Energy Modeling of System Settings: A Crowdsourced Approach

Ella Peltonen*, Eemil Lagerspetz*, Petteri Nurmi*†, Sasu Tarkoma*†
†Helsinki Institute for Information Technology HIIT, Department of Computer Science, University of Helsinki
*Department of Computer Science, University of Helsinki, PO Box 64, FI-00014, University of Helsinki, Finland
firstname.lastname@cs.helsinki.fi

***Abstract*—The question ”Where has my battery life gone?” remains a common source of frustration for many smartphone users. With the increased complexity of smartphone applications, and the increasing number of system settings affecting them, understanding and optimizing battery use has become a difficult chore. The present paper develops a novel approach for constructing *energy models* from *crowdsourced* measurements. In contrast to previous approaches, which have focused on the effect of a specific sensor, system setting or application, our approach can simultaneously capture relationships between multiple factors, and provide a unified view of the energy state of the mobile device. We demonstrate the validity of using crowdsourced measurements for constructing battery models through a combination of large-scale analysis of a dataset containing battery discharge and system state measurements and hardware power measurements. The results indicate that the models captured by our approach are both in line with previous studies on battery consumption and empirical measurements, providing a cost-effective way to construct energy models during normal operations of the device. The analysis also provides several new insights about battery consumption. For example, our analysis shows the energy use of high CPU activity with automatic screen brightness is actually higher (resulting in around 9 minutes shorter battery lifetime on average) than with a medium CPU load and manual screen brightness; a Wi-Fi signal strength drop of one bar can result in a battery life loss of over 13%; and a smartphone sitting in the sun can experience over 50% worse battery life than one indoors in cool conditions.**

***Index Terms*—Mobile, Subsystems, Energy**

I. Introduction

Samsung sold over 40 million Galaxy S4 smartphones in six months in 2013[1]. Apple sold over 43 million iPhones in the second quarter of 2014[2]. Most of us use a smartphone daily, for work, entertainment and a variety of other purposes besides communication [3]. The processing and transmission power of smartphones continues to grow [21], while their batteries remain largely unchanged [10]. Consequently, energy efficiency remains a high priority for current smartphone operating systems, and increasingly, for applications. The importance of energy efficiency has also been highlighted in several user studies, which have shown that users actively take measures to optimize the power consumption of their device [5], [19].

Modern smartphones incorporate several mechanisms for optimizing battery consumption. On the operating system level, complex on-demand resource optimization strategies are used to reduce battery consumption [20]. However, the effectiveness of these policies is highly dependent on the context of the user, and there often are complex interdependencies that make it difficult to determine the optimal policy for a given situation. Ensuring the effectiveness of these policies requires fine-grained models that can characterize how different contexts and device features influence the power consumption of the device.

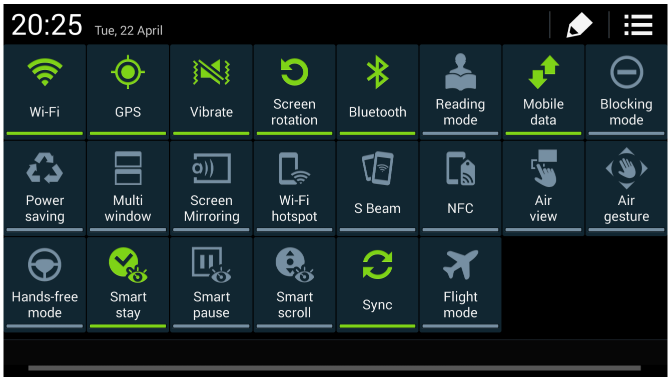

Fig. 1. The number of system settings available on current smartphones can be overwhelming.

The alternative to automated policies is to give users control over specific system settings, such as whether to prefer Wi-Fi or cellular networks, which screen brightness to use, and when to turn off the screen after inactivity. Indeed, contemporary smartphones have interfaces that allow this kind of operations with little effort. With an increasing number of user-controllable system settings, keeping track of each setting’s energy impact becomes unmanageable. To illustrate this problem, Fig. 1 depicts some of the system settings available on the Samsung Galaxy S4. Over 20 different settings are visible, most of which have a significant effect on energy. To fully understand the impact of all of the settings would require a considerable amount of learning. Furthermore, the total energy consumption of the smartphone is not simply the sum of the energy impacts of enabled system settings. Some subsystems, such as Bluetooth and Wi-Fi, are integrated on the same chip, and can be enabled simultaneously at much less than the sum of their combined individually measured energy impacts. Another example is the integration of accelerometers and gyroscopes on the same chip to provide energy savings for activity monitoring applications.

[1] http://www.theinquirer.net/inquirer/news/2302755/samsung-disappointed-with-galaxy-s4-sales-despite-hitting-40-million-milestone

[2] http://www.businesswire.com/news/home/20140423006671/en/Apple-Reports-Quarter-Results#.U17IYnEZscl

Enabling users to make optimal decisions would thus require fine-grained information about how different settings influence the overall battery consumption of the device in a given setting.

The present paper contributes by developing a novel approach for constructing *energy models* from crowdsourced battery discharge measurements. Such information can be increasingly collected through non-obtrusive instrumentation of the device [13], which in turn enables capturing battery information across a wide range of usage contexts and devices. Experiments conducted through a combination of power meter measurements and a large-scale analysis of crowdsourced discharge measurements demonstrate that our method is capable of constructing models that accurately capture complex interdependencies between system settings, sensors, and usage contexts, providing an accurate view of the *state* of the device. This contrasts with previous works, which have predominantly focused on capturing the effect of a specific sensor, system setting or application [4], [16]. Results from our evaluation provide novel insights about battery usage, demonstrating how complex the relationships between different factors and battery discharge are. For example, our analysis shows the energy use of high CPU activity with automatic screen brightness is actually higher (resulting in around 9 minutes fewer battery lifetime on average) than with a medium CPU load and manual screen brightness; a Wi-Fi signal strength drop of one bar can result in a battery life loss of over 13%; and a smartphone sitting in the sun can experience over 50% worse battery life than one indoors in cool conditions.

