This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

Smart Diary: A Smartphone-based Framework for Sensing, Inferring and Logging Users' Daily Life

Jilong Liao, Zhibo Wang, Lipeng Wan, Qing Cao and Hairong Qi

***Abstract*—In this paper, we present Smart Diary, a novel smartphone based framework that analyzes mobile sensing data to infer, predict, and summarize people's daily activities, such as their behavioral patterns and life styles. Such activities are then used as the basis for knowledge representation, which generates personal digital diaries in an automatic manner. As users do not need to intentionally participate into this process, Smart Diary is able to make inferences and predictions based on a wide range of information sources, such as the phones' sensor readings, locations, and interaction history with the users, by integrating such information into a sustainable mining model. This model is specifically developed to handle heterogeneous and noisy sensing data, and is made to be extensible in that users can define their own logic rules to express short-term, mid-term, and long-term event patterns and predictions. Our evaluation results are based on the Android platform, and they demonstrate that the Smart Diary framework can provide accurate and easy-to-read diaries for end users without their interventions.**

I. Introduction

A diary is a collection of records on what has happened over a period of time, usually sorted by dates. A personal diary usually records one's experiences, thoughts, and feelings on events in the outside world. Some personal diaries, such as those by Anne Frank, have become widely read books and the basis of plays and films. Besides their value in terms of being a hobby and a literature source, diaries also serve important roles for researchers in social science. Some studies use diaries to study the correlation between certain activities (e.g., exercise) and health problems (e.g., diabetes) in certain populations [34]. However, recent studies have reported the insufficiency of this research method due to the quality on diaries collected from a group of volunteers [11]. The contents are usually incomplete, fragmented, or incorrect. Furthermore, long-term studies with consistent volunteer participation are particularly challenging, if possible at all.

With the emergence of the Internet, the form of diaries has undergone significant changes. Online journals, blogs, micro-blogs, tweets, and Facebook status all contain continuous updates on a person's life events. Moreover, the complexity of people's lives have increased dramatically with the tight interweaving of both "online" and "offline" lives. On the other hand, however, with more and more activities attracting our attention, we have less and less time to write down the stories belonging to our unique life.

Motivated by the need to not only keep user's personal lives, but also to provide verified data sources for social studies, we propose a fully software-based approach to generate personalized diaries. Our approach is enabled by the widespread adoption of smartphones in recent years. According to a report by IDC [21], the worldwide smartphone vendors will ship a total of 918.6 million smartphones in 2013, up 27.2% from the previous year. These smartphones are usually equipped with both a wide range of sensors and communication capabilities such as Wi-Fi and Bluetooth. Our key idea in this paper is that, as the user carries the smartphone, this device will be able to detect the environment context through its sensors and infer users' behavior. For instance, Fig 1 presents a user's life activities (e.g. motion, SMS, phone call, etc.) throughout a day, from which we can infer many different types of user-interested events and record them. The inference results are then summarized into human-readable forms (diaries). We name this application as "Smart Diary", as it is purely based on software, and does not require user intervention.

Although the central idea of Smart Diary is simple, there are numerous challenges that we have to address to bring it into reality. Because of the battery limitations on the smartphone, it is usually not acceptable to keep the application active for extended periods of time. Therefore, tradeoffs have to be made in terms of the amount of data that the application collects, and the accuracy that it achieves. Specifically, we summarize the following challenges that our design addresses:

First, Smart Diary has to address the challenge of user privacy. Because the dairies generated contain unique daily events of users, such as their entertainments, social contacts, and health conditions, it is necessary to ensure that such information is not leaked to others. This problem is particularly urgent as recently exposed security holes have demonstrated that it is possible for these phones to be hacked through malware or NFC (near field communication) techniques [2].

Second, as each user has their own life styles and experiences, we need a mechanism to support the personalization of Smart Diary. In particular, different users may interpret the same sensing readings in their own manners, leading to user-specific output in diary contents. Indeed, the output from Smart Diary should reflect the user's personality, and consist of the most valuable and interesting experiences, which can be different from user to user. In addition, the system's framework should be adaptive to the dynamic requirements of the users.

Jilong Liao, Lipeng Wang, Qing Cao and Hairong Qi are with the Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville, TN, 37912 USA.
Zhibo Wang is with the School of Computer, Wuhan University, China, 430072, and Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville, USA, 37909. (Emails: {jliao2, zwang32, lwan1, cao, hqi}@utk.edu).
Copyright ©2013 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

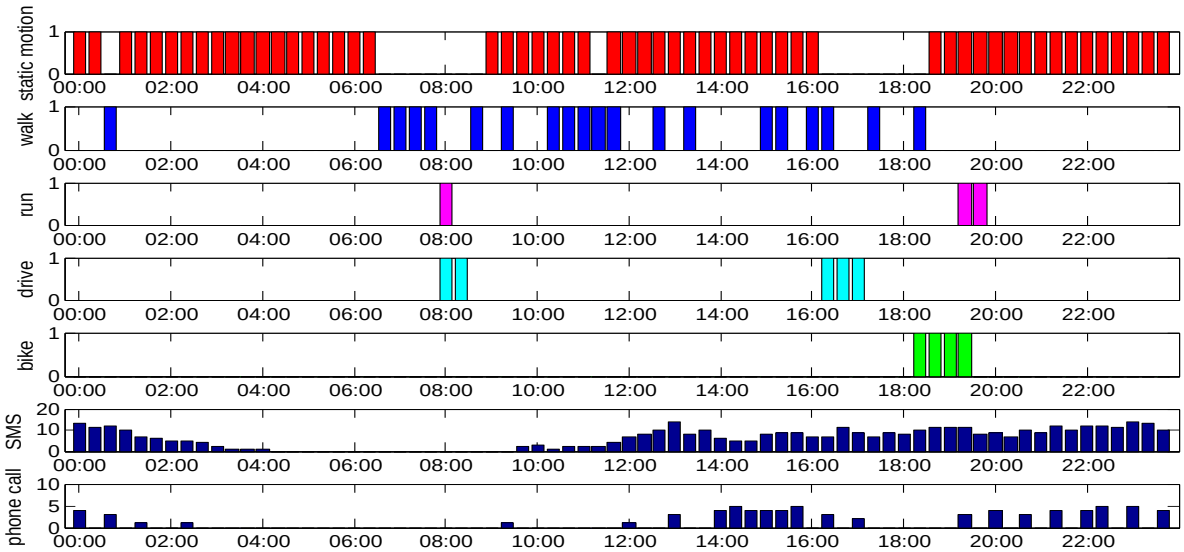


Fig. 1. A typical user's life context in a day. (Each slot is a 10-minute window.)

To address these challenges, we develop a variety of techniques, and summarize their major contributions:

- We present Smart Diary, a fully automatic software system that allows highly intelligent and human-centric generation of diaries based on the sensing readings from users' smartphones. The whole process is flexible and customizable, but does not require human input.
- To address privacy concerns, we perform all activity recognition and context inference on the phone side rather than on a centralized server, so that no data need to be transmitted over the Wi-Fi network. To achieve this local processing goal, we present energy-efficient and lightweight algorithms that adaptively classify users' activities based on sensing data, based on which we are able to infer higher level events.
- To address the needs of different users, we present a sustainable and extensible event mining model plus the logic language for rule-based event inferences, where users could follow their own inference rules for higher level event generation. These rules are written in a logic language manner, which make them easy to write and modify. We also develop a feedback loop so that users can provide optional opinions on the generated diaries, so that the system can learn continually over time to improve its diary generating capabilities.

