Personalized Battery Lifetime Prediction for Mobile Devices based on Usage Patterns

Joon-Myung Kang[*]
Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada
joonmyung.kang@utoronto.ca

Sin-seok Seo
Department of Computer Science and Engineering, Pohang University of Science and Technology (POSTECH), Pohang, Korea
sesise@postech.ac.kr

James Won-Ki Hong
Division of IT Convergence and Engineering, Pohang University of Science and Technology (POSTECH), Pohang, Korea
jwkhong@postech.ac.kr

Abstract

Nowadays mobile devices are used for various applications such as making voice/video calls, browsing the Internet, listening to music etc. The average battery consumption of each of these activities and the length of time a user spends on each one determines the battery lifetime of a mobile device. Previous methods have provided predictions of battery lifetime using a static battery consumption rate that does not consider user characteristics. This paper proposes an approach to predict a mobile device's available battery lifetime based on usage patterns. Because every user has a different pattern of voice calls, data communication, and video call usage, we can use such usage patterns for personalized prediction of battery lifetime. Firstly, we define one or more states that affect battery consumption. Then, we record time-series log data related to battery consumption and the use time of each state. We calculate the average battery consumption rate for each state and determine the usage pattern based on the time-series data. Finally, we predict the available battery time based on the average battery consumption rate for each state and the usage pattern. We also present the experimental trials used to validate our approach in the real world.

I. INTRODUCTION

As the growth of mobile networks and services has accelerated, battery powered mobile devices (e.g., mobile handsets or PDAs) have emerged as important tools in our lives [1]. Mobile devices are used for various applications such as making voice/ video calls, browsing the Internet, playing games etc. The battery consumption of these activities affects the battery lifetime of mobile devices. A mobile device that has a short battery lifetime can cause much inconvenience for users and reduce the device's usefulness [2, 3]. Mechanisms to provide long and stable battery life are therefore required. One of the methods to guarantee a long battery life is to minimize battery consumption by reducing unnecessary battery usage.

The key to minimizing battery consumption is accurately predicting the battery consumption of the various operating

states of mobile devices. Much work has focused on hardware or operating system (OS) level predictions [4, 5], model-based battery life predictions [6], and battery consumption by a CPU when executing a particular instruction [7-9]; furthermore, battery lifetime has usually been predicted using a static battery consumption rate, which the manufacturers of batteries and mobile devices have determined via experiments. The consumption rate has also been predicted by an arithmetic mean of the consumed power. However, previous methods have not been able to predict battery lifetime in terms of user characteristics, because such methods have not considered usage patterns. That is, every user has a different mix of voice call, data communication, and video call usage. If each user has a consistent usage pattern [10], we can predict battery lifetime on an individual basis.

In this paper, we propose a method for predicting available battery lifetime in mobile devices based on the usage patterns of individual users. We define possible mobile device states based on the device's basic functions, e.g., voice call and data communication. We develop daily (i.e., morning/afternoon/evening/night) and weekly (i.e., weekday/weekend) user profiles that reflect each user's life patterns, i.e., working, resting, and sleeping patterns for both weekdays and weekends. We measure the battery consumption and time spent in each operational state. Then, we develop usage patterns based on various modes (e.g., sleeping, resting, and working). In our previous work [11], we did not consider user life patterns for predicting battery lifetime. Here we present a method for predicting battery lifetime using usage patterns generated by the probability that a user will spend a certain time in each operational state for each mode. We then present the numerical results of a simple use case, which accurately predicts battery lifetime. We also present an experimental trial used to validate our proposed method in the real world.

The main contributions of the paper are as follows: 1) Because the proposed method presents real-time predicted battery lifetime in terms of an on-line analysis based on usage patterns rather than an off-line analysis, this method is more useful to users than previous ones. 2) The proposed method enables effective battery usage by reducing useless battery consumption and detects abnormal battery usage by comparing operating times between normal and abnormal states.

The organization of this paper is as follows. Section II describes previous work for predicting battery lifetime and mobile applications for battery management. Section III defines the problem and describes our approach. Section IV presents our experimental trials with real log data from smartphones. Finally, we draw conclusions and suggest future work in Section V.