The contributions of the paper are summarized as follows:

- We develop a novel approach for constructing energy models from crowdsourced measurements. The models constructed by our approach can capture the *combined* effects of multiple factors simultaneously, providing a characterization of the *energy state* of a mobile device.
- Experiments carried out through a combination of power meter measurements and a large-scale analysis of crowdsourced discharge measurements demonstrate that our approach can capture the state of the device accurately and cost-effectively, even in the presence of complex interdependencies between context factors. Through our analysis, we also reveal new insights, highlighting the complexity of factors that influence battery consumption.
- We make available a large-scale data set of 11.2 million data points from around 150.000 active Android users[3].

II. Related Work

Mobile device energy profiling has been studied in the past [8], [14], [24], [25]. Most previous works take a holistic view of the device and its energy use [24], while some target application energy use specifically [11]–[14]. There are a number of systems that take into account the hardware subsystems of the device [25]. Some of these monitor inside the device [8] while others utilize a support server [1]. Most of these systems target specific sensors, such as location or Wi-Fi, or a subset of the sensors integrated on a typical smartphone. However, the complete state of the device requires considering all subsystems that can use energy. The BattOr [17] system can be used to monitor a mobile device in the wild, but it needs to be manually connected to the device prior to monitoring, and has a limited operating time. The DeviceAnalyzer project [22] is gathering rich measurements of mobile device state, but the data has not yet been used for large–scale analysis. To our best knowledge, no previous works are capable of constructing fine-grained energy models from crowdsourced measurements. Also, the effect of different system settings on battery consumption remains under-explored.

Sensor power profiling focuses on constructing power consumption models for individual sensors or sensor combinations. One of the earliest works in this vein was introduced by Rice and Hay [16], who examine fine-grained hardware power measurements and their causes, and attribute energy drain to the networking stack version, packet size, and Wi-Fi handshake behavior. König et al. [9] measure power consumption of different sensors using a hardware power monitor. Kjærgaard et al. [7] use conditional functions, manually constructed from empirical power measurements, to represent power consumption of different sensors. Kjærgaard and Blunk [8] propose using genetic algorithms for learning the conditional functions in an unsupervised manner. Contrary to our work, none of these approaches are capable of capturing complex interdependencies in the energy consumption of different components.

Another alternative is to construct statistical models that characterize the *overall* battery consumption of a device. These approaches consider how application usage patterns, workload and other system level parameters, such as screen brightness and data transfer rate influence battery discharge. The estimated discharge rate of the device can then be used to predict the remaining lifetime of the device's battery. Wen et al. [23] propose constructing a reference curve of the battery consumption under different workloads. Once the reference curve has been constructed, a regression model is used to compare current discharge with an estimate calculated using the reference curve. The deviations from the reference curve can then be used to refine estimates of remaining battery lifetime. Instead of considering workload, Kang et al. [6] predict discharge behavior from application usage patterns. Zhao et al. [26] predict battery discharge using a regression model that considers multiple different system variables (e.g., CPU utilization, I/O rate and LCD backlight brightness). In addition to predicting battery discharge, Ravi et al. [15] predict when the user is likely to have the next charging opportunity and how much battery power is needed for maintaining essential functionality until then. If the system detects that the battery is likely to run out before the next charging opportunity, the system pro-actively provides a warning to the user instead of waiting for the battery to be nearly depleted. Falaki et al. [4] conduct an analysis of smartphone usage patterns, revealing that usage patterns contain significant variation across users and that personalized application usage models are essential for accurate prediction of battery drain. In contrast to our approach,

[3] http://carat.cs.helsinki.fi/research

which can capture how changes in device state influence battery discharge, these approaches can only provide aggregate level information of power usage.

III. Background: Dataset

We consider a large-scale dataset of crowdsourced battery discharge measurements collected from a collaborative energy diagnostic system Carat [13]. The application has collected data from around 725,000 Android and iOS devices since summer 2012. We consider a subset, which we have made publicly available for research purposes[4], of the data containing 11.2 million samples from around 150,000 active Android devices. The data includes information about the device's operating system and model, the current battery level, the set of currently active applications, and information about different system settings such as network connections and screen brightness and subsystem variables such as the CPU use and the distance traveled since the last measurement. We refer to these system settings and subsystem variables collectively as *context factors*.

As a baseline for energy consumption, we consider the *energy rates* reported by Carat, which reflect normalized energy consumption per time unit, i.e., energy rate = Δ battery / Δ t. The methodology used to derive rates and the validity of using energy rates as a measure for battery consumption has been shown in previous work by Oliner et al. [13].

We focus on 13 different context factors, including 5 user-changeable system settings and 8 other pieces of subsystem state information. These were selected based on previous studies on energy-efficiency, which have shown them to be dominant factors explaining battery consumption. We consider the status defined by all 13 factors as the *state* of the device. The five system settings that we have collected via the Android API and that we utilize in this work are:

- **Mobile data status**, either connected, disconnected, connecting, or disconnecting,
- **Mobile network type**, such as LTE, HSPA, GPRS, EDGE, or UMTS,
- **Network type** used for Internet connectivity, either none, Wi-Fi, mobile, or wimax,
- whether **Roaming** is enabled or disabled, and
- **Screen brightness**, 0-255 or ”automatic” (-1).

We have also collected information about 8 subsystem variables. These are not directly available as a user-modifiable system setting, but can give information about the state of the smartphone. For example, if we notice a decreased Wi-Fi link speed or signal strength, we can recommend that the user try to use the mobile network instead of Wi-Fi in this context. This state information includes:

- **Battery health**, determined by the smart battery API of the Li-Ion battery of the Android device,
- **Battery temperature** in degrees Celsius,
- **Battery voltage** in Volts,
- **CPU use** in percent,
- **Distance traveled** between two samples in meters,

Context Factor	Mean	Std	Median
CPU use	75%	33%	91%
Battery voltage (V)	3.78	0.61	3.84
Screen brightness (0-255)	128.03	85.71	109
Temperature (°C)	29.27	5.75	30
Wi-Fi signal strength (dBm)	-61.29	13.02	-61

TABLE I
Summary statistics of selected context factors.

