Using Decision-Theoretic Experience Sampling to Build Personalized Mobile Phone Interruption Models

Stephanie Rosenthal, Anind K. Dey, Manuela Veloso

Carnegie Mellon University
{srosenth, anind, veloso}@cs.cmu.edu

Abstract. We contribute a method for approximating users' interruptibility costs to use for experience sampling and validate the method in an application that learns when to automatically turn off and on the phone volume to avoid embarrassing phone interruptions. We demonstrate that users have varying costs associated with interruptions which indicates the need for personalized cost approximations. We compare different experience sampling techniques to learn users' volume preferences and show those that ask when our cost approximation is low reduce the number of embarrassing interruptions and result in more accurate volume classifiers when deployed for long-term use.

Keywords: interruptibility, preference elicitation, mobile devices, machine learning

1 Introduction

As mobile devices become increasingly ubiquitous in our environments, they increasingly ring or beep at inappropriate times or in inappropriate contexts such as in meetings or in movies. While we receive reminders to turn off our phones or put them in silent mode in these contexts, we often forget to do so which can result in embarrassing situations. Even when we do remember, we then forget to turn the ringer on afterwards resulting in missed calls [21] or missed notifications about SMS messages and calendar events. In this work, we are interested in learning users' *preferences* for receiving audible notification preferences in order to enable an application we built to automatically change the volume of users' phones.

Because users often forget to change their phone volumes themselves, we cannot automatically train a machine learning classifier using their volume settings as they are not an accurate indication of their actual volume preferences. Because we expect these preference rules to be complex, it is not feasible for users to define volume rules before using our application. Instead, our application elicits volume preferences from the user through experience sampling while they are using the phone [14, 24, 27]. However, the experience sampling itself may interrupt and embarrass the user in the same situations as the original notifications. In order to reduce these interruption *costs* associated with asking, Kapoor and Horvitz have proposed and demonstrated the success of a decision-theoretic experience sampling technique that builds accurate classifiers by asking for preferences only when the potential cost of misclassifying

that preference outweighs the interruption cost of asking now [13]. Our work builds upon this previous work to model user-specific costs (rather than an average cost for all users) while maintaining a high level of accuracy for all users.

In particular, while the previous work assigns constant costs for asking at an inappropriate time and for misclassifying preferences for all users and all situations, we show that different users have different costs and these costs vary for each user in different situations. One user may not want to be interrupted during work, another may not want to be interrupted during meetings at work but would answer if necessary, and another may have no problem being interrupted at work. A constant model cannot capture this complexity and the wrong model could severely impact the usability of the model for users who have high costs for interruption. We aim to address these potential usability problems by creating personalized cost models for each user. Although users may not be able to predefine their interruption costs for all situations (just as they cannot predefine preferences for a classifier), we assume they can approximate this cost for a broad set of situations that we survey them about. We propose that these approximations can be used to determine times to ask for notification preferences that reduce the interruption cost from asking while maintaining the high accuracy that Kapoor has shown previously.

We recruited participants to test the usability of our experience sampling technique against other commonly used techniques and to test the accuracy of the resulting classifiers' volume prediction. Prior to testing the on-phone application, participants filled out surveys about their predicted phone volume preferences in a variety of situations. Additionally, we asked for participants' predicted costs of being asked questions and the potential costs of an application misclassifying their preferences in each situation. Then, for two weeks, the application learned the users' preferences through one of three experience sampling techniques (random sampling, uncertainty sampling, and our augmented decision-theoretic sampling). For participants in our experience sampling condition, we used the survey costs to approximate, and to determine when to ask for, their preferences. Then, users tested the accuracy of their classifiers for an additional two weeks.

In this work, we make the following contributions. First, we contribute a method for approximating interruptibility costs and show that it improves the timeliness of questions asked during experience sampling. Second, we find that 7 out of 10 participants in the decision-theoretic condition reported very high accuracy (near 100%) with few or no errors while testing their classifier for two weeks. Third, we find that the user-specific cost models, while effective at improving usability for all users, reduced accuracy for the remaining 3 decision-theoretic participants as it asked too few questions and thus we caution using this technique for users with high asking costs. Finally, for these high cost users, we show that their initial preferences from the surveys can be used to create more accurate classifiers without sampling.

2 Related Work

As mobile phones are so ubiquitous and we increasingly have them available with us, it is becoming more important to understand when it is appropriate for them to

interrupt us through rings and beeps. While users can characterize their own interruptibility *preferences* by changing phone modes (*e.g.*, ring, vibrate, silent) to avoid unwanted phone calls [27], they often forget to set and reset their phone modes, resulting in unwanted interruptions or potentially missing important calls, or SMS or calendar notifications due to silent notifications [21]. With a model of interruptibility, a phone could automatically set its volume to avoid inappropriate interruptions and important missed calls.

Phones today offer a variety of sensors such as accelerometers, microphones, and GPS that can be leveraged to classify a user's context and interruptibility preferences. Studies have shown that human interruption in offices can be captured accurately by simple sensors such as these [6, 9], and other studies have found that users decide whether to answer their phones based on their activity, location, and who is calling – all of which are becoming more observable using current phone sensors [7, 15, 16]. With new applications to classify interruption preferences and react based on these predictions, it is not clear what accuracy level is acceptable for users. Kern and Schiele found that interruption classifiers generated by users predefining rules resulted in 80-85% accuracy (the highest of all classifiers they tested) [14]. In a simulated phone experiment, Khalil and Connelly found that users rated their simulated volume changer highly even though it incorrectly changed phone volume 9% of the time, but that different users had very different satisfaction levels with the classifier accuracy [15]. It is important to test machine learning classifiers to understand whether users find their accuracy tolerable for real world use.

