2</sup> Chittagong University of Engineering and Technology, Chittagong, Bangladesh.

Received: 4 March 2019 Accepted: 28 May 2019

Published online: 06 June 2019

**References**

1. Sarker IH. Mobile data science: towards understanding data-driven intelligent mobile applications. *EAI Endorsed Trans Scal Inf Syst.* 2018;5:19.
2. Sarker, I.H., Kabir, M.A., Colman, A., Han, J.: Understanding recency-based behavior model for individual mobile phone users. In: *Proceedings of the 2017 ACM International joint conference on pervasive and ubiquitous computing and proceedings of the 2017 ACM international symposium on wearable computers*, USA. New York: ACM; 2017. p. 916–21.
3. Srinivasan V, Moghaddam S, Mukherji A. Mobileminer: Mining your frequent patterns on your phone. In: *Proceedings of the international joint conference on pervasive and ubiquitous computing*, Seattle, WA, USA, 13–17. New York: ACM; 2014. p. 389–400.
4. Zhu H, Chen E, Xiong H, Yu K, Cao H, Tian J. Mining mobile user preferences for personalized context-aware recommendation. *ACM Trans Intell Syst Technol.* 2014;5(4):58.
5. Lee S, Seo J, Lee G. An adaptive speed-call list algorithm and its evaluation with esm. In: *Proceedings of the SIGCHI conference on human factors in computing systems*. New York: ACM; 2010. p. 2019–22.
6. Sarker IH, Colman A, Kabir MA, Han J. Behavior-oriented time segmentation for mining individualized rules of mobile phone users. In: *Proceedings of the 2016 IEEE international conference on data science and advanced analytics (IEEE DSAA)*, Montreal, Canada, IEEE; 2016. p. 488–97.
7. Han J, Pei J, Kamber M. *Data mining: concepts and techniques*. Amsterdam: Elsevier; 2011.
8. Sarker IH, Salim FD. Mining user behavioral rules from smartphone data through association analysis. In: *Proceedings of the 22nd Pacific-Asia conference on knowledge discovery and data mining (PAKDD)*, Melbourne, Australia. New York: Springer; 2018. p. 450–61.
9. Quinlan JR. *C4.5: Programs for machine learning*. Machine learning 1993.
10. Mehrotra A, Hendley R, Musolesi M. Prefminer: mining user's preferences for intelligent mobile notification management. In: *Proceedings of the international joint conference on pervasive and ubiquitous computing*, Heidelberg, Germany, 12–16 September. New York: ACM; 2016. p. 1223–34.
11. Freitas AA. Understanding the crucial differences between classification and discovery of association rules: a position paper. *ACM SIGKDD Expl Newslett.* 2000;2(1):65–9.
12. Witten IH, Frank E. *Data mining: practical machine learning tools and techniques*. New York: Morgan Kaufmann; 2005.
13. Agrawal R, Imieliński T, Swami A. Mining association rules between sets of items in large databases. In: *ACM SIGMOD Record*, vol. 22, New York: ACM; 1993. p. 207–16.
14. Agrawal R, Srikant R. Fast algorithms for mining association rules. In: *Proceedings of the international joint conference on very large data bases*, Santiago Chile, vol. 1215; 1994. p. 487–99.
15. Cheung DW, Han J, Ng VT, Wong C. Maintenance of discovered association rules in large databases: an incremental updating technique. In: *On IEEE data engineering, proceedings of the twelfth international conference*. 1996. p. 106–14.
16. Cheung DW-L, Lee SD, Kao B, et al. A general incremental technique for maintaining discovered association rules. In: *DASFAA*, vol. 6. 1997. p. 185–94.
17. Xu B, Yi T, Wu F, Chen Z. An incremental updating algorithm for mining association rules. *J Electron.* 2002;19(4):403–7.
18. Zhang Z, Li Y, Chen W, Min F. A three-way decision approach to incremental frequent itemsets mining. *J Inform Comput Sci.* 2014;11(10):3399–410.
19. Li Y, Zhang Z-H, Chen W-B, Min F. Tdup: an approach to incremental mining of frequent itemsets with three-way-decision pattern updating. *Int J Mach Learn Cybern.* 2015;2:1–13.
20. Amornchewin R, Kreesuradej W. Mining dynamic databases using probability-based incremental association rule discovery algorithm. *J UCS.* 2009;15(12):2409–28.
21. Thusaranon P, Kreesuradej W. A probability-based incremental association rule discovery algorithm for record insertion and deletion. *Artif Life Robot.* 2015;20(2):115–23.