- **Mobile data activity**, one of none, out, in or inout,
- **Wi-Fi link speed**, in Mbps, and
- **Wi-Fi signal strength** in dBm.

Most context factors are nominal-valued. To simplify comparison of these factors, we have discretized them into categories using an equal frequencies procedure, i.e., each factor was divided into categories containing approximately the same number of values. The number of categories was determined empirically and based on observations reported in previous battery usage studies. Summary statistics of selected context factors are given in Table I and the different categories are detailed below. For categorical variables (such as network type), we have considered the different possible values as categories.

CPU use: We consider measurements that reflect the percentage of time CPU is active. The mean and median in Table I indicate that CPUs are mostly active. We split the CPU use around the mean, resulting in three categories: Low (0 - 42%), Medium (43 - 85%), and High (86 - 100%).

Distance traveled: Most values are during stationary periods or with little movement. Based on this observation, we consider a split between stationary and non-stationary behavior.

Battery voltage: The safe operating voltage of a smartphone Li-Ion battery is 3 - 4.2V. The nominal voltage of such batteries is typically 3.7V. The mean, the median, and the standard deviation reflect this very closely. We consider three categories for voltage: Low (0 - 3V), Medium (3 - 4.2V), High (4.2V+).

Screen brightness: When screen brightness was manually controlled, the mean was around 128, or the exact midpoint. The standard deviation of the values suggests that almost the entire range of brightness settings is used, making it difficult to categorize the screen brightness values. Accordingly, we consider a binary split into manual and automatic brightness.

Wi-Fi signal strength: We consider RSS values in the range [-100, 0]. Good Wi-Fi signal strength values are normally between -30 and -10dBm, and the worst, while still being connected, is -95dBm. We consider four categories: Bad (-100 to -75dBm), Average (-74 to -61dBm), Good (-61 to -49dBm) and Excellent (-49 to 0dBm). The mean RSS is between the Average and the Good levels, and the Excellent and the Bad levels are within one standard deviation. These values are in line with typical values used in Wi-Fi positioning literature.

Ethical Considerations: We consider only aggregate level data which contains no personally identifiable data. The privacy protection mechanisms of Carat are discussed in detail by the authors [13]. Data collection by Carat is subject to the IRB process of UC Berkley. Users of Carat are informed about the collected data and give their consent from their devices.

[4] http://carat.cs.helsinki.fi/research

IV. Battery Modeling from Crowdsourced Measurements

Battery consumption has traditionally been based on empirical models taken either directly on the battery level [17] or through system-level APIs. The former requires specialized measurements tools, limiting the contexts where measurements can be taken. The latter, on the other hand, has been shown to result in inaccuracies in the resulting models [25]. In this section we demonstrate the validity of using crowdsourced battery discharge measurements for constructing battery consumption models. Our approach provides a cost-effective alternative for modeling battery consumption, and, as we later demonstrate, our approach can capture complex interdependencies affecting battery consumption in everyday use.

A. Methodology

We construct battery models by measuring the strength of statistical association between context factors and battery discharge rates. To measure statistical association, we consider two complementary metrics. As our first metric, we consider gain in battery life, denoted *BL Gain*, which measures how changes in context factors influence the lifetime of a device on average. As our second measure, we consider the *conditional mutual information* (CMI) between context factors and energy rates. For assessing the influence of a single context factor X and energy rate Z, the CMI is equivalent to the *mutual information* (MI) given by:

$$MI(X,Z) = \sum_{z\in Z}\sum_{x\in X} p(x,z)\cdot log\left(\frac{p(x,z)}{p(x)\cdot p(z)}\right).$$

For higher order combinations containing two or more context factors (denoted X and Y), the CMI is defined as follows:

$$CMI(X,Y|Z) = \sum_{z\in Z}\sum_{y\in Y}\sum_{x\in X} p(x,y,z)\cdot \log\left(\frac{p(z)\cdot p(x,y,z)}{p(x,z)\cdot p(y,z)}\right).$$

The battery life gain measurements provide information about *absolute differences*, whereas the (C)MI measurements can be used for *relative* comparison between different context factors. The two metrics provide complementary ways to analyze strength of associations, and in applications the choice of metric depends on the scenario being considered.

B. Individual Context Factors

We demonstrate the validity of using battery discharge measurements for constructing energy models by examining the mutual information between context factors and energy rates. We derive a ranking for different factors based on their mutual information values, and demonstrate that this ranking is in line with findings from empirical studies on battery consumption.

Estimations by mutual information MI of context factors and energy consumption are given in Table II. The results of the MI analysis are well in line with previous results [4], [18]. In particular, the major individual impact of CPU use and traveled distance on battery consumption is clearly observable, and the ordering of the settings is similar to those derived through explicit battery measurements.

Context Factor	MI Estimate
CPU use	1.330
Distance traveled	1.069
Battery temperature	0.143
Battery voltage	0.099
Screen brightness	0.030
Mobile network type	0.019
Network type	0.018
Wi-Fi signal strength	0.014
Wi-Fi link speed	0.014
Mobile data status	0.013
Mobile data activity	0.005
Battery health	0.004
Roaming	0.0002

TABLE II

Context factors’ impact on energy consumption, ordered by mutual information estimate.