II. RELATED WORK

Many mobile devices provide battery lifetime prediction via hardware and OS support as an integral component of power management systems. Advanced Power Management (APM) [4] has been widely used in power management for computer systems. The Advanced Configuration and Power Interface (ACPI) [5] recently proposed by Intel, Microsoft and Toshiba is an advanced version of APM. ACPI defines power configuration and management functions between a Basic Input Output System (BIOS) and the OS. In particular, it uses the remaining battery capacity and present rate of battery drain to predict the remaining battery life. Currently, ACPI is used in power management for desktops and laptops. The SBS Forum [12] has proposed a smart battery system, which is a specification for determining accurate battery readings; it allows the OS to perform power management operations based on estimated run times. This system predicts battery lifetime given a user-specified rate or rolling average over a fixed interval. Such simple approaches consider only a very short discharge history and are therefore inaccurate.

Another area of related research is model-based battery life prediction. Given the discharge profile of the entire lifetime of the battery, a well-designed battery model can provide a highly accurate battery life prediction. One such accurate model, described in [6], uses the low-level electrochemical phenomenon of battery discharge. This model is commonly used as a simulator to verify other battery models.

Moreover, Tiwari et al. [7, 9] and Saputra et al. [8] have studied power consumption characteristics at the processor instruction level. Combining these characteristics with battery models, one can estimate battery consumption. Model-based methods have limitations that make them unsuitable for online, dynamic and adaptive battery life prediction, such as situations where there is a need for a large number of parameters and there is a high computation cost (especially for simulation models). Our proposed method focuses on predicting battery lifetime based on the empirical usage patterns of each user, and no model-driven approach is needed.

Krintz et al. [3] proposed application-level prediction of battery dissipation by investigating the degree to which battery dissipation, as induced by program execution, can be measured by application-level software tools and predicted by a compiler and runtime system. They presented a black-box approach in which they attempted to observe the dissipation characteristics at the application-level. However, they did not consider the effect of usage patterns at the individual user level. Thus, while this method focuses on applications, our method focuses on a user's behavioral operating states and reflects the user's characteristics, taking a more user-centric approach.

Recently, Zhang et al. [13] proposed PowerBooter, an automated power model construction technique that uses built-in battery voltage sensors and knowledge of battery discharge behavior to monitor power consumption while explicitly controlling the power management and activity states of individual components. They also presented PowerTutor, which is an Android application for power management and activity state introspection based on a tool that uses the model generated by PowerBooter for online power estimation. This proposed method is highly related to our work but we mainly focused on using user characteristics for power estimation.

III. PROPOSED METHOD

In this section, we present our method for predicting battery lifetime based on usage patterns. We firstly present a problem

definition. We then describe our proposed method for solving the problem.

A. Problem Definition

In order to define the problem in this paper, we also assume that a mobile device has n possible states. Firstly, we present the following symbolic notations and formulae:

$$\vec{B} = (B_1, ..., B_n), \tag{1}$$

$$\vec{p} = (p_1, ..., p_n), \tag{2}$$

where B_i is the average battery consumption rate of the ith state, and p_i is the use time rate of the ith state, which is determined to satisfy $\sum_{i=1}^{n} p_i = 1$. Because each user has a different vector, $\vec{p}$, according to the usage pattern, this vector is defined for the symbol that represents their usage pattern.

We can formulate our problem as follows.

Problem Formulation: Determine the average battery consumption rate vector $\vec{B}$ and the usage pattern vector $\vec{p}$ to minimize the Mean Square Error (MSE) of a predictor

$$MSE(\hat{\theta}) = E[(\hat{\theta} - \theta)^2], \tag{3}$$

where $\hat{\theta}$ is an estimator from our prediction method and θ is the true values measured from users. We predict the available battery lifetime based on usage patterns using the following formula.

$$T = \frac{V}{\sum_{i=1}^{n} p_i \cdot B_i}, \tag{4}$$

where T is the available time predicted by usage patterns, and V is the total battery capacity. Because each user has a different $\vec{p}$, every user's battery has a different remaining lifetime. Previous approaches cannot represent a user's personal characteristics, because they do not consider usage patterns.