While it is possible for machine learning researchers to collect data and build classifiers that apply to all users in some applications, it is infeasible for creating personalized preference models such as those for interruption because different people have different preferences. Additionally, because users often forget to change their phone volumes, their current volume settings are not an accurate indication of their actual volume preferences and the labels cannot be captured automatically as in [5] to learn email classifiers. However, Kern and Schiele argue that if the mobile device could use experience sampling [2, 23] to elicit preferences while the user is using the device, the resulting classifiers would be more accurate [14].

Many different experience sampling techniques have been proposed to accurately elicit data labels from users in order to build classifiers including diary studies [3], device-initiated questions at different intervals of time [10, 20], and based on context-awareness [11] and previous labels [26]. The active learning literature have also proposed a variety of ways to choose which data should be labeled [1, 12, 17, 18]. However, it has been shown that the frequency and repetition of questions can affect the accuracy and compliance with experience sampling [22]. Horvitz has argued [8] and attempts have been made in both the machine learning and experience sampling communities [4, 12, 13, 14] to take into account users' interruption *costs* to determine when to ask. Kapoor and Horvitz propose a decision-theoretic sampling approach that trades off an interruption cost of asking and a future cost of misclassification to limit the number of questions but these costs are not personalized for each user [13]. For example, one user may be more willing to answer even when they are busy in favor of producing a higher accuracy classifier while another wants to receive as few questions as possible. Additionally, Kapoor and Horvitz's resulting preference classifiers were not deployed to users so it is unclear whether their 70% accuracy obtained during the

experience sampling is tolerable for users. For clarity in our paper, we differentiate *interruptibility preferences* that are learned by the classifier from *interruption costs* of asking used to determine when to ask for preferences.

In this work, we aim to approximate users' individual interruption costs to improve the usability of an experience sampler by limiting the questions that are asked when each particular user is busy. In particular, we build upon Kapoor's decision-theoretic experience sampling technique to include our personalized cost of asking models that we approximate with users' survey responses. We use the interruptibility preference data collected via experience sampling to build a classifier to determine when users want their phone to ring (*i.e.*, when they are interruptible). We compare decision-theoretic sampling using our personalized cost models to more traditional experience sampling approaches and show that our personalized cost models lead to more timely questions for users and often led to nearly 100% accurate interruptibility preference classifiers. Additionally, we test our classifiers over two weeks to understand not only the costs of collecting personalized data but also the required accuracy of classifiers deployed to users in the real world.

3 Domain: Mobile Phone Interruptibility Preferences

We designed an Android application that learned users' volume preferences for phone calls, SMS messages, and calendar alarms. The application ran as a background process on the phone and listened for notifications (phone calls, incoming SMS messages, and calendar alarms). When a new notification arrived (*e.g.,* when the phone is about to ring), the application collected a variety of sensor and user-generated features and ran a classifier on those features to determine if the phone volume should be loud or silent. We did not turn on or off the vibration for this study.

Phone Interruption Features

We collected a variety of features based on sensor and other data that we can actively collect and have been shown to be effective at determining mobile interruptibility (*e.g.*, [7, 25, 27]) (Table 1). Examples of these features include GPS longitude, latitude, the time of day, and whether the user is talking on the phone. Additionally, the Android API provides information about the notification itself, which we will call the *reason* for the notification (in bold in Table 1). For phone calls and SMS messages, this includes information about the type of person who was contacting the user (*e.g.*, if they were in the user's favorites list, contact list, or neither) and the frequency of contact by this contactor. Calendar notification reasons included information about whether the calendar event was repeating versus a one-time event.

Due to the high battery cost of collecting this information on the phone, we only collected it when a new notification arrived with the exception of GPS coordinates. GPS coordinates were collected once per minute when the accelerometer values were above a certain threshold. Otherwise, it was assumed that the user was not moving and the GPS was turned off. As a result, the application had to quickly analyze the features and run the classifier to change the volume before the first ring or beep occurs, in case it was necessary to suppress it – in approximately ½ second.

GPS: Longitude, Latitude, Speed	Accelerometer X, Y, Z axes	Time until Next Meeting
User in Meeting	Noise (in dB)	Hour of Day
Day of Week	User on Phone	Count of Times On-Phone Caller has Contacted User
User on Phone with Someone in Contact List	User on Phone with Someone in Favorite List	**Next Meeting is a Repeated Meeting**
Contactor is in Contact List	**Contactor is in Favorites List**	**Count of Times Contactor Has Contacted User**

Table 1. Features used in our personalized cost models - *bold* indicate the notification context, while the rest describe the participants' situations.

Interruptibility Classification Model

In this work, we use logistic regression (LR) classifiers because of the computational speed and efficiency on small platforms such as phones. The LR model distinguishes between two "classes" of interruption preferences – those in which the phone should audibly ring (LOUD = 1) and those in which it should not (SILENT = 0) – using the features F defined in Table 1. In particular, for a new situation with features F, LR calculates the probability of those features being labeled as LOUD as:

$$P(\text{LOUD}|F) = \frac{1}{1 + exp(w_0 + \sum_{i=1}^{|F|} w_i F_i)}$$

If P(LOUD|*F*) is greater than 0.5, then the prediction is LOUD. Otherwise, the prediction is SILENT. The classifier defines the weights w_i by minimizing differences (errors) between the labels y^j that the user provides through experience sampling (training data) and the classifier's predicted label Y^j for each training example *j*:

$$\text{while } \sum_i \Delta(w_i) > \epsilon: \quad \forall_i, w_i \leftarrow w_i + \eta \sum_j f_i^j [y^j - P(Y^j = LOUD|f^j, w)]$$

We use experience sampling techniques to generate the training preference data that is used to learn to a classifier that distinguishes users' interruption preferences – when they want audible notifications. Each time a user responds to the experience sampler's question, the features of the current notification and the user's response are given to the LR classifier as training data to update the weights. Additionally, two of the experience sampling techniques - uncertainty and decision-theoretic sampling - use the classifier to determine whether to sample for preferences on new notifications.