Context Factor	Value	BL Gain
CPU use	Low (0–42%)	+3.24%
CPU use	Medium (43–85%)	+5.72%
CPU use	High (86–100%)	-2.48%
Distance traveled	None	-0.76%
Distance traveled	>0	+8.20%
Battery voltage	Low (0–3V)	-16.60%
Battery voltage	Medium (3–4.2V)	-0.76%
Battery voltage	High (4.2V+)	+69.08%
Screen brightness	Manual	-4.96%
Screen brightness	Automatic	+6.29%
Wi-Fi signal strength	Bad (-100 – -75 dBm)	-2.29%
Wi-Fi signal strength	Average (-74 – -61 dBm)	+4.00%
Wi-Fi signal strength	Good (-61 – -49 dBm)	+6.29%
Wi-Fi signal strength	Excellent(-48 – 0 dBm)	+7.63%

TABLE III

The expected energy use typical values of context factors. BL gains can be compared to statistics in Table I.

The results also contain some exceptions to the findings of previous studies. The most prominent example of these is screen brightness, which is commonly considered the most battery heavy feature. In our analysis, screen brightness results in a lower score than many other attributes. In the next section, we demonstrate that the absolute energy impact of screen brightness actually is high. However, as mutual information effectively looks at the correlation between battery discharge and context factors, the changes are affected by other context factors. In most use contexts screen use is correlated with battery voltage and CPU use, both of which have a large impact on the battery drain, and hence also on the mutual information values. The main effect we observe for screen brightness comes from switching to automatic brightness.

C. Energy Consumption of Context Factors

We next consider how typical values of context factors influence battery discharge. We focus on the five factors discussed in Section IV-A (CPU use, battery voltage, screen brightness, temperature, and Wi-Fi signal strength) and consider expected gain in battery life as our evaluation measure. The results of this evaluation are shown in Table III.

In line with the results of mutual information analysis, the worst battery life is obtained for high CPU use. The benefit of maintaining a balanced CPU load is significant, as medium CPU use produces +5.72% energy benefit compared to average use. For screen brightness, the automatic setting of the device usually improves battery life, providing even +6.29% better battery life compared to the average. Manual brightness, in contrast, shows a major loss of battery life (-4.97%).

We can make a number of other observations from the results. First, higher battery voltage results in improved battery life. This is partially explained by voltage correlating with battery health and capacity. In addition, the rates reported by Carat are based on changes in battery percentage, which tend to follow voltage changes linearly. This contrasts with actual discharge, which is nonlinear, particularly when the battery is close to full charge. The results also suggest that battery life tends to be higher for mobile than for stationary users. Studies on application usage have shown that interactions with applications are common during mobility, with web browsing, news, music/video players, and gaming being the dominant application categories [3]. Hence, the difference is likely a result of shorter interaction periods rather than avoidance of energy intensive operations. Finally, a high Wi-Fi signal strength leads to better battery life, as the phone needs to spend less energy for receiving and sending data. Bad signal availability can lead to situations where the device has to reconnect to the network repeatedly, further increasing battery consumption.

V. Context Factor Combinations

The results of our analysis thus far show that estimates given by our method are in line with observations made in studies carried out in laboratory conditions with specialized hardware measurement tools, providing a strong indication of the potential of using our approach as a cost-effective mechanism to construct models of battery consumption. However, a limitation of our analysis thus far has been the focus on individual factors' impact on battery life, without considering the state of the device as a whole. As an example, consider the case of screen brightness, which according to previous studies is one of the main battery hogs on a smartphone. In terms of expected battery gain, our results also support this observation, indicating over 5% deviations from average consumption patterns. In actual application contexts, screen usage is highly correlated with interactions on the device, which in turn require CPU and network usage, suggesting that screen brightness is not the *dominant* factor explaining battery usage. To capture such nuanced differences in consumption, we argue that models reflecting the *state* of the smartphone are required. In the following we demonstrate the validity of considering combinations of context factors as part of battery models. We also compare our results against empirical power models constructed using a hardware power monitor, demonstrating that our approach can capture more fine-grained differences in battery consumption than empirical power models.

A. Statistical Results of Context Factor Combinations

We first demonstrate the complexity of battery consumption patterns by considering how pairs of context factors influence consumption. Similarly to the previous section, we derive a ranking for the different pairs by considering the conditional mutual information between each pair, and rank the pairs in descending order of CMI values. The results of these estimations are listed in Table IV.

Context Factors		CMI
Battery voltage	CPU use	4.29
CPU use	Screen brightness	2.17
Battery temperature	CPU use	2.07
CPU use	Distance traveled	1.81
CPU use	Wi-Fi signal strength	1.69
Battery voltage	Distance traveled	1.53
Battery temperature	Distance traveled	1.28
Distance traveled	Screen brightness	1.26
CPU use	Wi-Fi link speed	1.12
Battery voltage	Screen brightness	1.08
Wi-Fi link speed	Wi-Fi signal strength	0.99
Mobile data status	Network type	0.95
Network type	Wi-Fi signal strength	0.85
CPU use	Mobile network type	0.80
Battery temperature	Screen brightness	0.79
Distance traveled	Wi-Fi signal strength	0.75
Network type	Wi-Fi link speed	0.64
Mobile data status	Wi-Fi signal strength	0.60
Battery temperature	Battery voltage	0.56
Distance traveled	Wi-Fi link speed	0.54
Battery voltage	Wi-Fi signal strength	0.53

TABLE IV
Top of the conditional mutual information estimates for pairs of context factors for energy consumption rates.

Compared to the results of individual context factors' impact (see Table II), the combination of multiple factors gives more accurate explanations of the battery consumption. A prominent example is CPU use, for which we can observe significantly higher impact when combined with another factor than when considered alone. Also factors related to network connection, such as Wi-Fi signal strength and network type, differ clearly from the MI analysis. Both have lower MI values in Table II, but are more prominent when considered in conjunction with another context factor. Wi-Fi link speed and Wi-Fi signal strength have a combined MI of 0.99, which is higher than they get separately (0.014 each). Capturing this kind of nuances in consumption is particularly beneficial when giving suggestions to the end user on how to improve battery life. For example, from the results we can observe that changing the other system setting can help to improve battery life in cases where high CPU use is mandated, for example, when playing a game.