B. Solution

In this section, we describe a solution to the problem defined in Section III-A. The hypothesis we present in this paper is that the most accurate prediction of a mobile device's battery lifetime is based on usage patterns and minimization of the prediction error. When a mobile device is turned on, a battery monitoring application begins collecting data on the operating time and battery consumption of each state based on a timestamp. This time-series data is used to measure the average battery consumption rate ($\vec{B}$) and usage patterns ($\vec{p}$). Finally, the application provides a prediction of battery lifetime based on these two vectors. Firstly, we collect the time-series data for each state. To this end, we append the time and battery level to a log file after entering and exiting the state. The symbol for the log data is the vector $\vec{D}_j = (D_j^1, D_j^2, D_j^3, D_j^4)$, where $\vec{D}_j$ is a vector of the jth log data, D_j^i is the time-series data of the jth log data, D^1 is a state of the mobile device, D^2 is battery consumption, D^3 is use time, and D^4 is a timestamp. Next, we calculate the average battery consumption rate and the usage pattern using time-series log data. As defined in Section IIIA, we assume that a mobile device has n possible states and the log file has m entries. We denote the total battery consumption by the vector $\vec{b} = (b_1, ..., b_n)$. The value of this vector is the total battery consumption of the ith state via the sum of the second element of each log entry for which the state is i. We also denote the use time by the vector $\vec{t} = (t_1, ..., t_n)$. The value of this vector is the use time of the ith state via the sum of the third element of each log entry for which the state is i. Thus,

$$b_i = \sum_{j=1}^{m} D_j^2, \tag{5}$$

$$t_i = \sum_{j=1}^{m} D_j^3, \tag{6}$$

where $D_j^1 = i$. We define the average battery consumption rate and usage pattern for each user using Equations (5) and (6).

$$B_i = \frac{b_i}{t_i}, \tag{7}$$

$$p_i = \frac{t_i}{\sum_{i=1}^{n} t_i}, \tag{8}$$

where B_i is the average battery consumption rate of the ith state, and p_i is the use time rate of the ith state, which is determined to satisfy $\sum_{i=1}^{n} p_i = 1$. This usage pattern represents not only the past usage pattern, but also the probability of spending the same amount of time in each state in the future [10].

The graph of Fig. 1 plots the time-series data related to the battery power and the time spent in each state. The graph plots the time-series data collected by the data collector in terms of time and remaining battery capacity. In particular, the dotted lines indicate the power consumption and use time of the battery when the user is only using voice communication with the liquid crystal display (LCD) in the "OFF" state. With an arithmetic mean of all the samples of the time-series data in each state, the average power consumption rate per state can be obtained by using Equation (9):

$$\alpha_i = \frac{B_1 + B_2 + B_3}{t_1 + t_2 + t_3}, \tag{9}$$

where α_i is the average power consumption rate (*mA/sec*) in the
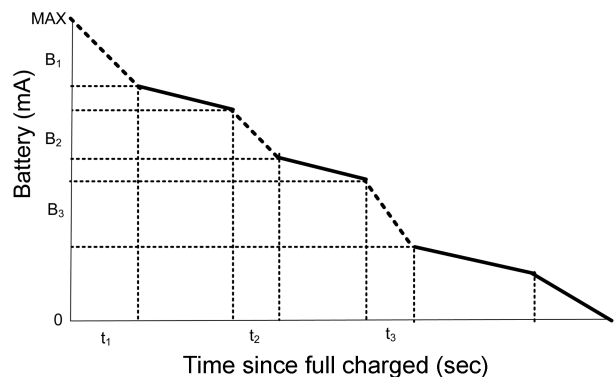

Fig. 1. Linear approximation of the duration of voice call. dotted line: voice call, solid line: waiting, B_n: nth battery consumption rate of nth voice call, t_n: nth duration of nth voice call.

state, B_j is the power consumption amount (mA) for the jth voice communication, and t_j is the call duration of the jth voice communication or the use time (sec). The usage pattern of each user can be determined as a numerical value based on their use time rate ($vecp_i$) in each state, because the rate is different for each user. There are numerous methods for calculating the use time rate per state. The simplest method is to divide the use time in each state by the sum of the use time in all states, based on the time-series data collected by the data collector. Another method to calculate the use time rate per state is to predict future data based on past time-series data, for example, the Autoregressive Model (AR) [14] or the Holt-Winters model [15]. Although the usage pattern is also based on past time-series data, weight factors can be adjusted so that recent data have more weight. In future work, we will apply these forecasting methods to calculate usage patterns. We have described how to determine the usage pattern vector $\vec{p}$ of each user by recording the time spent in each state. However, the life patterns of most users differ according to the day or week, e.g., working, resting and sleeping (a day) and weekday and weekend (a week). In addition, users have variable life patterns, e.g., vacations or business trips. We consider user life patterns for predicting battery lifetime. We divide each day and week into several modes based on a user's life patterns. If a mobile device has n possible states and a user defines l modes, we define the usage pattern vector with Equation (10).

$$\vec{P}_i = (P_i^1, ..., P_i^n), (i = 1, ..., l), \qquad (10)$$

where $\vec{P}_i$ is the usage pattern of the ith mode, and p_i^j is the use time rate of the jth state of the ith mode, which is determined to satisfy $\sum_{j=1}^{n} p_i^j = 1$. If we predict the battery lifetime of the ith mode, we use the vector $\vec{P}_i$.