Study Overview

Our study contains 3 parts. First, we surveyed users of mobile phones to understand their interruption preferences and interruption cost to learn those preferences in a variety of situations: at work, in the movies, at home. We will show that they not only had different preferences (as found in previous work) but also that they have different costs of asking. We then recruited participants to train a preference classifier for two weeks to understand the usability and accuracy of different sampling techniques. Finally, we tested the model of their personalized classifiers for an additional two weeks to understand whether the final accuracy is tolerable for the participants.

4 Experience Sampling to Acquire Training Data

Experience sampling was originally introduced to intentionally interrupt study participants in order to have them make notes about their current situations [2]. These interruptions could happen at regular or random intervals with the expectation that participants would be more accurate in describing their current situations in the moment rather than later during interviews. Rather than depend on users to define their preferences before our study or recall them each evening, we use this approach to collect user preferences for training our classifiers.

We want to use experience sampling to build and train personalized preference classifiers for mobile phone users without affecting the usability of our application. Unlike traditional experience sampling techniques in which the participant should be interrupted, we are interested in minimizing this interruption so that users are more likely to answer the questions over time [22]. Several techniques have been proposed for when to collect accurate data from users. However while some focused on minimizing the questions, they do not guarantee that questions minimize interruption.

Random Sampling

In *random sampling*, the decision to elicit the user's preferences is made irrespective of the classifier that is being built with the user's responses. It is likely that a preference may be asked for the same or very similar situations multiple times, making some of the elicitations extraneous. However, this sampler ensures that the there is a broad set of data to train a classifier with. In our work, we assume that a user's phone rings on average 3 times per day (participants were screened for this) and we want the phone to ask at least once per day so our random sampler elicits preferences approximately 1/3 of the time when the phone rings. To decide when to ask, the sampler generates a random number p between 0 and 1 and asks if $p < 0.3$.

Uncertainty-Based Sampling

Unlike random sampling, *uncertainty sampling* builds the preference classifier using the data collected so far and then decides whether to ask for a new preference based on the classifier prediction [1, 17]. The goal of uncertainty-based sampling is to reduce the number of labeled preferences by only asking in situations that have not previously been encountered. If a new situation is encountered, it may benefit the classifier to get the user's preferences in order to classify it correctly in the future. However, if a similar situation was already encountered, the user should not have to provide their preferences again.

Specifically, classifiers such as LR, output a real value p between 0 and 1 rather than the binary 0/1 classification with the rule that if $p <$ threshold of 0.5, then predict 0, otherwise predict 1. We use P(LOUD|F), defined above, as our uncertainty measure *p, where LOUD is defined as 1*. The closer to 0.5, the *less certain* the classifier is of the user's actual preference and the less likely it is that there is a previously labeled situation that is similar to the current one. Uncertainty sampling asks for the user's preference for the notification if the current classifier outputs a p between 0.3 and 0.7.

Decision-Theoretic Sampling

Recently, Kapoor and Horvitz introduced *decision-theoretic* sampling to limit the number of labels the sampler requests about the user's interruptibility by taking into account the *p* value from uncertainty sampling and other interruption cost information about the user [13]. When the uncertainty is high, this technique trades off a predefined cost of asking *A* (a user's cost of interruption for a question) with the cost of misclassification *M* (user's preference for accuracy) with the aim of collecting equal amounts of data when the user was busy and when the user was available. If the cost of asking is higher than the cost of misclassification, the assumption is that the user is busy. If the cost of misclassification is higher, the assumption is that he is more willing to answer. The decision-theoretic sampler asks for a user's volume preference if M > A, where M is defined in terms of the change in the prediction uncertainty (Δp) if the new data is added (details in [12]).

In Kapoor's work, the costs of asking and misclassification were kept constant across all users and equal – 1 each. However, some recent work has indicated that different users may deal with misclassifications differently [15]. Some users may have very high cost of misclassification and therefore may be much more willing to answer questions to train an accurate classifier or vice versa. By more accurately estimating these costs for each user, we argue that it is possible to create a more personalized asking mechanism that is more usable for each user. Like phone notifications themselves, it is difficult for users to predefine the situations in which they are willing to be asked questions. In order to approximate the cost of interruption to determine when to ask, we propose to survey users' interruption preferences with a set of concrete situations and use linear regression to interpolate to other situations that the user encounters during normal daily phone use. We will compare the usability and accuracy of our augmented decision-theoretic experience sampling approach against the other experience sampling techniques.

5 Approximating Cost Models with Surveys

In order to understand phone users' predicted volume preferences and interruption and misclassification costs across a variety of situations, we surveyed users of smart phones who receive several phone calls, SMS messages, and calendar alarms daily. Participants were asked to rate their preferences for receiving audible notifications in a variety of hypothetical, but real world, situations and their expected costs to train the classifier. We analyzed the differences in preferences and cost ratings between participants in the same situation as well as differences that a single participant provided across multiple situations to determine if a single approximation (as found in [15]) is sufficient or if personalized approximations are also needed.