The top context factors according to energy consumption seem to be battery voltage, CPU use, battery temperature, and movement (distance traveled) of the device, or combinations thereof. The effects of these factors are mediated by other factors, which in turn can cause significant increases or decreases in consumption. Accordingly, providing an accurate view of the battery consumption of a device requires models that can capture both the effects of multiple context factors and the effects of their interdependencies.

B. Battery Consumption of Context Factor Combinations

To further illustrate the complexity of battery consumption patterns, we consider how selected context factor combinations

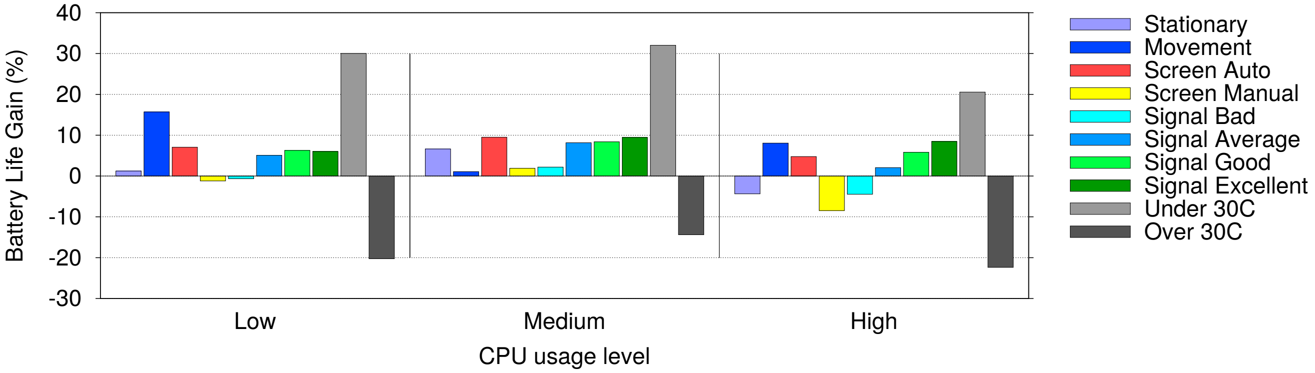

Fig. 2. Battery gain from CPU use combined with another context factor.

affect battery consumption. We chose combinations with high CMI values (see Table IV), and measured their expected battery life gain. As an example case, we consider the impact of CPU use and another factor on battery consumption. The results of this analysis are shown in Fig. 2. The y-axis shows the battery life gain in percentages when compared to the average expected battery life in our dataset. The different columns represent different values of context factors.

From the figure we can observe that the influence of context factor combinations on battery consumption is rather complex. For example, as long as the phone can observe average Wi-Fi signal strength, improving the connection will not provide significant savings unless CPU use is very high. In terms of screen brightness, as shown previously, the main effect results from switching to automatic brightness. However, this effect is most beneficial for moderate CPU use, and during high use, other factors can provide more pronounced changes. Another important factor is battery temperature, which can result in a loss of up to 50% battery life. The effects of temperature are consistent across all CPU use categories, indicating CPU use is not necessarily the (sole) cause for high battery temperature.

C. Power Meter Validation

The proposed approach of using crowdsourced measurements for constructing battery consumption models has been intended as a cost-effective way to capture fine-grained and nuanced differences in battery consumption. As we have demonstrated, these differences can have a significant impact on battery consumption and need to be accounted for to provide accurate estimates of the actual battery usage. We next compare our approach against empirical power models constructed using a hardware power monitor. We demonstrate that our approach is better at capturing the effects of changes in device state on battery consumption. In particular, our analysis indicates that empirical models are dominated by *instantaneous* effect, which tends to overestimate overall power consumption.

We consider measurements collected using a Samsung Galaxy S2 phone that was connected to a Monsoon Power Monitor[5]. The Power Monitor was set to output a constant voltage of 4.0V. The phone battery was still inserted, with the + and - terminals blocked, letting the phone start up normally. Each experiment run lasted 10 minutes. We connected the phone to the cellular network and turned the phone screen on for the duration of each experiment run. Wi-Fi was enabled and Bluetooth disabled for all our experiment runs. Automatic updating of applications was disabled. All applications were closed. Before each experiment run, we let the power consumption stabilize for several minutes to avoid the impact of background activity on the experiment. The experiment runs consisted of the following configurations:

- Full screen brightness, 30% CPU use, and bad (1); average (2); and good Wi-Fi signal strength (3).
- Full screen brightness, 60% CPU use, and bad (4); average (5), and good Wi-Fi signal strength (6).
- Full screen brightness, 100% CPU use (two tightly looping threads), and bad (7); average (8); and good Wi-Fi signal strength (9).
- Automatic screen brightness, average Wi-Fi signal strength, and 30% (10); 60% (11); 100% CPU use (12).

The impact of considering combinations instead of individual factors can be assessed by examining the relative standard deviations (i.e., ratio between standard deviation and mean) of the power monitor measurements. These are illustrated in Fig. 3. In the figure we consider separately the combined effect of CPU and Wi-Fi, and that of CPU and screen brightness. We also consider how decomposing these factors into categories influences the measurements.

The column groups in the figure correspond to the Wi-Fi signal strength range, and an average value for all of the values (signal all). Respectively, for screen brightness we consider automatic and manual, and average over all values (screen both). The columns in each group, from left to right, are CPU All (average over all use levels), Low, Medium, and High CPU use level. From the figure we can observe that the relative standard deviations for combined CPU use (i.e., columns with CPU All) are much higher than those of individual use levels by a factor of at least 1.5. The same observation applies for signal strength and screen brightness, indicating that considering individual factors is much less accurate at explaining battery consumption than the combination of multiple factors.