Fig. 2 shows a block diagram of our system for predicting battery lifetime based on usage patterns. The mobile device includes a mobile operating system, state detector, data manager, battery consumption calculator, usage pattern analyzer, and lifetime predictor. In general, a battery's power consumption is affected by usage patterns that include the operational modes of voice communication, data communication, active LCD, video communication, short message service, running application programs, music listening and standby. Among these operational modes, those of voice communication, data communication, and active LCD have the largest influence on battery consumption. The state detector detects the operational state by collecting call status, WiFi and 3G status, and screen status from the mobile

Fig. 2. Block diagram of our prediction method.

operating system or middleware. It then provides this information to the data manager, which creates and stores real-time data on battery and use time in each state. The battery consumption calculator, based on the real-time data created by the data manager, calculates an average battery and use time rate for each state. The usage pattern analyzer determines a usage pattern distribution, which represents the time spent in each state. The lifetime predictor, based on the average power consumption rate and the usage pattern, predicts the available battery lifetime.

IV. EVALUATION

In this section, we evaluate our proposed method using real usage patterns from smartphones.

A. Overview

To evaluate the proposed method, we present one case study for which there were good prediction results. Firstly, we assume that a mobile device has n possible states and r users are using it. We define two symbols that represent the usage patterns of each user.

$$\vec{p}^k = (p_1^k, ..., p_n^k), \qquad (11)$$

$$\vec{U} = (U_1, ..., U_r), \qquad (12)$$

$$\vec{U}_k = \vec{p}^k \cdot \vec{B}^T, \qquad (13)$$

where $\vec{p}^k$ is the usage pattern of the kth user, p_i^k is the fraction of operating time that the kth user spent in the ith state, and R_k is the average battery consumption rate of the kth user considering the usage pattern. In other words, p_i^k represents the probability that the kth user will spend a certain amount of time in the ith state, because we assume that the usage pattern will be consistent with the user's history. Table 1 lists a total of eight states that are defined under the three operational modes of active LCD, voice communication, and data communication.

In Table 1, "ON" indicates an operational mode of the mobile device, and "OFF" indicates a non-operational mode of the mobile device. State S0 occurs when a user is using voice communication (e.g., calling) and data communication at the same time with the LCD in the "ON" state. This is not the current state but a potential future state. In this state, the LCD, voice communication and data communication all affect battery consumption. State S1 occurs when the user is only using voice communication with the LCD in the "ON" state. State S1 is activated from the starting point of the voice communication. In this state, both the LCD and voice communication affect power consumption. State S2 occurs when the user is only using data communication with the LCD in the "ON" state. State S2 is activated from the starting point of data communication. In this state, both the LCD and data communication affect power consumption. State S3 occurs when the user is not using either voice communication or data communication. Only the LCD is in the "ON" state. This corresponds to the standby state when a call is incoming or when the user slides up or flips up the mobile device. In this state, only the LCD affects power
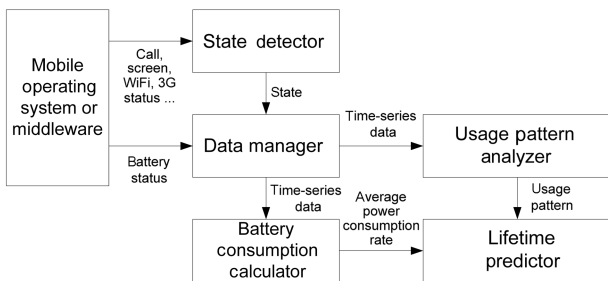
Table 1. Possible states of a mobile device

	S0	S1	S2	S3	S4	S5	S6	S7
LCD	ON	ON	ON	ON	OFF	OFF	OFF	OFF
VOICE	ON	ON	OFF	OFF	ON	ON	OFF	OFF
DATA	ON	OFF	ON	OFF	ON	OFF	ON	OFF
STATE	-	VOICE (SS1)	DATA (SS2)	WAITING (SS0)	-	VOICE (SS1)	-	WAITING (SS0)