Method

Before the survey began, participants were first asked a series of questions about their work schedule and common modes of transportation, which might affect their survey responses about situations in which they want audible notifications. Participants were then given 20 hypothetical situations when their phone might display a notification

Notification Type	Notification Context	Question
Phone	Favorite List, Contact List, Frequently Calls	If you were at work in a meeting and someone in your **favorites list** called, would you want your phone to ring aloud?
Phone	Not in Favorite, Contact List, Occasionally Calls	If you were at work in a meeting and someone in your **contact list** called, would you want your phone to ring aloud?
Phone	Not in Favorite, Not in Contact List, Few (if any) Calls	If you were at work in a meeting and someone **not in your contact list** called, would you want your phone to ring aloud?
SMS	Favorite List, Contact List, Frequently Texts	If you were at work in a meeting and someone in your **favorites list** texted you, would you want your phone to beep aloud?
SMS	Not in Favorite, Contact List, Occasionally Texts	If you were at work in a meeting and someone in your **contact list** texted, would you want your phone to beep aloud?
SMS	Not in Favorite, Not in Contact List, Few (if any) Texts	If you were at work in a meeting and someone **not in your contact list** texted, would you want your phone to beep aloud?
Calendar	Repeating Meeting	If you were at work in a meeting and a **repeating meeting** was about to start, would you want your phone to beep aloud to remind you?
Calendar	Non-repeating Meeting	If you were at work in a meeting and a **non-repeating meeting** was about to start, would you want your phone to beep aloud to remind you?

Table 2. Eight questions were asked about whether the user's phone should ring in a meeting at work. Prior to taking the survey, participants were given definitions of the notification contexts to help them answer the questions.

for each notification type. These situations were drawn from the sensor features in Table 1 and described participants' environments (*e.g.*, work or movie theater) or activities at the time of the interruption (*e.g.*, driving a car or relaxing at home).

Participants were given a short description of each of the situations and notification reason for the interruption, and were asked 1) if they would want audible notifications in that situation (interruption preference). Then they were asked to rate 2) their expected annoyance if the phone has the wrong volume setting (cost of misclassification) and 3) their expected annoyance if the phone asked which volume it should use (cost of asking). The questions were as follows:

1) In this situation, would you want your phone to ring out loud? Answer: Yes/No
2) How upset would you be if the phone did the opposite (rang when it should have been silent or *vice-versa*)? Answer: Likert scale 1 (no problem) to 7 (I would be very upset).
3) In this situation, how upset would you be if your phone asked what it should do if it didn't know? Answer: Likert scale 1 (no problem) to 7 (I would be very upset).

An example of the questions for a situation where a user is in a meeting at work is found in Table 2, Additionally, participants were able to list exceptions to their interruption preferences for each situation.

All combinations of situations, notification reasons and notification types (phone call, SMS message, or calendar alarm) were presented to participants. Because of the number of situations that would be necessary to train a classifier, we split the survey into twelve parts. Each participant was given the option of answering all questions through all 12 surveys, but was not required to complete them all. Before each survey, participants confirmed that they did receive each notification type the survey focused on (*e.g.*, only those who received calendar alarms filled out the calendar surveys).

Participants

Participants were recruited through a Carnegie Mellon participant recruiting website to complete the online surveys. We are interested in both within-subject differences across notification types, as well as between-subject differences for each situation. In total 44 participants took all 12 surveys and 50 more participants took subsets of the surveys for an average of 69.25 participants per survey. Sixty-five out of 94 participants reported that they were students. The rest reported jobs such as cashier, machine shop manager, photographer, and administrative assistant. The average age of the participants was 25.27 with standard deviation 6.3.

Approximating Participants' Costs

We received a total of 9219 responses to our surveyed situations questions and analyzed the proportion of participants who wanted audible notifications for each notification type (calls, SMS messages, or calendar alarms), situation, and notification reason to understand interruption preferences. We found that participants had *very* different interruption preferences for each type of notification, which is contrary to current phone settings that only allow a single phone volume for all notification types. For example, at work, 45% of participants wanted calendar notifications during meetings compared to 7% on average who wanted phone calls or text messages in the same situation (Figure 1). Only 35% of participants wanted to receive phone calls at work, but more wanted text messages, especially from those on their favorites list.

Participants noted that, currently, they often kept their phone on vibrate rather than silent or loud volume because of these situational and notification type differences. One participant said that they prefer to err on the side of caution when it comes to phone volume and "I can find the time to check the onscreen message if I'm not too busy" rather than listening for an audible notification. When they had to decide on a loud or silent volume setting, participants often responded that they would not want their phone to ring "unless it was a family emergency" or "unless I'm getting a ride from that person." These exceptions are hard to enumerate and predefine and indicate a need to use experience sampling to capture preferences *in situ*.

In order to be able to collect these *in situ* responses, we use their surveyed costs of misclassification and asking. Participants reported varying costs of misclassification responses on the Likert scale from 1-7 (mean 4.3, s.d. 2.1). Participants responded nearly half of the time (4436/9219 responses) that they would have "No Problem" if their phone asked them for their preference (mean 2.6, s.d. 1.95). There was no particular situation where a majority of participants indicated that they would not be willing to answer. In fact, some participants indicated that they would always be willing to answer questions while others indicated there were situations when they

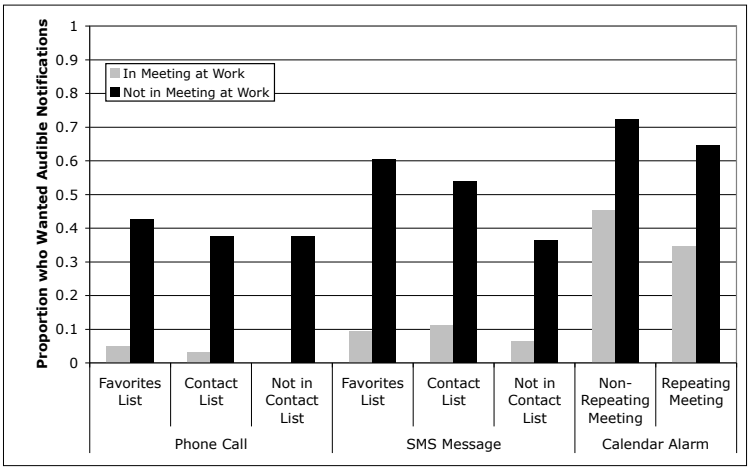

Figure 1. Participants varied greatly in their preferences for audible notifications at work when they were not in meetings, but mostly agreed that they should not receive calls or text messages during meetings.

never wanted to answer questions. These results show that a single cost model for all situations and/or all participants (from [13]) would likely interrupt many participants who indicated they did not want questions.