In the long term, we believe that this system we have developed will not only prove useful for people to recall their life events, but also very helpful for social studies where accurate profiles of user activities are needed. Our evaluation results on Smart Diary illustrate that it has achieved its goals with satisfactory performance.

The rest of this paper is organized as follows. We discuss the related work in Section II. We present the design of Smart Diary in Section III. In Sections IV-VII, we present the detailed implementation of Smart Diary. Section VIII systematically evaluates the performance of Smart Diary. Finally, Section IX gives the conclusions and discuss future work.

II. Related Work

Modern smartphone platforms are equipped with an impressive set of embedded sensors. Such advances have made it possible to infer people's behaviors to provide better services to people, e.g., tourist recommendation [14] and guidance [13]. Our research in this paper is built on top of these previous ones, so we briefly survey them in this section.

Mobile Sensing with Activity Understanding: One key topic in early work on context-aware mobile sensing has been focusing on understanding people's activities [8]. Although this problem has been extensively studied in the context of wearable sensors [5], [19], [17], those who use smartphones adopted different approaches due to the unique combination of sensor types available [25], [12], [22], [23], [26], [30], [36]. Most efforts used the accelerometer as the the primary sensor to infer a person's activity, where the problem is formulated as a classification problem that is solved with models such as the Hidden Markov Model (HMM). Later applications used smartphone sensors for other purposes such as writing and localization, as reported by Cenceme [24], TagSense [28], FindingMimo [33], SoundSense [22], and UbiFit[10].

Smartphone Usage Traces and Contexts: Previous research efforts have also studied the patterns and contexts of smartphones usage. LiveLab [32] presented a comprehensive measurement of how a person uses smartphones in the field. Later applications like [20] tried to use smartphone to analyze a user's mood. In [35], the authors characterized how people interact with the mobile web. In addition, [38] used reinforcement learning to learn the context in wireless network, and [3] analyzed mobile objects' social relationships. All these efforts relied on the smartphone usage context, which partially inspired our work. But our paper is significantly different in that Smart Diary aims to generate diaries from these contexts, a goal that was not attempted in these previous efforts.

Semantic Representation of Sensor Data: Previous work has also attempted different ways of semantic representation and analysis of sensor data. Semantic Streams [37] proposed

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

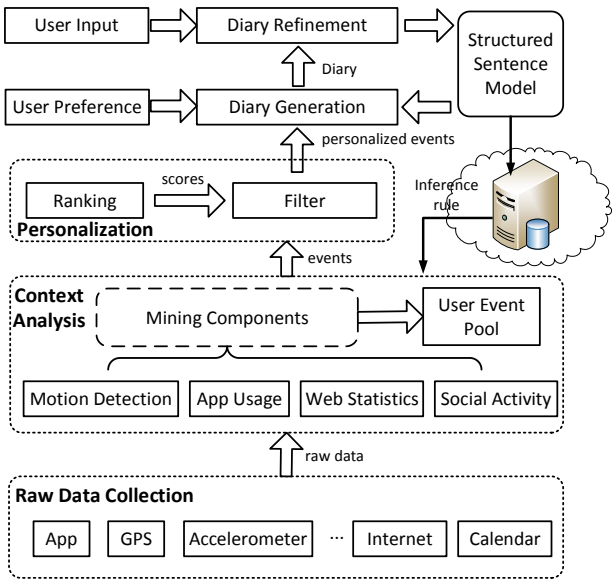


Fig. 2. Framework of Smart Diary

a solution to represent sensors in a semantic way so that it can respond to users' queries. The authors in [6] presented a study of snippets, or short messages based on sensors, to help users write down diaries more easily. A similar work is done by a commercial organization Narrative Science [1], who has developed techniques to synthesize natural language from raw data. More recently, [16] presented a method for generating personal diaries in a semiautomatic manner. Although it also tries to generate diaries, it focuses on activity recognition only, and does not address challenges for higher level event representation. Therefore, Smart Diary differs significantly from these efforts in that it handles heterogeneous data (both numeric and text) from smartphones, utilizes multiple types of sensors, as well as provides an end-to-end solution in practice.

III. System Design and Overview

In this section, we first present a high level view of Smart Diary and then elaborate those components in details in the following sections.

The framework is shown in Fig. 2, which includes four layers: raw data collection, context analysis, event personalization, and diary generation. Through those four layers, Smart Diary captures critical events according to users' preferences, and automatically generates diaries to the user.

A. Raw Data Collection

Equipped with a range of sensors, smartphones provide an ideal platform for user data collection. We are particularly interested in six representative data sources: motion activities, location data, app usage, calendar events, phone calls or SMS messages, and web history.

Motion Activity: To capture motion activities, we compared the performance of using accelerometer alone versus using accelerometer and gyroscope together. Based on our findings, we observe that it is sufficient to use accelerometer readings alone to infer users' activities such as driving, walking, sitting, and playing games.

Location Data: The accurate location readings provide us hints on the user's most visited places (such as office, restaurants, etc.) and commuting routes. We use these locations to infer the location context of users' activities.

App Usage: The usage patterns of apps provide an attractive resource to identify a user's behavior, social activities and personal interest. In our system, we log such activities over time to support dynamic inference of users' interest.

Calendar Events: As an explicit resource that reflects users' agenda, the events in a calendar usually give us the most direct insight on the user's life, such as their business meetings, friend parties, and travel plans.

Phone Calls and SMS Messages: These information sources provide us with the users' "off-line" social groups and the interaction of the user with his/her friends via phone calls and SMS messages.

Web History: The history from the smartphone's browser helps Smart Diary to learn and analyze those topics that the user is interested in, and monitor how the users' personal preferences change with time.

B. Context Analysis

The context analysis layer takes the processed raw data collected in the lower layer as input, so that it may extract multiple types of events from the users' life. Each event is produced by a mining component, and we develop multiple types of mining components in the system. To better manage the reuse of resources, we propose a novel *sustainable mining model*, which decomposes a mining component's algorithm procedures into separate processing units. These units will continuously shuffle raw data, and provide the relevant ones to all the mining components where events are assembled. Specifically, we classify the events into three classes: entertainment activities, social activities, and health conditions. Processing of these events either adopts existing algorithms or relies on user-specific logic rules.

C. Event Personalization

However, the events extracted by the context analysis layer are purely objective. From the perspective of the users, however, some events are more meaningful than others. Event personalization, as its name suggests, allows Smart Diary to select those most interesting events for a user based on their preferences. Two major modules are involved in this layer: the ranking module and the filtering module. The ranking module calculates the importance based on our three criteria, then ranks these events. The filtering module only provides those most interesting events as output, which are handed over to the next layer for diary generation.

D. Diary Generation

Given the personalized events, this layer translates the events into recognizable sentences. To facilitate this process, we develop a model called the narrative structured sentence model. One key feature of this model is that it uses regular expression formats to construct natural language sentence

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

TABLE I
THE RAW DATA DETAILS.

Data type	Details
Accelerometer	(`timestamp`, `acc_x`, `acc_y`, `acc_z`)
Location Service	(`timestamp`, `latitude`, `longitude`, `speed`)
Phone Call/SMS	(`timestamp`, `incoming number`, `outgoing number`, `duration`)
App Usage	(`timestamp`, `app name`, `duration`)

templates. During runtime, real event properties, such as the time of the day and the user's activities, replace these wild cards in the regular expression templates to generate diary outputs. Furthermore, to make language output natural, each type of events has multiple corresponding structured sentence templates for use.