LCD: liquid crystal display.

consumption. State S4 occurs when the user is using voice communication and data communication at the same time with the LCD in the "OFF" state. This is not the current state but a potential future state. In this state, both voice communication and data communication affect power consumption. State S5 occurs when the user is only using voice communication with the LCD in the "OFF" state. This occurs after the user has started voice communication and a certain amount of time has elapsed. In this state, voice communication affects power consumption. State S6 occurs when the user is only using data communication with the LCD in the "OFF" state. This is not the current state but a potential future state.. In this state, data communication affects power consumption. State S7 occurs when the user is not using either voice communication or data communication with the LCD in the "OFF" state. This corresponds to the standby state when there is no incoming call, when the mobile phone has been slid down or flipped down, or when the mobile phone has simply timed out. In this state, periodic connection with a base station affects power consumption. The problem can be simplified by reducing the number of possible states to minimize the complexity. The bottom row of Table 1 shows the simplified possible states of a mobile device. We merge states S3 and S7 to form state SS0. In this state, a user is not using either voice or data communication and periodic connection with a base station and waiting for calls affect power

consumption. We also merge states S1 and S5 to form state SS1. In this state, a user is only using voice communication and this affects power consumption. Finally, we define the state S2 as state SS2. In this state, data communication affects power consumption.

Fig. 3 shows a state transition diagram of a mobile device after simplifying the possible states. Initially, a mobile device starts from SS1 with the power on and waits to receive a call. When a user starts data communication, the current state is changed to SS3, and when they finish, the current state returns to SS1. When the user starts a voice call, the state is changed to SS2, and when they finish, the state returns to SS1. Based on these states and state transitions, we will show how we can predict the battery lifetime if the average battery consumption rate and usage patterns are known by monitoring data from real smartphones.

B. Experimental Trials

In this section, we present our experimental trials for predicting battery lifetime by collecting real data from an Android smart phone, which is one kind of mobile device. In this experiment, we show that users have different usage patterns and our method provides better prediction results than static prediction methods in terms of personalization. Currently, we are still col-
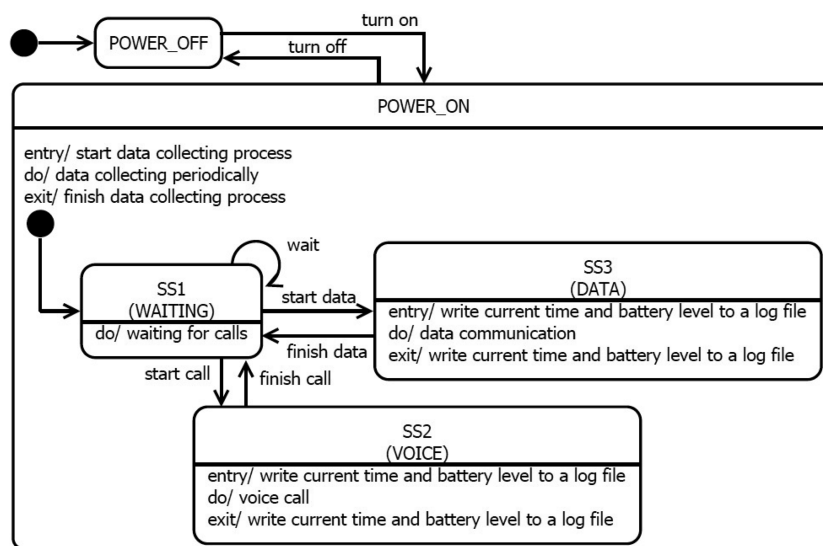

Fig. 3. Simplified state transition diagram of a mobile device.

lecting a large volume of battery usage data from various participants. We present our experimental setup and ongoing work.

Firstly, we implemented a data collecting application for analyzing user usage patterns. Our application was written in Java and runs on any Android-based platform. It periodically records the following information to a log file. Currently, it is scheduled to store the information every second.

- **Timestamp.** The timestamp for the date and time of the measurement.
- **Screen status.** Whether the screen (LCD) status is in the "ON" state or "OFF" state.
- **WiFi status.** Whether the WiFi status is in the "ON" state or "OFF" state.
- **3G status.** Whether the 3G status is in the "ON" state or "OFF" state.
- **Voice call status.** Whether the voice call status is in the "ON" state or "OFF" state.
- **Battery status.** The remaining battery power, temperature, voltage, and charging status as reported by the Android platform.