In order to approximate the costs for all situations in our phone app, we created artificial but plausible sensor values for each of the features in our application. Then, we used those sensor values to train a linear regression (easily computable on a phone) with the surveyed Likert ratings. For example, in order to model situations in the car, we averaged sampled accelerometer, microphone, and GPS values collected while driving with phones and labeled it with the corresponding Likert rating. For any new sensor data, the linear regression model will predict the cost of asking and misclassifying. Our linear regression models varied in their ability to capture each participant's predicted asking costs, as measured by the R^2 test, but overall was successful for such a simplistic model. Because we used only the features in Table 1 and did not use complex features, our cost approximations are easy to calculate on phones but may not always be predictive. Some of our linear regressions had R^2 values near 1; others were only about 0.3 (mean 0.65, s.d. 0.15).

Based on these findings and analysis, our phone volume application will need to learn a separate preference classifier (and use a personalized cost model) for each notification type and each participant.

6 Learning Interruption Preferences using Experience Sampling

In order to understand the impact of personalized cost models on the usability of experience sampling and the accuracy of the resulting preference models, we

designed a four-week experiment. Participants in the study were given our phone application, which learned their volume preferences and *actually* changed the volume of the phone based on learned classifiers. The application used one of three experience sampling algorithms - random, uncertainty, or decision-theoretic sampling – which asked them about their interruptibility preferences for each of the notification types, and used those preferences to build the volume classifiers.

Study Design and Procedure

Twenty of the survey participants who filled out all 12 surveys and had Android version 2.0 or higher phones were recruited to participate in our study to learn their phone volume preferences. Participants were asked to train their application, providing their volume preferences when asked, for two weeks and then test the resulting models for another two weeks, each night filling out surveys about the accuracy of the application and their current annoyance with either the questions or the volume changer itself. Participants were randomly but evenly assigned to one of four conditions – including two for decision-theoretic sampling – which determined when to ask for their preferences for phone volume when new notifications arrived:

- Random Sampling
- Uncertainty Sampling
- Decision-Theoretic Sampling
- Decision-Theoretic Sampling with Notification Reason

Because user preferences varied so greatly across participants, we did not test Decision-Theoretic (DT) sampling with a single cost model. Additionally, we do not test Kapoor's DT-dynamic condition (shown to be most accurate in highly changing domains) because we assume that users' preferences remain constant over the four weeks of the study. However, we did find in our surveys that the *reason* for the notification (*e.g.*, who is calling or whether the meeting is regularly scheduled) is a feature that users often use to determine whether they want an audible notification. We test the accuracy of preference classifiers that use this additional feature versus ones that do not, but do not test its use in experience sampling because the identity of the caller should not affect the cost of answering a question. The two DT techniques asked using the same algorithm.

Our volume changing application was loaded on each participant's phone, with a parameter file indicating which experience sampling technique to use and the linear regression cost models that were calculated from the participant surveys. Participants were told about the features that the application monitored and that it logged the features of each incoming notification, the classifier's prediction, and labels into a text file that we would collect once the study was complete. In addition to answering the application's questions, they were asked to fill out nightly online surveys on their phone about the accuracy of the model each day as well as the application's usability.

Participants were asked to keep the application running at all times during the 4 weeks of the study and were notified via email if the application quit at any time. After two weeks, the application automatically switched from training mode, which asked users for preferences but did not change the phone volume, to testing mode, which used the prediction to turn on or off the volume of the phone for each type of notification. One participant left the study after the training phase because of a family

emergency that required her to hear her phone all the time. After four weeks, researchers paid the participants $80, removed the application and collected the logs that were written to the phone over the course of the study.

Measures and Analysis

We measure four dependent variables: the number of questions asked, the accuracy of the classifier (collected each night over the 4 weeks) and the annoyance of both the asking and misclassification. The classifier accuracy is measured by comparing the classifier's predictions and the user's actual preferences collected from nightly surveys. We compare the experience sampling techniques using a repeated measures ANOVA of the accuracy, number and timeliness of responses over time. We collected annoyance ratings in the nightly surveys, but because participants did not have any other condition to compare to, they all rated their application as usable. Instead, we asked participants during their final interviews to recall specific situations when their application interrupted them, when the volume was incorrect as well as any other general impressions that they had about the application. We used these findings to distinguish the different sampling techniques.

Results

Overall, we found our approximated cost models had a significant effect on the number of questions that participants were asked and the usability and accuracy of the application. Participants in both decision-theoretic conditions reported that they were overall very satisfied with the timeliness of their questions and the resulting models were more accurate for most of the participants compared to the participants in random and uncertainty sampling conditions. We find that decision-theoretic participants who predicted they would have high interruption costs had lower accuracy because they were asked fewer questions, but that we can use participants' survey results to add more training examples and increase the accuracy.

Number and Timeliness of Questions

Participants received an average of 285 (min 32, max 717) phone calls, SMS notifications, and calendar alarms during the 14-day training period and received an average of 13 (s.d. 9.1), 41 (s.d. 59), and 3.2 (s.d. 5.8) questions respectively over the same period of time. Participants received far more SMS messages than phone calls and calendar alarms and the number of questions about them reflects this difference.