22. Phithakkitnukoon S, Dantu R, Claxton R, Eagle N. Behavior-based adaptive call predictor. *ACM Trans Autonom Adap Syst*. 2011;6(3):21–12128.
23. Sarker IH, Colman A, Kabir MA, Han J. Phone call log as a context source to modeling individual user behavior. In: *Proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing (Ubicomp): Adjunct, Germany*. p. 630–4. New York: ACM; 2016.
24. Bell S, McDiarmid A. Nodobo: Mobile phone as a software sensor for social network research. In: *Vehicular technology conference*. New York: IEEE; 2011.
25. Pielot M. Large-scale evaluation of call-availability prediction. In: *Proceedings of the international joint conference on pervasive and ubiquitous computing*; New York: ACM; 2014. p. 933–7.
26. Eagle N, Pentland AS. Reality mining: sensing complex social systems. *Person Ubiquitous Comput*. 2006;10(4):255–68.
27. Kim J, Mielikäinen T. Conditional log-linear models for mobile application usage prediction. In: *Machine learning and knowledge discovery in databases*. New York: Springer; 2014. p. 672–87.
28. Liao Z-X, Pan Y-C, Peng W-C, Lei P-R. On mining mobile apps usage behavior for predicting apps usage in smart-phones. In: *Proceedings of the 22nd international conference on information & knowledge management*. New York: ACM; 2013. p. 609–18.
29. Halvey M, Keane MT, Smyth B. Time based segmentation of log data for user navigation prediction in personalization. In: *Proceedings of the international conference on web intelligence, Compiegne, France, 19–22 September*. Washington: IEEE computer society; 2005. p. 636–40.
30. Halvey M, Keane MT, Smyth B. Time based patterns in mobile-internet surfing. In: *Proceedings of the SIGCHI conference on human factors in computing systems, Montreal, Quebec, Canada, 22–27 April*. New York: ACM; 2006. p. 31–4.
31. Bordino I, Donato D. Extracting interesting association rules from toolbar data. In: *International conference on information and knowledge management*. New York: ACM; 2012.
32. Paireekreng W, Rapeepisarn K, Wong KW. Time-based personalised mobile game downloading. In: *Transactions on Edutainment II*; 2009. p. 59–69.
33. Rawassizadeh R, Tomitsch M, Wac K, Tjoa AM. Ubiqlog: a generic mobile phone-based life-log framework. *Person Ubiquitous Comput*. 2013;17(4):621–37.
34. Mafrur R, Nugraha IGD, Choi D. Modeling and discovering human behavior from smartphone sensing life-log data for identification purpose. *Human-centric Comput Inform Sci*. 2015;5(1):31.
35. Sarker IH. A machine learning based robust prediction model for real-life mobile phone data. *Internet Things*. 2019;5:180–93.
36. Phithakkitnukoon S, Dantu R. Adequacy of data for characterizing caller behavior. In: *Proceedings of KDD inter. Workshop on social network mining and analysis (SNAKDD 2008)*. Citeseer; 2008.
37. Barzaqi OO, Loke SW. Adapting the mobile phone for task efficiency: the case of predicting outgoing calls using frequency and regularity of historical calls. *Person Ubiquitous Comput*. 2011;15(8):857–70.
38. Bergman O, Komninos A, Liakopis D, Clarke J. You never call: demoting unused contacts on mobile phones using dmtr. *Person Ubiquitous Comput*. 2012;16(6):757–66.
39. Stefanis V, Plessas A, Komninos A, Garofalakis J. Frequency and recency context for the management and retrieval of personal information on mobile devices. *Perv Mobile Comput*. 2014;15:100–12.
40. Sarker IH, Colman A, Kabir MA, Han J. Individualized time-series segmentation for mining mobile phone user behavior. *Comput J*. 2018;61(3):349–68.
41. Sarker IH, Kabir MA, Colman A, Han J. Identifying recent behavioral data length in mobile phone log. In: *Proceedings of the 2017 ACM EAI international conference on mobile and ubiquitous systems: computing, networking and services (MobiQuitous 2017)*, Melbourne, Australia 2017.
42. Sarker IH. Understanding the role of data-centric social context in personalized mobile applications. *EAI Endorsed Trans Context-aware Syst Appl*. 2018;5:15.

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)