[5] https://www.msoon.com/LabEquipment/PowerMonitor/

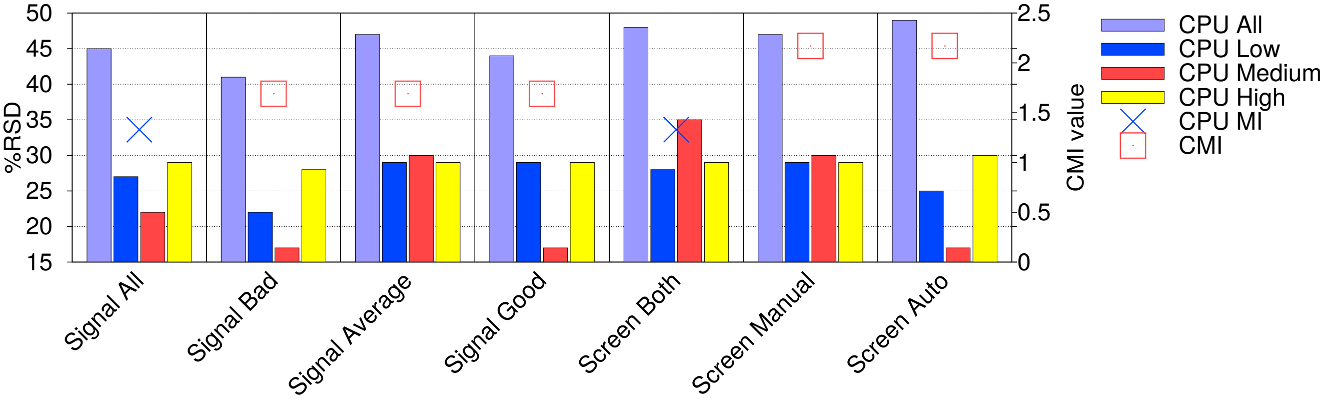

Fig. 3. Relative Standard Deviation (%RSD) of various CPU use and Wi-Fi signal strengths and screen brightnesses, along with their MI and CMI values.

CPU Use	All	Low	Medium	High
Experiment				
All Wi-Fi	2.60	4.77	2.23	1.90
Bad Wi-Fi	2.51	4.52	2.20	1.93
Average Wi-Fi	2.60	4.99	2.27	1.91
Good Wi-Fi	2.53	4.82	2.21	1.87
All screen br.	2.69	5.40	3.27	1.92
Manual screen br.	2.60	4.99	2.27	1.91
Auto screen br.	2.78	5.89	2.50	1.92
Crowdsourced data				
All Wi-Fi	5.24	5.53	5.54	5.53
Bad Wi-Fi	5.12	5.21	5.36	5.01
Average Wi-Fi	5.45	5.51	5.67	5.35
Good Wi-Fi	5.57	5.57	5.68	5.54
All screen br.	5.24	5.53	5.54	5.53
Manual screen br.	4.98	5.18	5.34	4.80
Auto screen br.	5.57	5.61	5.74	5.49

TABLE V

Comparison of average battery lifetime improvement (in hours) between power monitor estimates and our approach.

We next compare battery life estimates between power monitor measurements and our approach. The results of this analysis are shown in Table V. From the results we can make two important observations. First, for low CPU use, the power monitor measurements are closely in line with the estimates provided by our approach. However, as CPU use increases, the estimates provided by power monitor measurements indicate significant decrease, whereas the estimates provided by our approach are much less affected. Current smartphones have several mechanisms to adjust CPU use and consistently high use is rare. Accordingly, while the power monitor estimates reflect actual battery consumption, they are over-zealous and overfit on the instantaneous consumption. In contrast, our approach can average over different usage contexts, providing a more realistic estimate of the practical impacts on battery consumption. Second, the results demonstrate that power monitor measurements are poor at identifying the relative importance of different context factors. For low CPU use, the difference between manual and automatic screen brightness is clearly observable. However, as CPU use increases, the power monitor models overfit on the high power consumption of CPU, providing limited information about the importance of other context factors. In contrast, the impacts of changing screen brightness to automatic and the impact of improving Wi-Fi signal level remain observable across all CPU use levels for our approach. In Fig. 3, we can observe that these differences are even more observable from the conditional mutual information.

D. Highlights and Example Cases

As the final step of analysis, we demonstrate how our approach can also be used to obtain new insights into battery consumption. Examples of selected context factors' impact on battery consumption are listed in Table VI. We have selected CPU use and temperature from subsystem variables, and distance (motion or stationary) and screen brightness from system settings. In all examples, connection type has been a cellular data connection. Table VI presents the estimated time to drain the battery from 100% to 0%, while actively using a smartphone with the given context factor and value combination. With different values of CPU use, battery temperature, movement, and screen brightness, the battery life can range from 3.45 hours up to 9.12 hours.

The table is sorted by the time to drain the battery, descending. We can see that the main deciding factor for battery life is the temperature of the battery. With a lower temperature, we get a higher battery life. After that, traveling instead of staying still seems to increase battery life. This may be due to users driving and not using their mobile phones while mobile. After these factors, the CPU is the most dominant, and changing screen brightness brings the smallest, but still significant, battery life differences. These results show that while CPU use alone is a good indicator of energy consumption, significant gains can be obtained by considering more complex combinations. In addition to this, battery temperature and distance traveled can be used together to predict battery life.

Complex combinations of factors, such as those listed in Table VI, can be used to decide which factors to change to improve battery life, while keeping others constant. For example, while moving and playing a game, the CPU is often high. If the phone can be kept relatively cool, 78% more battery life can be expected compared to warmer battery (increase from 4.08h to 7.27h). Further savings can be obtained by switching screen brightness to automatic.