We also implemented a server system to collect log files from smart phones. The data collecting application sends the log file to the server system and then removes it from the smart phone. When we collect log files, we also collect a device specification and a device ID by using the phone number. We do not collect the phone number directly but convert it to a hash value, which we are not allowed to inversely convert due to privacy concerns. Firstly, we implemented the application on the project homepage (Battery Logger Project, http://dpnm.postech.ac.kr/battery) on February 17, 2011. To obtain users for the study, we made an announcement that we were seeking anonymous volunteers, using an online bulletin board for our campus, and we posted online advertisements to social network services such as Twitter or Facebook to obtain more users outside campus. Overall, we collected 40 users from March to April, 2011 (We are still collecting usage logs for our future work). For this paper, we used the logs from the 20 users with the longest total recorded time. The data from these 20 users is 1.5 Gbytes and 1.26 Mbytes per log on average.

Fig. 4 shows a comparison of individual users' time and battery consumption for five operational states: 1) waiting, 2) voice call, 3) data communication via WiFi, 4) data communication via 3G, and 5) other. From the comparison of time spent in each state (Fig. 4a), we can see that each user spent a different amount of time in each state. The state where a user spent more time will be used more in the future based on Zipf's law
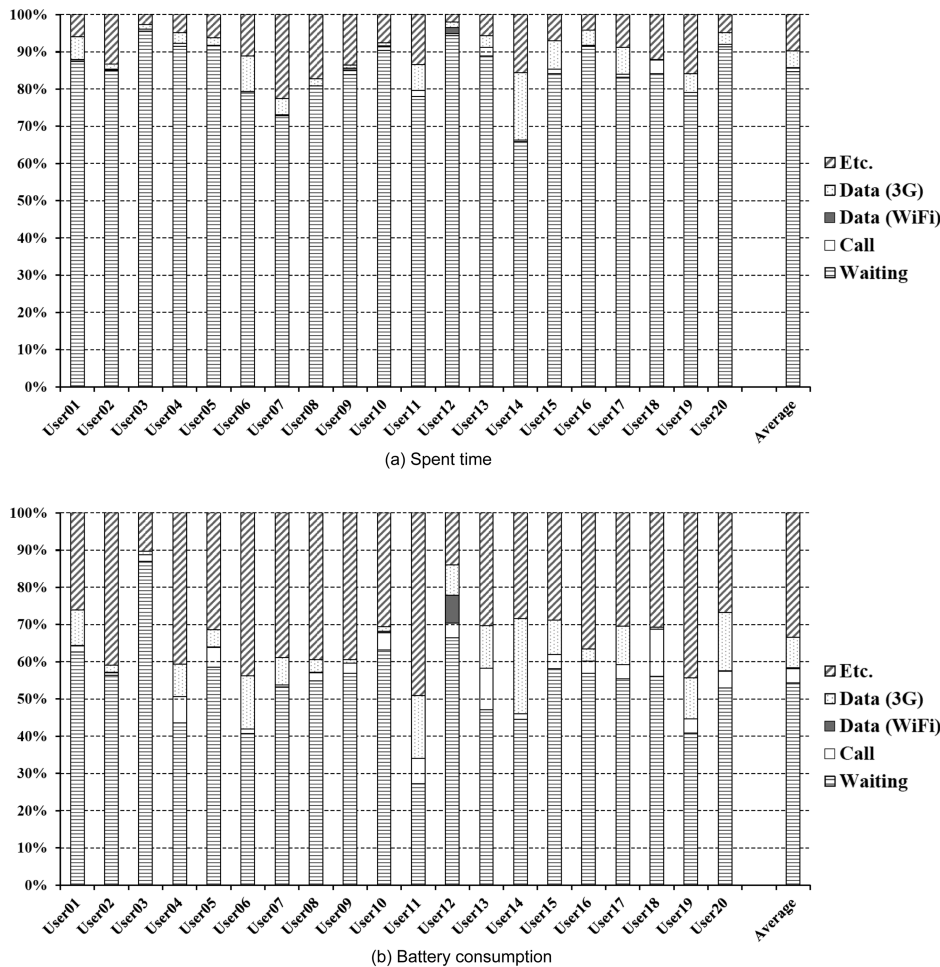

(a) Spent time

(b) Battery consumption

Fig. 4. A comparison of individual users' smartphone usage for five operational states in terms of (a) time spent and (b) battery consumption.