We compared the number of questions that participants received in each condition of the study for each type of notification (phone call, SMS message, calendar alarm) using a repeated measures test to understand whether the number of questions decreased over time and differed between conditions. We found that, for phone calls, both day of training ($F[13,195] = 4.67$, $p < 0.01$) and condition ($F[3,15] = 4.95$, $p = 0.01$) played a role in the number of questions participants received, but there was no interaction effect ($F[39,195] = 1.0$, $p > 0.05$). For SMS messages, there was high variability in the number of questions by participant mainly because some participants received many more text messages than others so we found that there was only a significant effect of day of training on the number of questions ($F[13,195] = 3.55$, $p < 0.01$). There were no significant effects on the calendar alarms as all participants

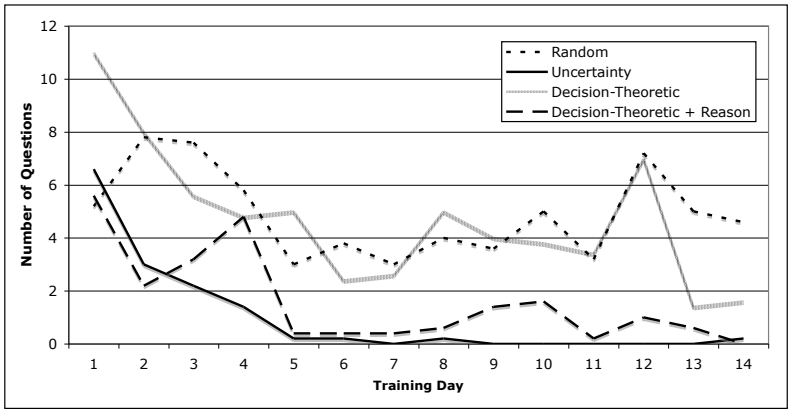

Figure 2. As the classifier uncertainty decreased through training, the number of questions decreased for Uncertainty and both Decision-Theoretic conditions. However, it did not decrease for Decision-Theoretic participants who said they were willing to answer more questions to increase accuracy.

received very few questions to learn an accurate classifier. Next, we analyzed the specific effects that the training day and experimental condition had on the number of questions.

A Tukey HSD test on the day of training for each of the phone and SMS messages showed that participants received statistically significantly more questions on days 1 and 2 (mean phone 2.33, SMS 6.96) compared to each of days 5-14 (all phone means less than 1.0 questions per day, SMS means less than 2.5). After day 2, the number of questions decreased for both phone and SMS notifications (Figure 2). The drop in notifications in the random condition is not significant.

Interestingly, a Tukey HSD test on the experimental condition for phone calls showed that the Decision-Theoretic Sampling resulted in a statistically higher number of questions (mean 1.6 questions per day) compared to Uncertainty sampling (mean .47 questions) and Decision-Theoretic with Notification Reason (mean 0.65 questions). There was no statistical difference between Random sampling (mean 0.96) and any other condition. Because we expected the two Decision-Theoretic sampling conditions to have similar results, we investigated this anomaly further. We found that 4/5 participants in the Decision-Theoretic condition reported low estimated costs of asking - each had an average cost of less than 4 out of 7 – compared to only 2/5 with low costs of asking in the DT + reason condition. When we add an extra independent variable representing a binary high or low cost of asking in our analysis, we find (as expected) that participants in both Decision-Theoretic conditions who indicated they had a low cost of asking were asked statistically significantly more questions per day compared to those with a high cost - on average 1.45 compared to 0.52 ($F[1,6] = 6.51$, $p < 0.05$). This cost accounts for the differences in the Decision-Theoretic conditions.

Despite the higher number of questions for 6 out of 10 of the decision-theoretic condition participants, all participants in both DT conditions reported that they were very satisfied with the timeliness of the experience sampling questions. Many participants in the random and uncertainty sampling conditions said they “eventually

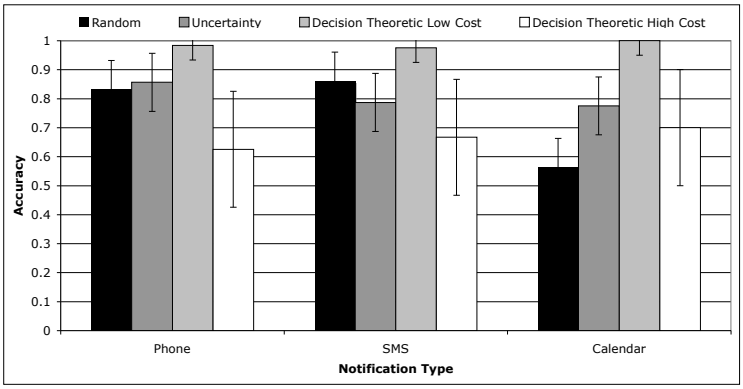

Figure 3. Participants with low costs of asking in Decision-Theoretic conditions had the highest accuracy classifiers for each notification type (mean 0.99, 0.97, 1.00 respectively). Three participants in the two Decision-Theoretic conditions had high costs of asking because they were not asked enough questions to create accurate classifiers.

got used to the questions" but were annoyed by them before that. This indicates that our personalized models had the effect we intended, in reducing the number of questions when users had high interruption costs and asking at more appropriate times for all participants including those who received questions everyday.

Accuracy

Thirteen out of nineteen participants reported at the end of the study that they were happy with the accuracy of their application. Three requested to see the application in the Android app store to download again. The accuracies of the conditions were 0.83 (s.d. 0.1) for random sampling, 0.85 (s.d. 0.1) for uncertainty, 0.85 (s.d. 0.23) and 0.9 (s.d. 0.21) for decision-theoretic without and with notification reason respectively. The difference in accuracy between conditions is not statistically significant. Although participants indicated that notification reasons were important in determining their volume preferences, classifiers trained with these extra features had the same accuracy as those trained without them.

We combine the decision-theoretic conditions to show the differences in accuracy between the 6 participants with low costs of asking compared to the 4 with high costs (Figure 3). Three of the four high cost participants in the decision-theoretic conditions had accuracy lower than 0.8 for phone calls and text messages (mean 0.66, s.d. 0.16) compared to an average accuracy of 0.98 for participants with low cost of asking. Our decision theoretic samplers with approximated cost models are capable of very high accuracy when users are willing to answer questions. The experience samplers with high costs could not identify enough situations to ask but maintain usability, and the lack of labeled training data resulted in low accuracy for these classifiers.