After diaries are generated, we also provide an optional step where the user may provide additional feedback regarding the generated diaries. For example, the user may want to share their sentence with other users or revise an existing sentence. In practice, this stage is not only useful for improving the quality of the diaries, but also for enhancing the narrative structured sentence model by adopting better structured sentences for each event. Note that the raw data like sensor readings and locations are not collected, the user's privacy concern has been preserved.

IV. RAW DATA COLLECTION

Equipped with multiple types of sensors, a smartphone can obtain heterogeneous data ranging from motion activities to location data. Among the various data sources, we select the following types of sensors: the accelerometer, GPS location service, phone call history, SMS service, and app usage. Table I summarizes the types of raw data we collect to mine the user's behavior patterns and daily activities. For example, we use the accelerometer and location data to infer whether a user is working in his/her office or driving on the highway. In general, we are interested in the "when-where-who-what" aspect for each event, namely, what the user is doing, who is involved in, where the event happens, and when it happens.

To retrieve raw data, the simplest way is to poll the source directly and periodically. For example, we can set up a timer to detect the current active app used by the user. Unfortunately, the raw data coming out from the various sources may not be exactly periodical (e.g. the accelerometer in Android OS does not sample in equal rate due to limitations of the Android operating system), the mining component may need different sampling frequency, and the loss of GPS signal may unexpectedly happen. Therefore, we propose a layer of *middle buffer* between the source and the raw data. The sensor source frequently updates the readings in the middle buffer's memory, while the independent timers trigger the data collection on their own schedules. The collected data, hence, reflect the most recently measurement. This middle buffer is especially useful when data sources are unreliable: for example, if the smartphone loses the GPS signal in an indoor environment, as long as the user does not walk out of the building, their location is still recorded as the last GPS update as they move in the building because the middle buffer has not been changed. In this way, we can still use the last updated data when the sensor is not responsive and we can sample the data in different frequency.

Initially, to detect motions, we used both accelerometer and gyroscope in our study. We collected 7 user's motion sensor data for about 1 month and asked them to label their motion activity. Then we use the first half of the data to train our classifier, and use the rest of the data to test it. We calculate the standard deviation features of both accelerometer and gyroscope instead of using the original data values (Fig. 3). However, based on the empirical study, shown in Table II, we conclude that using the accelerometer alone is enough even though using both accelerometer and gyroscope sometimes achieves a slightly higher accuracy. The primary reason is that the gyroscope's sampling rate is much higher than the accelerometer, leading to much redundant data, which explains why there is only a 3% of accuracy increase. On the other hand, keeping the gyroscope on will significantly increase the energy overhead. Therefore, we make the trade-off by only reading the accelerometer.

V. CONTEXT ANALYSIS

A. The Sustainable Mining Model

We now describe the context analysis layer, which infers events related to a person's daily life using the raw data. As an example to illustrate this layer's functionality, consider the following two events:

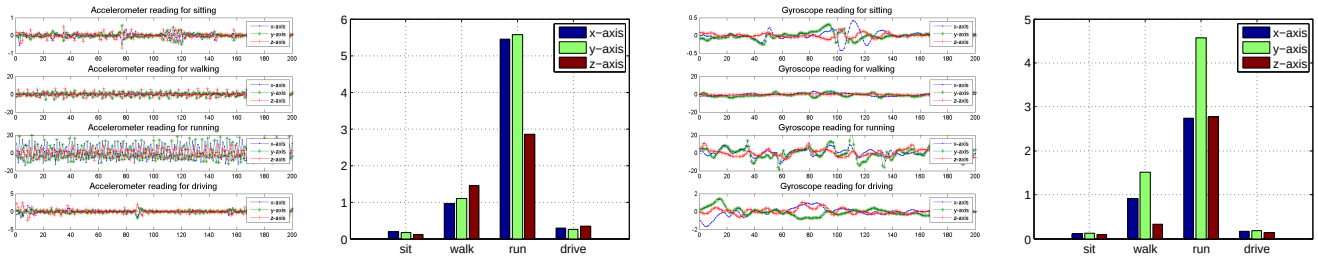
- The user is quite busy with his/her study.
- The user has increased his/her level of social activity.

To correctly differentiate between these two events, the context analysis layer needs to exploit different types of sensors and their readings. In particular, the most useful data sources include the most visited places (e.g., library vs. restaurants and clubs), motion activities (e.g., mostly sitting vs. standing and talking), and phone call records (e.g., reduced durations and frequency of phone conversations vs. increased durations and frequency). Such data sources are used to infer users' activities and predict their life patterns.

One challenge we face in inferring users' activities is that there are many different types of possibilities on how to process users' sensing data. To facilitate code reuse, an interesting observation is that for each type of sensing data, its processing algorithms usually have limited complexity, and could be shared with other sources of data. For example, the Kalman filter is a widely used signal processing algorithm that can be applied to more than one type of raw data without modifying the algorithm itself. Based on this observation, we develop the concept of Mining Components (MC), which is a conceptual encapsulation of a processing algorithm based on certain input data. Multiple MCs can be applied to the same type of data, such as users' activities and phone call statistics. MCs can also be hierarchical, where the output of one MC can be used by other MCs as input. Each MC is implemented as a system background service which is continueously accumulating and analyzing data until it is enough to extract an event. The extracted events will be buffered in the system until it is

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal



(a) Raw accelerometer readings (b) Accelerometer standard deviation (c) Raw gyroscope readings (d) Gyroscope standard deviation

Fig. 3. Raw readings and the standard deviation feature

TABLE II
MOTION DETECTION COMPARISON

	Naive Bayesian (%)			Decision Tree (%)		
Motion Type	Accelerometer	Gyroscope	Both	Accelerometer	Gyroscope	Both
Sit	94.91	97.53	97.4	96.9	95.4	97.0
Drive	90.78	38.73	87.91	93.67	96.55	95.19
Walk	91.06	89.62	89.75	93.81	91.37	92.25
Run	93.13	97.25	98.44	98.75	95.31	96.87

consumed by other MCs or event personalization phase. Based on the concept of MCs, our core contribution in the context analysis layer is called the *sustainable mining model*, which decomposes inference and prediction tasks into a set of MCs that are integrated together through input/output data flows.

As an example, we want to extract two events in the diary: one is the person's living habits such as exercise frequency, and the other is the person's favorite restaurants. For the first event, we need the motion mining component to detect if the user is exercising, and the location mining component to tell if the user is in a gym or outdoor. Existing location-based services like Google Maps and Yelp maintain a large database for the users to query and learn, and they provide third party APIs so that we can fully utilize their database. The second event needs the phone usage mining component to see if the user is using certain apps to search restaurants, and the location mining component as well. So the location mining component can be shared and reused by the two events. Additionally, the statistical phone call information can be reused across different MCs so that they can infer events at different levels: a lower level MC may infer that the user has made phone calls at a higher frequency, while a higher level MC may conclude that the user may have increased their level of social activity.

There are multiple advantages of this sustained mining model. First, having multiple MCs allows us to handle all types of events. Indeed, it is very hard, if possible at all, to develop one single mining algorithm that simultaneously supports inference tasks of all types of events. Second, as the user's preference may change, we frequently need to adjust the overall mining algorithm. In such cases, the sustainable mining model reduces these efforts by allowing dynamic reconfiguration of MCs, replacing old ones with new ones dynamically. Third, the overall efficiency of the system is increased because the potentially redundant functionality will be eliminated by following a modular design of MCs. Finally, if a new type of event needs to be supported, we only need to analyze its MC dependence, and achieve the goal by reusing existing MCs as much as possible.