[10] and users have their own long-term usage patterns. Hence, we assume that this usage pattern is one of the user's characteristics. The ultimate goal of our study is to make an analytic model of user smartphone usage behavior. Fig. 4b shows the battery power consumed in each state for individual users and the differences between the time patterns shown in Fig. 4a. This chart shows that battery consumption differs in each state. As shown in Fig. 4a, most users spent more time on average in the waiting state, but every user had their own usage pattern.

The average battery consumption rates ($\vec{B}$) of each state based on the collected time-series data are as follows:

$$\vec{B} = (0.048, 0.252, 0.300, 0.132, 0.271), \quad (14)$$

If the usage pattern is given as Fig. 4a, we can define $\vec{p}$ for User12, User14, and User18 as follows:

$$\vec{p} = \begin{pmatrix} 0.94504 & 0.00527 & 0.01345 & 0.01622 & 0.02003 \\ 0.63739 & 0.00788 & 0 & 0.22672 & 0.12801 \\ 0.81914 & 0.04231 & 0.00001 & 0.00146 & 0.13708 \end{pmatrix}, \quad (15)$$

where the row and column values represent the usage pattern of each user and each state, respectively: column 1 is Waiting, column 2 is voice call, column 3 is WiFi, column 4 is 3G, and column 5 is other. For example, User12 spent more time in the WiFi state than other users. User14 spent more time in the 3G state, and User18 spent more time in the voice call state. To predict the battery lifetime of users with different usage patterns, we compare our proposed method with a method based on a static value derived from the arithmetic mean of all battery consumption rates.

The average battery consumption rate based on the arithmetic mean is about 0.2006. However, the following are the average battery consumption rates for each user by our proposed method.

$$\vec{U} = \vec{p} \cdot \vec{B}^T = \begin{pmatrix} 0.058294 \\ 0.097198 \\ 0.087325 \end{pmatrix}, \quad (16)$$

where the average battery consumption rates of User12, User14, and User18 are 0.058294, 0.097198, and 0.087325, respectively. From this case study, we can present different battery consumption rates for each user, based on their usage patterns. Furthermore, we can adaptively predict the battery lifetime of a mobile device using dynamic usage patterns.

V. CONCLUDING REMARKS

Predicting the battery lifetime of mobile devices is important to minimize battery consumption at the application level. In this paper, we have proposed a prediction model based on usage patterns, such as the battery consumption rate when making voice calls, using data communication, or waiting for calls. We have also shown the usefulness of our proposed method through a case study.

In future work, we hope to offer a more concrete validation of this method using log data that we continue to collect from end users. We will show usage pattern distributions, battery lifetime prediction results for each user, and comparisons with other prediction methods. We will also develop and deploy a mobile application to provide personalized battery management for mobile devices including battery lifetime prediction. Ultimately, we will contribute to personalized power management for mobile devices.

ACKNOWLEDGMENTS

REFERENCES

1. S. Rollins and C. Chang-Yit, "A battery-aware algorithm for supporting collaborative applications," *Collaborative Computing: Networking, Applications and Worksharing. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, E. Bertino and J. B. D. Joshi, Eds., Heidelberg, Germany: Springer Berlin, 2009, pp. 594-608.

2. L. Benini, A. Bogliolo, and G. De Micheli, "Dynamic power management of electronic systems," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, San Jose, CA, 1998, pp. 696-702.

3. C. Krintz, Y. Wen, and R. Wolski, "Application-level prediction of battery dissipation," *Proceedings of the International Symposium on Lower Power Electronics and Design*, Newport Beach, CA, 2004, pp. 224-229.

4. Intel Corporation and Microsoft Corporation, Advanced Power Management (APM) BIOS Interface Specification Revision 1.2, 1996.

5. Compaq Computer Corporation, Intel Corporation, Microsoft Corporation, Phoenix Technologies Ltd., and Toshiba Corporation, Advanced Configuration and Power Interface Specification, Revision 2.0b, 2002.

6. M. Doyle, T. Fuller, and J. Newman, "Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell," *Journal of the Electrochemical Society*, vol. 140, no. 6, pp. 1526-1533, 1993.

7. V. Tiwari, S. Malik, A. Wolfe, and M. T. C. Lee, "Instruction level power analysis and optimization of software," *Journal of VLSI Signal Processing Systems*, vol. 13, no. 2-3, pp. 223-238, 1996.