Battery Temperature	Distance Traveled	CPU Use	Screen Brightness	Estimated Battery Life (h)
Under 30°C	>0	Low	Automatic	8.83 – 9.12
Under 30°C	>0	Low	Manual	8.49 – 8.82
Under 30°C	>0	High	Automatic	8.09 – 8.24
Under 30°C	>0	Medium	Automatic	7.65 – 7.89
Under 30°C	>0	Medium	Manual	7.34 – 7.60
Under 30°C	>0	High	Manual	7.27 – 7.41
Under 30°C	None	Medium	Automatic	6.57 – 6.64
Under 30°C	None	Low	Automatic	6.28 – 6.35
Under 30°C	None	Medium	Manual	6.13 – 6.20
Under 30°C	None	Low	Manual	5.88 – 5.96
Under 30°C	None	High	Automatic	5.78 – 5.82
Over 30°C	>0	Low	Automatic	5.08 – 5.22
Under 30°C	None	High	Manual	5.00 – 5.04
Over 30°C	>0	Low	Manual	4.73 – 4.88
Over 30°C	>0	High	Automatic	4.62 – 4.69
Over 30°C	>0	Medium	Automatic	4.59 – 4.70
Over 30°C	>0	Medium	Manual	4.28 – 4.39
Over 30°C	None	Medium	Automatic	4.25 – 4.29
Over 30°C	>0	High	Manual	4.08 – 4.14
Over 30°C	None	Medium	Manual	4.06 – 4.09
Over 30°C	None	Low	Automatic	4.02 – 4.06
Over 30°C	None	High	Automatic	3.91 – 3.94
Over 30°C	None	Low	Manual	3.74 – 3.78
Over 30°C	None	High	Manual	3.45 – 3.46

TABLE VI

BATTERY LIFE IN HOURS FOR SELECTED COMBINATIONS OF FOUR CONTEXT FACTORS.

With respect to the worst possible configuration, moving to a cooler place (45% battery life gain) and changing screen brightness without changing the CPU use can result in a battery life increase from 3.45h to 5.78h (68%). Our results in Table VI and in Fig. 2 show that the battery temperature is not always directly related to CPU use. High battery temperature can be caused, for example, by the ambient temperature in warmer countries, battery misbehavior or a battery bug, or because the smartphone has been forgotten under the windshield inside a car on a sunny day. Battery temperature alone can shorten the battery lifetime even by 50%. If cooling the device is not possible, because of the ambient climate, for example, re-configuring other context factors can help to improve the battery lifetime.

With low CPU use and a cool battery, no movement, and manual screen brightness, we can obtain an active battery life of 6 hours, which improves to almost 9 hours by only changing movement. That behavior can be caused by the users mostly walking or driving a car and not using their smartphones while moving from place to another. It is also possible, that energy saving policies activate as movement requires re-connections to the cellular base stations. As Table IV shows in Section V-A, distance traveled is ranked high together with CPU use. It is possible that the most CPU heavy actions, such as gaming, are only done in longer periods while stationary.

VI. DISCUSSION AND SUMMARY

The present paper has provided three contributions. Our first contribution has been the development of a novel approach for constructing energy models using crowdsourced battery discharge measurements. Contrary to previous works on energy modeling, our approach is not restricted to capturing the effects of individual sensors, features or system settings, but can capture complex interdependencies between all of these. As we have experimentally demonstrated, estimates provided by our approach are in line with battery meter measurements, providing an accurate view of the energy state of the device. The second contribution is a large-scale analysis of the influence of different system settings on battery consumption. Our analysis validated our method and confirmed findings in previous studies. It also provided novel insights about battery consumption and quantified their effects. For example, we demonstrated that a Wi-Fi signal strength drop of one bar can result in a battery life loss of over 13% and that a smartphone sitting in the sun can experience over 50% worse battery life than one indoors in cool conditions. As our third contribution, we have made available the large-scale (anonymized) dataset used in our analysis[6].

Energy models that can accurately capture the energy state of a device and that can estimate how system state changes influence energy, are beneficial for several reasons. Our approach can be used to bootstrap and support battery management interfaces developed to support end users. Instead of merely allowing users to switch off (or on) different settings, our approach can estimate how these changes are expected to influence device lifetime. Our approach can be used to construct device-specific resource optimization strategies that can estimate changes in battery use more accurately. Our approach could be used to construct empirical energy models for comparing and evaluating energy-effectiveness of different sensing strategies.

In terms of battery management interfaces, an interesting avenue of investigation are task-based recommendations that provide actionable feedback to the user on how to preserve battery for her current tasks. For example, if the user intends to perform high CPU use activities, they can save battery life by setting screen brightness to automatic or moving to an area

[6] http://carat.cs.helsinki.fi/research

with a better Wi-Fi signal. Such recommendations can also help to increase the user's knowledge over time, familiarizing them with the inner workings of their smartphone. Similarly, if personal measurements would be available, our approach could be used to identify "bad" behaviors for a user and provide guidance on how to mitigate these. Another benefit of our approach is the capability to construct device or OS-specific energy models with minimal effort. The energy consumption of sensors and system settings can contain significant variations across platforms, e.g., Bhattacharya et al. [2] reported over 200% differences for GPS power consumption on two different Nokia smartphone models. Accordingly, we can tailor the guidance given to the user according to the model and operating system version of her device.

The results presented in this paper are also potentially beneficial for understanding long-term effects of sensor and battery management strategies on battery life. The comparison of battery life estimates between our approach and power monitor measurements showed that our approach can average effects over different usage contexts, whereas empirical power models tend to focus on instantaneous effects. As the overall state of a smartphone is complex, and in constant flux, instantaneous estimates tend to result in overestimates of battery consumption. Assessing the benefits of using crowdsourced battery models for these purposes is another interesting venue for future investigations.

Acknowledgments

The work of Ella Peltonen has been supported by Doctoral School of Computer Science (DoCS). This research was partially supported by the Academy of Finland grant 277498. The publication only reflects the authors' views. The authors are grateful to Dr Stephan Sigg, Teemu Pulkkinen, and Samuli Hemminki for comments on earlier versions of the paper.

References

[1] S. Agarwal, R. Mahajan, A. Zheng, and V. Bahl. Diagnosing mobile applications in the wild. In *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, pages 22:1–22:6, New York, NY, USA, 2010. ACM.