B. Rule-based Event Inferences

Although some MCs are implemented based on existing algorithms, users also need to specify flexible event processing rules that handle multiple types of activities in users' lives. To this end, we develop a model of rule-based event inference, which is based on a language grammar that is similar to logic programming, such as Prolog. This language adopts logic symbols to connect results between mining modules. As an example of the logic language to infer an event, we use a *shopping event* as an example, where the inference occurs when the user's activities satisfy several constraints including spatial, temporal, and motion requirements. Specifically, a user can be considered as shopping if this user's location belongs to a shopping center or a mall, the motion is mostly walking, the day is not Monday or Tuesday, and the duration is longer than a threshold, say, 10 minutes. Accordingly, we can write the following logic rule to infer the event:

```
shopping(User, Day) :-
   member(Location, ["mall", "shopping center"]),
   Motion is "walking",
   Duration > 600,
   not member(Day, ["Monday", "Tuesday"])
```

Note that, the logic predicates are inherently hierarchical, which means that lower level predicates, when satisfied, will trigger higher level predicates to become true. This is particularly useful for expressing some long-term behavior of users, such as their habits. To this end, we illustrate its use by specifying the frequency of a certain type of activity, say, shopping, and consider a user to have a shopping habit if he/she goes to shopping at least twice each week, once in a

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

TABLE III
THE SUMMARY OF EVENTS AND SENSING SOURCE REQUIREMENTS

Event Name	App Statistics	Contact Statistics	Visited Places	Motion Detection	External Device/Module
App Usage	√				
Shopping			√	√	
Dinning Behavior			√	√	
Life Pace	√		√	√	
Friendship Management	√	√			
Party Activities		√	√	√	
Health and Fitness			√	√	√

weekday, and another in a weekend. This rule can be written as follows:

```
habit(User, "shopping") :-
   shopping(User, Weekday),
   shopping(User, Weekend),
   member(Weekday, ["Wednesday", "Thursday",
        "Friday"]),
   member(Weekend, ["Saturday", "Sunday"])
```

Note that in describing these rules, we use the same set of keywords as Prolog, such as the keyword `member` and `is`. The logic language rules have the advantage that each user may contribute their own rules, so that we can have a database of inference rules for all other users. A back-end server can maintain this database, and each user may select a unique subset of rules according to their needs.

C. A Study of Representative Events

Based on the sustainable mining model, we now describe the MCs and rules we have integrated into the Smart Diary framework. Generally, we classify these MCs into three categories: entertainment activities, social activities, and health conditions.

Entertainment Activities: We consider the following representative entertainment activities: app usage, shopping, dining, and life style. For app usage, we are interested in the top apps that this user spends most time on, as well as the distribution statistics between different apps. For shopping, we want to record the frequency, location, and type of shopping activities, e.g., in grocery stores versus in a shopping mall. The dinning behavior is represented by the favorite restaurants and how often the user visits them. The life style reflects the pace of the user's life: whether it is busy or relaxed. For example, a user who works very hard may have a very simple daily routine: "home → office → home".

To develop MCs for these four types of events, we observe that only three types of sensing sources are needed: app statistics, location information, and motion activities. A summary of the events and their sensing source requirements are shown in Table III. The app statistics help us answer the app usage patterns; the location data can be mined to find shopping, dinning, and life styles; the motion activities can be used to increase inference accuracy: for example, those types of activities such as walking or sitting will help confirm the higher level activities of the user.

Social Activities: Similar to entertainment activities, we develop multiple MCs for social activities. In our prototype, two MCs are supported: friendship management and party activities. For the first type, we mine the user's contact history and app usage (especially those social network apps) to determine their level of friendship management activities. For the second type, we mine the contact, location, and motion activities simultaneously to detect party-related activities.

TABLE IV
SAMPLE GSR DATA CLASSIFICATION RESULTS

Subject	Amusement	Fear	Relax	Sadness
1	5	0	9	1
2	1	3	0	1

Fig. 4. GSR wearable sensor

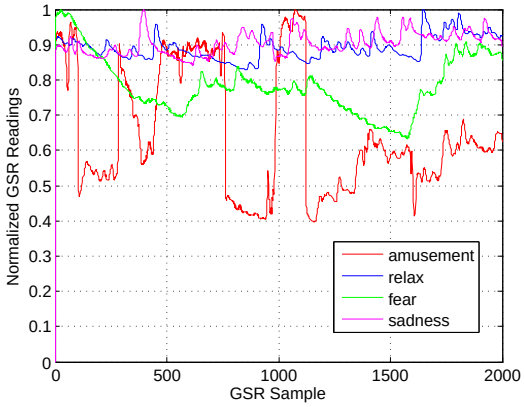


Fig. 5. Sample GSR sensor reading

Health Conditions: The health conditions of the user are

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

most closely related to their motion activities. For example, a user may have certain unhealthy habits such as spending most of his/her time siting in the office or sleeping too late. By exploiting the output from sensor readings corresponding to motion activity, we can detect these healthy condition of the user, and provide suggestions in the diary for user's awareness and improvement. In addition, Smart Diary can also incorparate external device or module's data, such as wearable Galvanic Skin Response (GSR) signal monitor (shown in Fig. 4), to extract human emotion information like stress, fear, amusement and sadness. As an example, we utilize the data set and classification technique provided by [15], and incorporate the GSR mining component into Smart Diary. The sample GSR data readings are shown in Fig. 5, then we use this classifier and obtain a sample detection results in Table IV. Finally, a sample logic rule about a user's emotional health using GSR mining component can be described as the following:

```
emotion(User, "healthy") :-
   Amusement + Relax > Sadness + Fear,
   Amusement is GSR("amusement"),
   Relax is GSR("relax"),
   Sadness is GSR("sadness"),
   Fear is GSR("fear")
```

In summary, observe that, as shown in Table III, the sustainable mining model considerably decreases system complexity by decomposing inference tasks into multiple MCs. For the sensing sources, although app statistics and contact statistics are easily retrieved, we implement the location mining through the kernel density estimation method [31], and the motion detection through a combination of the Naive Bayesian method and the Decision Tree method. Both methods are chosen because they have well known performance advantages.

D. Adaptive Component Selection

After we deploy the mining components and inference rules to the user's device, the mining process become a static and pre-define inference in terms of the model. However, the deployed template can neither cover all the cases nor satisfy the needs of the user. For instance, we may use GSR mining component to obtain the user's emotional information, but the user may even want to know why he is in an emotion like "relax". This requires a new inference rule from the existing component, such as we can combine the motion and GSR component together so that we can collect the frequency of "relax" emotion occurs under certain motion state like "running". Thus, if the user can give Smart Diary the target event when the event happens, we can distinguish the major factors leading to this event by collecting the exisitng component's output for a long enough time.

Suppose s_i stands for the event of the user input and $m_1, m_2, ..., m_q$ are the existing mining components. We also maintain a vector of the mining components' influence statistics associated with each s_i. As long as the exising mining component m_j has an output, we treat the component as the one can infer s_i and update the influence statistics of m_j. As an example, Table V shows two user events and the mining components' influence statistics vectors. As we can see, the major components could infer the user events are different: s_1 can be inferred by m_3 and m_4 while s_2 can be inferred by m_2 and m_5.