In an effort to create more accurate classifiers for these 3 participants with high costs of asking, we examined the participants' survey responses to understand if their predictions were accurate. One participant's schedule and corresponding volume

preferences changed after providing survey responses and the training period. Because the participant did not anticipate these changes, a classifier trained on these survey responses could not have been accurate. For the two other participants, however, the survey responses would have increased the classifier accuracy. For example, one participant's classifier turned the volume off in the evenings when he was relaxing causing him to miss many phone calls and text messages. The decision-theoretic experience sampler never asked for his preferences in this situation in order to preserve usability. If the classifier had used his single response to the survey – that he did want his phone to ring and beep - his accuracy would have increased from 75% to over 92%. We conclude that we can use participants' survey responses as additional training data for inaccurate classifiers.

In summary, participants in both decision-theoretic conditions reported that they were very satisfied with the timeliness of the questions they were asked compared to the participants who received random and uncertainty sampling. The resulting models were more accurate for most of the participants in these conditions as well. However, some decision-theoretic condition participants received fewer questions than others due to their high cost models and this affected the accuracy of their classifiers. We find that in most cases we can use participants' survey responses to increase the accuracy of the classifiers when they have high interruption costs.

7 Discussion

We have compared the accuracy and usability of three different experience sampling algorithms and found that our decision-theoretic sampling with personalized cost models was most accurate and asked questions at the most appropriate times. Next we address some of the participants' difficulties and suggestions that they made after using our application for four weeks.

Survey Responses as Approximate Interruption Models

Our main assumption in using experience sampling was that participants have difficulty predicting their preferences in advance, but that we could use these predictions to approximate interruptibility. We found that overall, this approach was very successful in maintaining very high accuracy while limiting the interruptions at inappropriate times. Thirteen participants also preferred answering questions over time and thought their *in situ* responses were more accurate than their survey predictions, and three thought a combination of surveys and experience sampling would be most accurate. Participants who preferred the questions reported that they liked that "it prompted me because it made me think of what I'm doing now" and that is hard to do before using it. This finding mirrors other experience sampling findings that participants answer more accurately in the moment, but contradict other HCI arguments that users should not be interrupted to train classifiers [5].

Participants who received few questions resulting in poor accuracy said that they would have been willing to answer more questions if they were told that their costs affected the classifier accuracy. A visualization showing the costs of interruption and the average resulting accuracy could allow participants to see the results of their

tradeoffs concretely before using the application. Future work is needed to evaluate whether such visualizations are understandable and affect users' predicted interruption costs.

Volume Preferences Change over Time

We also found that participants' volume preferences changed throughout the study. Participants started new routines in the middle of the study – either starting classes or their kids started new activities. Because they had already started or even completed the training of their classifier, they could not reverse or change the previous responses and their classification accuracy suffered. Participants reported at the end of the study that they wanted to change or start the training over because they had such different preferences. As a result, we argue that applications should be able to employ lifelong learning techniques such as forgetting [12] or at least allow users to change their preferences to maintain accuracy as they drift or schedules change over time.

Some participants reported that there were sometimes unexpected circumstances that their classifiers could not handle. For example, some students were willing to receive audible text message notifications in class, but they did not want them on days when they had exams. Participants were not thinking about exams during their classes when they answered questions during training but had no way of changing the classifier's prediction on that particular day. For circumstances like these, we suggest the use of an override button to force the phone volume to be at a set level for a set amount of time. This button could also give users a better sense of control about their phone notifications if they are uncertain about what their classifier will predict.

Need for Intelligibility

Intelligibility became a big issue for our participants as their phone applications transitioned to testing mode. Uncertain of what their classifiers had learned, many participants emailed the authors asking how to find out what they should do if their classifiers learned the wrong thing. We argue that offering a "what if" interface (in which participants could have set different features to see the resulting prediction [19]) could have reduced some of the uncertainty and lack of control that users felt during testing mode in our study. Users could check that their classifiers make accurate predictions and provide extra examples for those situations in which it does not.

Participants also requested an interface in which they could see and change the rules that were generated for their classifier, especially if it was consistently wrong about a set of situations. We found that the classifiers were most overconfident in the uncertainty sampling condition and if users could adjust the classifiers during both training and testing phases, it could have reduced the potential errors and helped identify opportunities for the sampler to request more preference data. One student participant, for example, said that his classifier learned to turn his ringer off too early in the evening and this could have been easily resolved if he could have set the time feature. However, it is often difficult to show the rules of a classifier in a simplified way. More work is needed in order to understand what information users really want to know about their classifiers and what is too complicated or not important to show.

8 Conclusion

In this work, we have presented a phone volume application that classifies users' interruptibility and adjusts the volume accordingly. Because users have difficulty predicting their interruption level when they are not actually in the asked-about situation, we introduce an experience sampling technique that asks users to predict their costs of interruption and uses these predictions to approximate a cost model and determine when to actually ask for preferences. We deployed our volume application to learn users' preferences over 2 weeks and test the resulting classifier for 2 weeks, comparing the usability and accuracy of our experience sampling technique against other traditional techniques.

We find that our method for approximating interruptibility improves the timeliness of questions asked during experience sampling. Additionally, we find that 7 out of 10 participants in the decision-theoretic condition reported very high accuracy with few or no errors while testing their classifier for two weeks. However, we find that the cost models, while effective at improving usability for all users, actually harmed accuracy for the remaining three Decision-Theoretic participants by asking too few questions and thus we caution using this technique for users with high asking costs. Finally, for these high asking cost users, we show that their initial predictions from the surveys can be used to create more accurate classifiers than the experience sampling could. Future work is needed to increase the intelligibility of the classifiers and the cost models to give users more control over their phone. Additionally, more work is needed to understand how phone preferences change over time and how we can develop classifiers to maintain high accuracy during through lifelong learning.