8. H. Saputra, M. Kandemir, N. Vijaykrishnan, M. J. Irwin, J. S. Hu, C. H. Hsu, and U. Kremer, "Energy-conscious compilation based on voltage scaling," *Proceedings of the Joint Conference on Languages, Compilers and Tools for Embedded Systems and Software and Compilers for Embedded Systems*, Berlin, Ger-

many, 2002, pp. 2-11.

9. V. Tiwari, S. Malik, and A. Wolfe, "Power analysis of embedded software: a first step towards software power minimization," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 2, no. 4, pp. 437-445, 1994.

10. G. K. Zipf, Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology, New York, NY: Hafner Publishing Co., 1965.

11. J. M. Kang, C. K. Park, S. S. Seo, M. J. Choi, and J. Hong, "User-centric prediction for battery lifetime of mobile devices," *Challenges for Next Generation Network Operations and Service Management. Lecture Notes in Computer Science Vol. 5297*, Y. Ma, D. Choi, and S. Ata, Eds., Heidelberg, Germany: Springer Berlin, 2008, pp. 531-534.

12. Benchmarq Microelectronics Inc., Duracell Inc., Energizer Power Systems, Intel Corporation, Linear Technology, Maxim Integrated Products, Mitsubishi Electric Corporation, National Semiconductor Corporation, and Toshiba Battery Co., Smart Battery Data Specification, Revision 1.1, 1998.

13. L. Zhang, B. Tiwana, R. P. Dick, Z. Qian, Z. M. Mao, Z. Wang, and L. Yang, "Accurate online power estimation and automatic battery behavior based power model generation for smartphones," *Proceedings of the 8th IEEE/ACM International Conference on Hardware/Software-Co-Design and System Synthesis*, Scottsdale, AZ, 2010, pp. 105-114.

14. R. G. Brown, Statistical Forecasting for Inventory Control, New York, NY: McGraw-Hill, 1959.

15. P. R. Winters, "Forecasting sales by exponentially weighted moving averages," *Management Science*, vol. 6, no. 3, pp. 324-342, 1960.

Joon-Myung Kang

Joon-Myung Kang (joonmyung.kang@utoronto.ca) received his B.Sc. and Ph.D. in Computer Science and Engineering from POSTECH, Pohang, Korea in 2005 and 2011, respectively. From 2000 to 2004, he worked at Alticast Corporation, as a software engineer. He was a Postdoctoral Felllow in the Dept. of Computer Science and Engineering at POSTECH, Pohang, Korea from March 2011 to September 2011. Currently, he is a Postdoctoral Fellow in the Dept. of Electrical and Computer Engineering at University of Toronto, Toronto, ON, Canada. His research interests include mobile device management, autonomic network management, personalized services, and future internet management.

Sin-seok Seo

Sin-seok Seo (sesise@postech.ac.kr) received his B.Sc. degree in Computer Science and Engineering from Inha University in 2008. Currently, he is a PhD candidate in the Dept. of Computer Science and Engineering, POSTECH, Pohang, Republic of Korea. His research interests include autonomic network management and mobile device management.

James Won-Ki Hong

James Won-Ki Hong (jwkhong@postech.ac.kr) is the Head and professor in the Division of IT Convergence Engineering and Dept. of Computer Science and Engineering, POSTECH. He received a Ph.D. degree from University of Waterloo in 1991 and an M.S. degree from the University of Western Ontario in 1985. His research interests include network and systems management, distributed computing, network monitoring and analysis, and network planning. He has served as Technical Chair (1998-2000), Vice Chair (2003-2005), and Chair (2005-present) of the IEEE ComSoc Technical Committee on Network Operations and Management (CNOM). He has also served as Director of Online Content for IEEE ComSoc (2004–2005). He is a NOMS/IM Steering Committee Member and a Steering Committee Member of APNOMS. He was technical co-chair of NOMS 2000 and APNOMS '99. He was Finance Chair for IM '09, NOMS '06 and IM '05, and Finance Chair and Chair of the Local Planning Committee for NOMS '04. He was General Chair of APNOMS '06 and General Co-Chair of APNOMS'08 and APNOMS 2011. He was a General Co-Chair of NOMS 2010, which was held in Osaka, Japan in April 2010. He is an Associate Editor-in-Chief of IJNM and an editorial board member of IEEE TNSM, JNSM, JCN, and JTM. He is also Editor-in-Chief of KNOM Review Journal. He is a member of KICS, KNOM and KISS.