TABLE V
SAMPLE MINING COMPONENT USAGE TABLE

User Events	m_1	m_2	m_3	m_4	m_5	m_6
s_1	0	0	5	7	2	4
s_2	1	6	1	2	5	0

One challenge in this approach to adapt the component selection is how can we distinguish the influential mining components. First, we calculate the normalized influential factor $H(m_j)$ in Eq. 1. Then, we sort the mining components by the normalized influential factor, and if the accumulated influence of the first t mining components (defined in Eq. 2) is larger than the user's confidence preference μ, then the first t mining components will be selected and be used to mine the associated user event in the future. When the user do not want the event appear again, he can just remove the event from the user event pool as shown in Fig. 2.

$$H(m_j) = \frac{m_j}{\sum_{k=1}^{p} m_k}, \tag{1}$$

where p is the number of mining components in the system.

$$G(s_t) = \sum_{k \,\in\, top\ t\ components} H(m_k). \tag{2}$$

Suppose we have the user confidence preference $\mu = 0.6$, we can calculate the accumulate influential factor for s_1 as the following from Table V. So we can conclude that m_3 and m_4 are the two influential components to infer the user event s_1.

$$G(s_1) = \frac{7}{18} + \frac{5}{18} > 0.6.$$

VI. EVENT PERSONALIZATION

Although the context analysis layer continuously extracts events based on the sustainable mining model, not all of them should be included in the final diary output for two reasons. First, the number of events can be huge as the number of mining components increases, leading to bloated output with redundant information. For instance, the system may write around 50 sentences about your motion activity throughout a day (e.g. walk into multiple buildings, drive to different locations, etc.). Second, each user may have a different taste on diary content and length, and does not want to adopt the same template. Therefore, the generated diary should take such preferences into consideration.

Motivated by this observation, we develop the next layer, *event personalization*, which filters diaries according to the user's expectations. In particular, we define the personalization as the following: suppose we have an event set $\mathbf{E} = \{e_1, e_2, ..., e_n\}$, we hope to order these events and obtain a new set $\hat{\mathbf{E}} = \{\hat{e}_1, \hat{e}_2, ..., \hat{e}_n\}$, where events are sorted based on their importance to a particular user. The ranking module (Fig. 2) in the system is responsible for the task.

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

TABLE VI
EXAMPLE OF EVENT PERSONALIZATION.

Events	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6
$e1$	2	0	2	0	0	3
$e2$	2	5	3	4	4	0
$e3$	1	2	0	0	3	1

The event importance reflects the importance of an event objectively and decides how frequently an event will appear in the diary. It should satisfies the following properties,

- An event is getting more and more important as its frequency increases, and highest importance should be those events which occurs many times within a small number of days.
- An event is less important when it occurs a few times a day, or occurs in a lot of days.
- The event with lowest importance is the one which occurs everyday. This leads to tedious information in a diary.

Mathematically, the event importance factor $T(e_i)$ over a D-day window is defined as:

$$T(e_i) = \frac{|e_i|}{|E|} \cdot \log(\frac{|D|}{|\{d \in D : e_i \in E_d\}|}), \quad (3)$$

where E is the set of all events, E_d is the set of events in day d, and $|\{d \in D : e_i \in E_d\}|$ indicates the number of days that e_i appears. This formula intuitively represents the relative importance of events in a user's life through their frequency. Observe that if an event happens for only a few times during a period of time, it may be interesting because it may indicate something that is not so often, then it is valuable to be recorded in the diary. For example, if a user goes to the shopping mall every weekend, such events should be recorded. On the other hand, if something happens very frequently, it tends to be less interesting since it may be something trivial, e.g., walking from the parking lot to the office. Equation 3 reflects such differences by taking into account the number of times a particular event occurs among all possible event candidates over a period of time.

More specifically, Table VI shows 3 events' frequency record in a 6-day window. As we can see, $e2$ is the most frequent events extracted from the system which appears every day, and $e1$ and $e3$ are less frequent. If we write a sentence for each event, $e2$ will be overwhelming the diary. But Eq. 3 changes the weight by their event importance as follows. As we can see, $e1$ is becomes the most important event while $e2$ and $e3$ are equivalent. So the event importance decreases the weight of the redundant events.

$$T(e_1) = \frac{7}{30} log(\frac{6}{3}) = 0.070$$

$$T(e_2) = \frac{16}{30} log(\frac{6}{5}) = 0.042$$

$$T(e_3) = \frac{7}{30} log(\frac{6}{4}) = 0.041$$

The equation assigns a score for each event, and the filtering module relies on the score to quickly select those events per a user's request. Algorithm 1 illustrates the quick selection process running in $O(n \log k)$ time and $O(k)$ space, where n is the number of events, and k is the user preferred diary length, as measured by the number of events.

Algorithm 1 Event Filter Algorithm

Input: The event sequence: *events*.
Input: The user's preferred length k.
Output: Top k event *candidate*.

```
Initial candidate ← min heap.
for all e in events do
    if candidate.size < k then
        candidate.insert(e)
    else if e.rank > candidate.top().rank then
        candidate.insert(e)
        candidate.pop()
return candidate
```

VII. DIARY GENERATION

A. The Narrative Structured Sentence Model

We now describe the diary generation layer. Essentially, the problem is a variant of natural language processing (NLP) problem (e.g. "speech transcription"), except that the signal source here is not from the speech, but from the events generated by the personalization layer. Traditional ways to handle the NLP problem can be divided into two categories: one is based on statistical methods, and the other is based on language grammar rules and semantics. However, neither of the two approaches fits our problem well for the following reasons: first, the statistical NLP solution requires explicit data features to perform the classification tasks. Usually it needs a huge data set (in GB size) to support the training phase for complex tasks, so that the output can be accurate. In the Smart Diary case, the source is a series of data which are inherently expensive to collect (which requires deploying smart phones to a large population over an extended period of time). Therefore, it is hard to build a large training data set. The second NLP approach is based on analysis of large amount of annotated data to achieve accuracy. However, with the limited memory and battery capacities of smartphones, we find this approach not suitable either. Therefore, we decide to pursue an alternative approach where the events are directly translated into language through a novel and lightweight narrative structured sentence model.

The structured sentence model works as follows. It takes the generated events from the previous layer as input, and chooses a pre-defined sentence template from a candidate list for this particular event. This sentence template comes with a list of *sensitive fields* that wait to be filled, which are represented by globally unique symbols. In other words, the sensitive fields are essential fields of information whose real values are decided in the diary generation step. As an example, we take a look at the following sentence:

> Structured Sentence Template: *You and # know each other so well that you made % calls and spent an average of % minutes on each call.*

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

Generated Sentence by Smart Diary: *You and **Anne** know each other so well that you made **30** calls and spent an average of **45** minutes on each call.*

B. Machine-Interpretable Entity-Relationship Model

Besides the structured sentence model used to generate output directly for the user, we also integrate a second, machine-interpretable model. In our design, we adopt one representative semantic model, i.e., the ER model [29], which has been widely used for the conceptual modeling of relationship data in databases. However, it has not been used to describe user's mobile activities. Therefore, adapting them for our needs represents a novel application.

In the ER model, it provides conceptual abstractions such as entities, attributes, and relationships. We can map these abstractions to natural languages as follows:

ER Structure	Natural Language Elements
Entity	Users, Nouns
Relationship	Transitive Verbs
Attribute for Entity	Adjectives, Intransitive Verbs
Attribute for Relationship	Adverbs

As an example showing how this process works, suppose we want to represent a simple user activity: the user watched a movie last Sunday evening. Such a description is straightforward in natural language, and can also be represented easily with the ER model as shown in Figure 6.