References

1. Cohn, D., Atlas, L., and Ladner, R. Improving Generalization with Active Learning. Machine Learning, 15(2):201–221, 1994.
2. Csikszentmihalyi, M. and Larson, R. Validity and Reliability of the Experience Sampling Method. Journal of Nervous and Mental Disease, 175(9):526–536, 1987.
3. Czerwinski, M., Horvitz, E., and Wilhite, S. A Diary Study of Task Switching and Interruptions. In Proc. of CHI 2004, 175–182, 2004.
4. Donmez, P. and Carbonell, J. G. Proactive learning: cost- sensitive active learning with multiple imperfect oracles. In Proc. of the Conference on Information and Knowledge Management (CIKM), 619–628, 2008.
5. Faulring, A., Myers, B., Mohnkern, K., Schmerl, B., Steinfeld, A., Zimmerman, J., Smailagic, A., Hansen, J., and Siewiorek, D. Agent-assisted Task Management that Reduces Email Overload. In Proc. of IUI 2010, 61–70, 2010.
6. Fogarty, J., Hudson, S. E., Atkeson, C. G., Avrahami, D., Forlizzi, J., Kiesler, S., Lee, J. C., and Yang, J. Predicting Human Interruptibility with Sensors. ACM Trans. Computer-Human Interaction 12(1):119-146, 2005.
7. Ho, J. and Intille, S.S. Using Context-Aware Computing to Reduce the Perceived Burden of Interruptions from Mobile Devices. In Proc. of CHI 2005, 909–918, 2005.
8. Horvitz, E. Principles of Mixed-Initiative User Interfaces. In Proc. of CHI 1999, 159-166, 1999.

9. Horvitz, E. and Apacible, J. Learning and Reasoning about Interruption. In Proc. of the International Conference on Multimodal Interfaces (ICMI), 20-27, 2003.
10. Horvitz, E., Koch, P., and Apacible, J. BusyBody: Creating and Fielding Personalized Models of the Cost of Interruption. In Proc. of Conference on Computer Supported Cooperative Work (CSCW), 507-510, 2004.
11. Intille, S. S., Rondoni, J., Kukla, C., Anacona, I., and Bao, L. A Context-Aware Experience Sampling Tool. Extended Abstract in Proceedings of CHI, 972-973, 2003.
12. Kapoor, A. and Horvitz, E. On Discarding, Caching, and Recalling Samples in Active Learning. In Proc. of Uncertainty in Artificial Intelligence (UAI), 209-216, 2007.
13. Kapoor, A. and Horvitz, E. Experience Sampling for Building Predictive User Models: a Comparative Study. In Proc. of CHI 2008, 657-666, 2008.
14. Kern, N. and Schiele, B. Towards Personalized Mobile Interruptibility Estimation. In Proceedings of the International Workshop on Location- and Context-Awareness, 134-150, 2006.
15. Khalil, A. and K. Connelly. Improving Cell Phone Awareness by Using Calendar Information. In Proc. of International Conference on Human-Computer Interaction (INTERACT), 588-600, 2005.
16. Krishnan, M.V. Availability and Mobile Phone Interruptions: Examining the Role of Technology in Coordinating Mobile Calls. Masters Thesis MCS-2008:18, Blekinge Institute of Technology, March 2008.
17. Lewis, D. D. and Catlett, J. Heterogeneous Uncertainty Sampling for Supervised Learning. In Proc. of the International Conference on Machine Learning (ICML), 148–156. 1994.
18. Lewis, D. D. and Gale, W. A. A Sequential Algorithm for Training Text Classifiers. In Proc of the Conference on Research and Development in Information Retrieval (SIGIR), pages 3–12, 1994.
19. Lim, B. Y., Dey, A. K., and Avrahami, D. Why and Why Not Explanations Improve the Intelligibility of Context-Aware Intelligent Systems. In Proc. of CHI 2009, 2119–2128, 2009.
20. McFarlane, D. Coordinating the interruption of people in human-computer interaction. In Proc. of International Conference on Human-Computer Interaction (INTERACT), 295-303, 1999.
21. Milewski, A.E. and Smith, T.M. Providing Presence Cues to Telephone Users. In Proc. of the Conference on Computer Supported Cooperative Work (CSCW), 89-96, 2000.
22. Scollon, C., Kim-Prieto, C., and Diener, E. Experience Sampling: Promise and Pitfalls, Strengths and Weaknesses. Journal of Happiness Studies, 4:5–34, 2003.
23. Shadbolt, N. and Burton, A. M. The Empirical Study of Knowledge Elicitation Techniques. SIGART Bulletin, 108:15–18, 1989.
24. Schmidt, A., Takaluoma, A. and Mäntyjärvi, J. Context-Aware Telephony Over WAP. Personal and Ubiquitous Computing, 4(4): 225-229, 2000.
25. Siewiorek, D.P, Smailagic, A., Furukawa, J., Krause, A., Moraveji, N., Reiger, K., Shaffer, J. and Wong, F.L. SenSay: a Context-Aware Mobile Phone. In Proc. of the International Symposium on Wearable Computers, 248–249, 2003.
26. Stumpf, S., Rajaram, V., Li, L., Burnett, M., Dietterich, T., Sullivan, E., Drummond, R., and Herlocker, J. Toward harnessing user feedback for machine learning. In Proc of IUI 2007, 82–91, 2007.
27. Toninelli, A., Khushraj, D., Lassila, O., and Montanari, R. Towards Socially Aware Mobile Phones. In Proc. of the Social Data on the Web Workshop, 2008.