[2] S. Bhattacharya, H. Blunck, M. Kjærgaard, and P. Nurmi. Robust and energy-efficient trajectory tracking for mobile devices. *IEEE Transactions on Mobile Computing*, 99:1, 2014.

[3] M. Böhmer, B. Hecht, J. Schöning, A. Krüger, and G. Bauer. Falling asleep with Angry Birds, Facebook and Kindle: A large scale study on mobile application usage. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, 2011.

[4] H. Falaki, R. Mahajan, S. Kandula, D. Lymberopoulos, R. Govindan, and D. Estrin. Diversity in smartphone usage. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services (MobiSys 2010)*, pages 179–194. ACM, 2010.

[5] D. Ferreira, A. K. Dey, and V. Kostakos. Understanding human-smartphone concerns: A study of battery life. In *Proceedings of the 9th International Conference on Pervasive Computing*, 2011.

[6] J.-M. Kang, C.-K. Park, S.-S. Seo, M.-J. Choi, and J. Hong. User-centric prediction for battery lifetime of mobile devices. In *Proceedings of the 11th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, pages 531–534. Springer, 2008.

[7] M. B. Kjærgaard, S. Bhattacharya, H. Blunck, and P. Nurmi. Energy-efficient trajectory tracking for mobile devices. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys 2011)*, pages 307–320, New York, NY, USA, 2011. ACM.

[8] M. B. Kjærgaard and H. Blunck. Unsupervised power profiling for mobile devices. In *Proceedings of the 8th Annual International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (Mobiquitous)*, 2011.

[9] I. Koenig, A. Q. Memon, and K. David. Energy consumption of the sensors of smartphones. In *Proceedings of the Tenth International Symposium on Wireless Communication Systems (ISWCS 2013)*, pages 1–5, Aug 2013.

[10] D. Linden and T. B. Reddy. *Handbook of Batteries*. McGrew-Hill Professional, 2001. 3rd edition.

[11] Y. Liu, C. Xu, and S. C. Cheung. Where has my battery gone? Finding sensor related energy black holes in smartphone applications. In *2013 IEEE International Conference on Pervasive Computing and Communications, PerCom 2013, San Diego, CA, USA, March 18-22, 2013*, pages 2–10, 2013.

[12] X. Ma, P. Huang, X. Jin, P. Wang, S. Park, D. Shen, Y. Zhou, L. K. Saul, and G. M. Voelker. eDoctor: Automatically diagnosing abnormal battery drain issues on smartphones. In *Proceedings of the 10th USENIX conference on Networked Systems Design and Implementation*, pages 57–70, Berkeley, CA, USA, 2013. USENIX Association.

[13] A. J. Oliner, A. P. Iyer, I. Stoica, E. Lagerspetz, and S. Tarkoma. Carat: Collaborative energy diagnosis for mobile devices. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, pages 10:1–10:14, New York, NY, USA, 2013. ACM.

[14] A. Pathak, Y. C. Hu, and M. Zhang. Where is the energy spent inside my app? Fine grained energy accounting on smartphones with Eprof. In *Proceedings of the 7th ACM European Conference on Computer Systems*, pages 29–42, New York, NY, USA, 2012. ACM.

[15] N. Ravi, J. Scott, L. Han, and L. Iftode. Context-aware battery management for mobile phones. In *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 224–233, 2008.

[16] A. Rice and S. Hay. Decomposing power measurements for mobile devices. In *Proceedings of the 2010 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 70–78. IEEE, 2010.

[17] A. Schulman, T. Schmid, P. Dutta, and N. Spring. *Demo: Phone Power Monitoring with BattOr*, 2011. ACM Mobicom 2011. Available at http://www.stanford.edu/ aschulm/battor.html.

[18] A. Shye, B. Scholbrock, and G. Memik. Into the wild: Studying real user activity patterns to guide power optimizations for mobile architectures. In *Proceedings of the 42nd Annual IEEE/ACM International Symposium on Microarchitecture*, pages 168–178, New York, NY, USA, 2009. ACM.

[19] K. N. Truong, J. A. Kientz, T. Sohn, A. Rosenzweig, A. Fonville, and T. Smith. The design and evaluation of a task-centered battery interface. In *Proceedings of the12th International Conference on Ubiquitous Computing*, 2010.

[20] N. Vallina-Rodriguez and J. Crowcroft. Energy Management Techniques in Modern Mobile Handsets. *IEEE Communications Surveys and Tutorials*, 15:179–198, 2013.

[21] C. H. K. van Berkel. Multi-core for mobile phones. In *Proceedings of the Conference on Design, Automation and Test in Europe (DATE '09)*, pages 1260–1265, 2009.

[22] D. T. Wagner, A. Rice, and A. R. Beresford. Device Analyzer: Large-scale mobile data collection. *ACM SIGMETRICS Performance Evaluation Review*, 41(4):53–56, 2014.

[23] Y. Wen, R. Wolski, and C. Krintz. Online prediction of battery lifetime for embedded and mobile devices. In *Proceedings of the 3rd International Workshop on Power-Aware Computer Systems (PACS)*, pages 131–138. Springer, 2005.

[24] F. Xu, Y. Liu, Q. Li, and Y. Zhang. V-edge: Fast self-constructive power modeling of smartphones based on battery voltage dynamics. In *Proceedings of the 10th USENIX Conference on Networked Systems Design and Implementation*, pages 43–56, Berkeley, CA, USA, 2013. USENIX Association.

[25] L. Zhang, B. Tiwana, Z. Qian, Z. Wang, R. P. Dick, Z. M. Mao, and L. Yang. Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In *Proceedings of the 8th IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis*, pages 105–114, New York, NY, USA, 2010. ACM.

[26] X. Zhao, Y. Guo, Q. Feng, and X. Chen. A system context-aware approach for battery lifetime prediction in smart phones. In *Proceedings of the 2011 ACM Symposium on Applied Computing (SAC)*, pages 641–646. ACM, 2011.