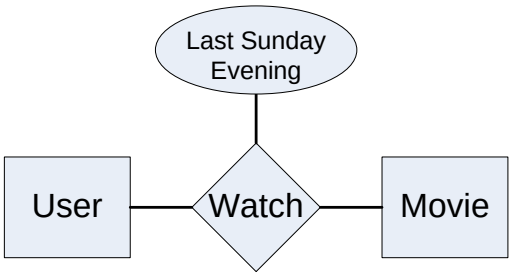
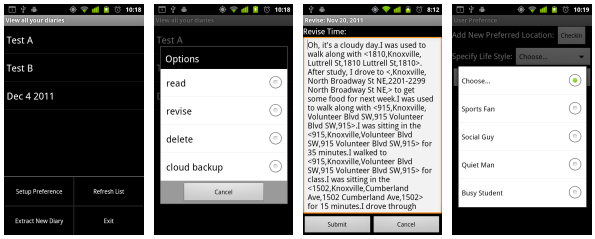


Fig. 6. Example ER Diagram with User Watching Movies

Note that in this ER model, we consider that there are two entities, the user, and the movie, both of which are nouns. The transitive verb "watch" is mapping to a relationship in this ER model. The attribute, "last Sunday evening", comes from the adverb on the description of the user activity.

More generally, in our application domain, the generation process for the ER models is as follows. First, the entity is a noun, such as the user. When a user performs an action, the action will either be translated into a relationship or an attribute. For example, if the action is transitive, such as "the user calls a friend", this will be translated into a relationship; on the other hand, if the action is intransitive, such as "the user goes shopping", it will be translated into attributes.

A challenge when we apply the ER model is the original model was developed to describe static relations in databases. In our target domain, however, the generated model is greatly affected by time, i.e., it should be evolving over time. That is, we need to add a time dimension to the solution, so that the ER model generated will change over the timeline. Therefore, we consider the ER model we use to be a time-enhanced. Specifically, whenever we derive higher level user actions from the mining process, we also generate the corresponding ER model representations in XML formats. Such formats are not intended to be directly read by users, but provide a machine interpretable method. In the following, we give an example of the user's shopping activity represented in XML:



(a) (b) (c) (d)

Fig. 7. The user interfaces of Smart Diary: (a) the diary list; (b) diary options; (c) diary view/refinement; (d) user preference.

```
<entity>
   <user> Joe </user>
   <activity> shopping </activity>
   <date> 12/7/2013 </date>
   <wdate> weekend </wdate>
</entity>
```

C. Sentence Generating Process

Smart diary does not require that the diary generation has to be done on a daily process. Instead, we provide a user interface as shown in Fig. 7 to control the whole procedure. Additionally, Smart Diary provides the capability of time span selection, so that the user can decide to keep records of the most interesting events within a day, a week or a month in one piece of a diary.

Recall that during the generation, the personalization layer chooses those events that are most related to the user's personal life based on the ranking scores. When an event has multiple structured sentence templates to choose, a *random selection* is applied to pick one in equal chance. This selection approach (Algorithm 2) is used because we have no *priori* knowledge of a user's language preference. In addition to the *random selection* process, we leave a choice for the user to promote their favoriate sentence templates. The diary generation component maintains a preference table which records the user's preference on each sentence. The sentence templates in the table starts with equal preference, and the preference will be increased when the user promotes this template. At the diary generation process, the probability of selecting a sentence is proportional to the preference in the table.

D. User Refinement

After the diaries are generated, we also provide an optional user refinement layer so that a user may add personal flavors to their diaries by providing additional feedbacks. To this end, Smart Diary integrates user inputs in the following manner. Once the diary generation finishes, a user may modify a sentence, which may lead to a new structured sentence

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

Algorithm 2 Diary Generation Process

Input: The personalized event sequence: *events*.
Output: A full diary of the sequence of events: *diary*.

```
Initial diary ← ∅.
for all e in the event sequence events do
    fetch structured sentences s by e.id.
    if s.size > 1 then
        select sentence by random in s.
    else
        sentence ← s.front.
    fill e.fields into sentence.symbols.
    append sentence to diary.
return diary
```

template to be stored in the dictionary; or the user may change certain sensitive information directly, which, however, does not generate new structured sentence templates. If a user particularly prefers a certain sentence, they can vote on this structured sentence template to increase the probability that this sentence will be selected in the future.

VIII. System Performance

The performance evaluation consists of three parts: experiment setup (VIII-A), diary quality (VIII-C), and application performance (VIII-D).

A. Experiment Setup

The prototype of the Smart Diary client app implements 12 events' mining components (MCs) and 12 structured sentence templates in the deployed application. Among the 12 events, 6 belong to the entertainment activity category, 3 belong to the social activity category, and 3 belong to the health condition category. The experiment consists of two parts: first we evaluate the performance of our motion recognition component, then we evaluate the end-to-end diary generation.

In order to perform these two experiments, we recruited five graduate students in our department, and each of them takes the Nexus S smartphones all day, on which we have deployed the Smart Diary client app. The students' normal activities on campus include taking classes, meeting and walking around the campus; and they usually drive back-and-forth between their home and campus. Finally, they also go to different places they are interested.

The first experiment asks the students to execute each of the motions *sit*, *drive*, *walk* and *run* for an hour in the campus area, then we compare the recognition results with the ground truth to validate the component's performance. The results are shown in Section VIII-B.

In the second experiment, we ask the five students to take the phones every day for a month. They record their daily events as the ground truth. During the experiment, the Smart Diary application can capture their daily life activities completely and generate a diary for them everyday. Instead of using the user's subjective opinions on the generated diary's quality, which may be affected by uncontrolled biases, we use two objective score systems, one named the Bilingual Evaluation Understudy (BLEU) [27], [7], and the other named Metric for Evaluation of Translation with Explicit Ordering (METEOR) [4], [18], which are commonly used to measure the language quality in natural language processing field. The BLEU score focus on the number of words correctly used by the diary generation process when compare to a human written diary under the same event. The percentage of correct words is the score. On the other hand, METEOR score is more complicated by considering sentence alignment between the generated one and the human written one, and it weights recall higher than precision of a generated sentence. Therefore, it is safer to use both score together in order to provide the objective diary quality evaluation.

Additionally, the five students write their own diary for the events happened in a day, and we compare their diaries with the ones from the Smart Diary client app using BLEU and METEOR scores. The results are illustrated in Section VIII-C, and a sample generated diary is provided as an example of the quality of the generated diary texts as well.

TABLE VII
Motion Type Recognition using decision tree

		Recognized Motion			
		Sit	Drive	Walk	Run
True Motion	Sit	**96.9%**	3.01%	0.09%	0%
	Drive	4.3%	**93.67%**	2.03%	0%
	Walk	2.6%	4.7%	**93.81%**	0.23%
	Run	0%	0%	1.25%	**98.75%**

TABLE VIII
Motion Type Recognition using Naive Bayes algorithm

		Recognized Motion			
		Sit	Drive	Walk	Run
True Motion	Sit	**94.91%**	4.98%	0.12%	0%
	Drive	7.5%	**90.78%**	1.72%	0%
	Walk	1.26%	5.85%	**91.06%**	1.83%
	Run	0%	0%	6.87%	**93.13%**

B. Motion Type Recognition

Several components in the MCs are related to the purpose of motion type recognition. We use both accelerometer and gyroscope to recognize the motion type of a user, and we test two approaches to classify the motion type from the sensor readings. One is the decision tree approach and the other is the Naive Bayes approach. The motion types in the system prototype are *sit*, *walk*, *drive* and *run*. The confusion matrices are shown in Table VII and VIII, respectively.

According to the results from the above two tables, we conclude that it is safe to use the decision tree approach as the major mechanism for motion type recognition. Thereafter, we integrate this technique into the users' smartphones and the recognition results from the user's phone are shown in Fig. 8. As illustrated, for all four types of motions in the system, we can reach more than 90% precision and recall, which provides a high belief for event inference.

In comparison, we also investigate the activity recognition methods in state-of-art. To our best knowledge, [5] and [17] covers adequate alternative approaches, including J48, logistic regression and multiple-layer perceptron, to recognize the

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

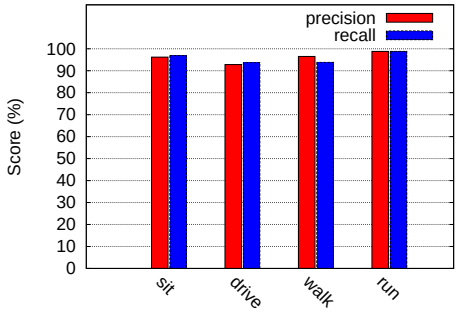


Fig. 8. Precision/Recall of motion recognition

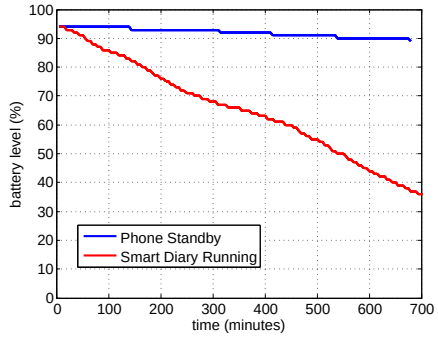


Fig. 9. Battery level

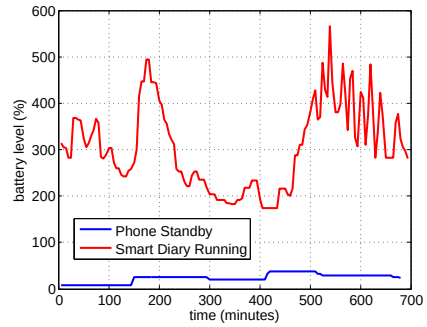


Fig. 10. Dynamic energy

TABLE IX
ACTIVITY RECOGNITION COMPARSON WITH STATE-OF-ART

Motion Type	Our best result	Best in [5]	Best in [17]
Sit	96.90%	94.78%	95.70%
Drive	93.67%	–	–
Walk	93.81%	89.71%	93.60%
Run	98.75%	87.68%	98.30%

activity. Table IX shows a comparison results with their approaches. Although [5] and [17] does not cover "drive" activity, we can see the advantages of our methods in recognizing the other activities.

C. Diary Quality

In this study, we require the users to log their major events that happened during the day, so that we can obtain ground truth for our evaluation purpose. For each of those events, Smart Diary generates the diaries sentence by sentence. We then evaluate the output of the Smart Diary app using the BLEU precision score and the METEOR score.

The results are shown in Table X by category. As we can see, all of the sentences in three categories show reasonable results in this small-scale deployment because the structured sentence templates are derived from natural sentences in a supervised way. Note that the METEOR score implies that the entertainment category obtains the lowest score while in BLEU it is the highest. This phenomenon is normal because the two objective score systems have different focuses, and the small-scale deployment only have 12 structured sentences whose performances may vary a lot based on different criteria. In general, all three categories in total have an average score at 0.5, which, according to the suggested use of BLEU and METEOR scores, reflects acceptable performance for the target experiment.

TABLE X
THE OBJECTIVE LANGUAGE QUALITY SCORES.

Category	BLEU score	METEOR score
Entertainment Activity	0.6067	0.3215
Health Conditions	0.4455	0.4578
Social Activity	0.6055	0.4297

Beyond the BLEU and METEOR scores, a sample diary for one of our volunteers is provided in the following:

> *Your life seems to follow a repeating pattern. You frequently drive from* ***your home (1544 Holman Dr. Knoxville, TN 37909)*** *to* ***your office (Min Kao University of Tennessee, 1520 Middle Dr, Knoxville, TN 37916)****. For the past* **four** *weeks, you were living a very busy life, since you were sitting too much at front of your office desk. Wow, finally you spent some time in* ***West Town Mall (7700 Kingston Pike Knoxville, TN 37919)*** *for shopping. You seem to use a lot of apps in a day. Among them, your favorite app is* ***com.yelp.android****. Your top contact is* ***"little-mushroom"*** *because you guys made* **37** *calls in total for a duration of* **1,783** *minutes for the past* **four** *weeks. You have no updates on your* ***Facebook connections****. You have not been to* ***the gym*** *for a long time. Any new relationships?*

As is seen, the bold font is the information generated from the event while the other part belongs to the structured sentences. The diary covers the user's daily routine, life pace, entertainment and social activity. Note that the generated diaries are by no means complete, because it is constrained by the mining components and the structured sentence templates used in the prototype.

D. Application Performance

Since Smart Diary is a mobile phone application, the energy consumption is critically important. To evaluate the energy efficiency of Smart Diary, Fig. 9 and 10 display the energy consumption in the Nexus S smartphone. As is shown in Fig. 9, the battery level curve indicates that the battery life stays more than 11 hours before dropping down to 30%, which is reasonably good given that many people have the habit of recharging during the night in their daily lives. On the other hand, in Fig. 10, the dynamic energy consumption shows that the power consumption of the smartphone when Smart Diary is deployed is around 250 mW. Comparing to the average usage range of $[200, 1100]$ mW [9], it is reasonable to say that Smart Diary is lightweight and not energy-hungry even when it is used for continuous sensing.

On the other hand, Table XI shows the Smart Diary's runtime fingerprints (by turning on all the services in Smart Diary) along with other commonly used applications in Android system. From this table, we can see the runtime memory is equivalent to Dropbox while the CPU consumption is close to Google Services. Therefore, it is safe to conclude that Smart Diary app performance is competitive to other widely used app in Android.

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

TABLE XI
THE ANDROID RUNTIME FOOTPRINT OF UNO@HOME AND UNO@SENSE

Application name	Runtime memory	App size	CPU usage
Gmail	29 MB	9,850 KB	< 1%
Google Docs	29 MB	4,660 KB	< 1%
Dropbox	24 MB	3,550 KB	< 1%
Smart Diary	24 MB	3,600 KB	3%
Google Services	23 MB	2,820 KB	2%
Android Keyboard	21 MB	930 KB	< 1%
Google Maps	21 MB	292 KB	< 1%
Google +	20 MB	23,500 KB	< 1%
Google Search	16 MB	44 KB	< 1%

IX. CONCLUSIONS AND FUTURE WORK

In this paper, we discussed the motivation, design, implementation and evaluation of Smart Diary, an integrated sensing framework that integrates mobile context analysis, human activity inference, and natural language processing on the smartphone platform. Our preliminary evaluation results on the working prototype are exciting. We have demonstrated that the system works as desired in that it may infer people's daily life faithfully and generate dairies automatically and flexibly according to users' preferences. In the future, we hope that we can further expand the types of MCs by adding more inference rules, develop more possible ways to display people's daily activities, and make the sensing framework to be publicly available.

X. ACKNOWLEDGEMENT

The work reported in this paper was supported in part by the National Science Foundation grant CNS-0953238, CNS-1017156, CNS-1117384, and CNS-1239478.

REFERENCES

[1] Narrative science. http://narrativescience.com.
[2] Nfc-based hacking report. http://www.networkworld.com/news/2012/072612-researcher-wows-black-hat-with-261162.html?hpg1=bn.
[3] J. An, X. Gui, W. Zhang, J. Jiang, and J. Yang. Research on social relations cognitive model of mobile nodes in internet of things. *Journal of Network and Computer Applications*, 36(2):799–810, 2013.
[4] S. Banerjee and A. Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proc. of ACL Workshop*, pages 65–72, 2005.
[5] L. Bao and S. S. Intille. Activity recognition from user-annotated acceleration data. *Pervasive Computing*, pages 1–17, 2004.
[6] J. Brandt, N. Weiss, and S. Klemmer. Lowering the burden for diary studies under mobile conditions. In *Proc. of CHI*, pages 2303–2308. ACM, 2007.
[7] C. C. Burch, M. Osborne, and P. Koehn. Re-evaluating the role of bleu in machine translation research. In *Proc. of EACL*, volume 2006, pages 249–256, 2006.
[8] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, R. A. Peterson, H. Lu, X. Zheng, M. Musolesi, K. Fodor, and G.-S. Ahn. The rise of people-centric sensing. *IEEE Internet Computing*, 12(4):12–21, July 2008.
[9] A. Carroll and G. Heiser. An analysis of power consumption in a smartphone. In *Proc. of USENIX ATC*, pages 21–21. USENIX Association, 2010.
[10] S. Consolvo, D. McDonald, T. Toscos, M. Chen, J. Froehlich, B. Harrison, P. Klasnja, A. LaMarca, L. LeGrand, R. Libby, et al. Activity sensing in the wild: a field trial of ubifit garden. In *Proc. of CHI*, pages 1797–1806. ACM, 2008.
[11] M. Czerwinski, E. Horvitz, and S. Wilhite. A diary study of task switching and interruptions. In *Proc. of CHI*, pages 175–182. ACM, 2004.
[12] N. Eagle and A. S. Pentland. Reality mining: sensing complex social systems. *Personal Ubiquitous Computing*, 10(4):255–268, March 2006.
[13] C. Emmanouilidis, R.-A. Koutsiamanis, and A. Tasidou. Mobile guides: taxonomy of architectures, context awareness, technologies and applications. *Journal of Network and Computer Applications*, 2012.
[14] D. Gavalas, C. Konstantopoulos, K. Mastakas, and G. Pantziou. Mobile recommender systems in tourism. *Journal of Network and Computer Applications*, 2013.
[15] R. Guo, S. Li, L. He, W. Gao, H. Qi, and G. Owens. Pervasive and unobtrusive emotion sensing for human mental health. In *Proceedings of the 7th International Conference on Pervasive Computing Technologies for Healthcare*, pages 436–439. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2013.
[16] S. Hammerl and T. H. adn Helge Ritter. Towards a semi-automatic personal digital diary. In *Proceedings of the PETRA Conference*, June 2012.
[17] J. R. Kwapisz, G. M. Weiss, and S. A. Moore. Activity recognition using cell phone accelerometers. In *Proc. of KDD*, July 2010.
[18] A. Lavie, K. Sagae, and S. Jayaraman. The significance of recall in automatic metrics for mt evaluation. *Machine Translation: From Real Users to Research*, pages 134–143, 2004.
[19] S. Lee and K. Mase. Activity and location recognition using wearable sensors. *Pervasive Computing, IEEE*, pages 24–32, 2002.
[20] R. LiKamWa, Y. Liu, N. Lane, and L. Zhong. Can your smartphone infer your mood? *Proc. of PhoneSense workshop*, 2011.
[21] R. T. Llamas and W. Stofega. Worldwide smartphone 2013 c2017 forecast and analysis. *IDC Market Analysis*, March 2013.
[22] H. Lu, W. Pan, N. D. Lane, T. Choudhury, and A. T. Campbell. Soundsense: Scalable sound sensing for people-centric applications on mobile phones. In *Proc. of MobiSys*, pages 165–178, June 2009.
[23] E. Miluzzo, C. T. Cornelius, A. Ramaswamy, T. Choudhury, Z. Liu, and A. T. Campbell. Darwin phones: the evolution of sensing and inference on mobile phones. In *Proc. of MobiSys*, pages 5–20, 2010.
[24] E. Miluzzo, N. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. Eisenman, X. Zheng, and A. Campbell. Sensing meets mobile social networks: the design, implementation and evaluation of the cenceme application. In *Proc. of SenSys*, pages 337–350, 2008.
[25] E. Miluzzo, N. D. Lane, S. B. Eisenman, and A. T. Campbell. Cenceme - injecting sensing presence into social networking applications. In *Proc. of EuroSSC*, pages 1–28, October 2007.
[26] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, and P. Boda. Peir, the personal environmental impact report, as a platform for participatory sensing systems research. In *Proc. of MobiSys*, pages 55–68, June 2009.
[27] K. Papineni, S. Roukos, T. Ward, and W. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proc. of ACL*, pages 311–318, 2002.
[28] C. Qin, X. Bao, R. Choudhury, and S. Nelakuditi. Tagsense: a smartphone-based approach to automatic image tagging. In *Proc. of MobiSys*, 2011.
[29] R. Ramakrishnan and J. Gehrke. *Database management systems*. Osborne/McGraw-Hill, 2000.
[30] S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, and M. Srivastava. Using mobile phones to determine transportation modes. *ACM Transactions on Sensor Networks (TOSN)*, 6(2):13, 2010.
[31] D. Scott. *Multivariate density estimation*, volume 139. Wiley Online Library, 1992.
[32] C. Shepard, A. Rahmati, C. Tossell, L. Zhong, and P. Kortum. Livelab: measuring wireless networks and smartphone users in the field. *ACM SIGMETRICS Performance Evaluation Review*, 38(3):15–20, 2011.
[33] H. Shin, Y. Chon, K. Park, and H. Cha. Findingmimo: tracing a missing mobile phone using daily observations. In *Proc. of MobiSys*, pages 29–42, 2011.
[34] R. Sigal, G. Kenny, D. Wasserman, and C. Castaneda-Sceppa. Physical activity/exercise and type 2 diabetes. *Diabetes care*, 27(10):2518–2539, 2004.
[35] C. Tossell, P. Kortum, A. Rahmati, C. Shepard, and L. Zhong. Characterizing web use on smartphones. In *Proc. of CHI*, pages 2769–2778. ACM, 2012.
[36] Y. Wang, J. Lin, M. Annavaram, Q. A. Jacobson, J. Hong, B. Krishnamachari, and N. Sadeh. A framework of energy efficient mobile sensing for automatic user state recognition. In *Proc. of MobiSys*, pages 179–192, 2009.
[37] K. Whitehouse, F. Zhao, and J. Liu. Semantic streams: A framework for composable semantic interpretation of sensor data. *Wireless Sensor Networks*, pages 5–20, 2006.

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2014.2331970, IEEE Sensors Journal

[38] K.-L. A. Yau, P. Komisarczuk, and P. D. Teal. Reinforcement learning for context awareness and intelligence in wireless networks: Review, new features and open issues. *Journal of Network and Computer Applications*, 35(1):253–267, 2012.

1530-437X (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.