

Predicting time-sharing in mobile interaction

Miikka Miettinen · Antti Oulasvirta

Received: 14 July 2006 / Revised: 23 February 2007 / Accepted in revised form: 3 June 2007 /
Published online: 28 July 2007
© Springer Science+Business Media B.V. 2007

Abstract The era of modern personal and ubiquitous computers is beset with the problem of fragmentation of the user's time between multiple tasks. Several adaptations have been envisioned that would support the performance of the user in the dynamically changing contexts in which interactions with mobile devices take place. This paper assesses the feasibility of sensor-based prediction of time-sharing, operationalized in terms of the number of glances, the duration of the longest glance, and the total and average durations of the glances to the interaction task. The data used for constructing and validating the predictive models was acquired from a field study ($N = 28$), in which subjects performing mobile browsing tasks were observed for approximately 1 h in a variety of environments and situations. The predictive accuracy achieved in binary classification tasks was about 70% (about 20% above default), and the most informative sensors were related to the environment and interactions with the mobile device. Implications to the feasibility of different kinds of adaptations are discussed.

Keywords Time-sharing · Attention · Multitasking · Interruptions · Mobile interaction · Mobility · Classification · Predictive models · Bayesian networks

1 Introduction

Human-computer interaction in the era of modern personal and ubiquitous computers is beset with the problem of fragmentation of time between multiple tasks (Adamczyk and Bailey 2004; Card and Henderson 1987; González and Mark 2004; Ho and Intille 2005;

M. Miettinen (✉) · A. Oulasvirta
Helsinki Institute for Information Technology (HIIT),
University of Helsinki and Helsinki University of Technology, P.O. Box 9800,
FIN-02015 TKK, Espoo, Finland
e-mail: miikka.miettinen@hiit.fi

A. Oulasvirta
e-mail: antti.oulasvirta@hiit.fi

Hudson et al. 2003; Jameson et al. 1999; McFarlane and Latorella 2002; Tamminen et al. 2004). Users must switch back and forth, temporarily leaving some tasks on hold or slowing them down. Thus there are nearly always several unfinished, simultaneous, successive, and overlapping tasks. For the user, the cognitive challenge is to plan and execute the sharing of time in such a way that the length and frequency of interactions with the device, as well as the timing of shifts between tasks, are in balance with the demands of the situation. Figure 1 illustrates the variety of time-sharing patterns a 30-s time window can exhibit when the user is mobile.

The psychological notion of *time-sharing* refers to performing two or more tasks simultaneously by sequentially handling information from perceptual channels (Wickens 1984).¹ Time-sharing behavior is pervasive and has been observed repeatedly in various settings (see e.g., González and Mark 2004; Oulasvirta et al. 2005; Wikman et al. 1998). Because the allocation of processing time to an interaction task is an essential precondition for its advancement, the user's time can be seen as a kind of *resource*—a resource that is abundant at times and scarce at others. The frequency, length, and timing of interaction give character to this resource. When forced into a situation of inappropriately timed, too long, or too short shifts, progress in the task is compromised. Moreover, people constantly interleave tasks and subtasks across psychological modalities (Jameson and Klöckner 2005; Vera et al. 2004).

Sadly, however, present-day computers are ignorant of the way users share time between the user interface and the environment. There are several examples, given in the next subsection and elaborated throughout the rest of the paper, demonstrating adaptations that could address this problem, assuming that real-time prediction of time-sharing was possible. To critically assess the feasibility of such adaptations, we present a wizard-of-oz feasibility study (Hudson et al. 2003; Fogarty et al. 2005) looking at automatic prediction of time-sharing based on sensors of varying degrees of sophistication. Rather than just pooling results from empirical work, we contribute to the field by examining the possibility of using predictive models for the development of real-world computing and communication applications (see Horvitz and Apacible 2003). Our analysis and modeling focuses on the sharing of time in mobile interaction, one of the increasingly more important domains of user modeling and adaptation (Kobsa 2001). For these ends, four questions are addressed in the paper:

1. *Information needs.* Of all possible quantifications of time-sharing, which ones are useful for the proposed adaptations?
2. *Phenomenon.* What regularities are there in the users' time-sharing behavior such that they might be captured by available and foreseeable sensors?
3. *Engineering.* What kind of sensors and computational models are needed for successful prediction of time-sharing?
4. *Feasibility.* What is the overall feasibility of predicting time-sharing and what kind of adaptations are realistic?

1.1 Consequences of suboptimal time-sharing strategies to interaction

The notion of time-sharing has originated from analyses of various domains where time-sharing is an issue of safety—like driving, piloting, air traffic control, and radar operation (Salvucci 2005). Cognitive models of the scheduling of cognitive, perceptual, and motor operations in interaction tasks explain why people often select suboptimal scheduling strategies that have to be continuously corrected as the tasks proceed (Fu and Gray 2004;

¹ Please note that in this paper time-sharing does not refer to the sharing of CPU time among multiple users, but the sharing of the *user's* time among multiple tasks.

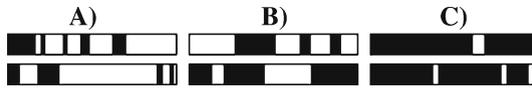


Fig. 1 Examples of gaze deployment patterns demonstrating that users allocate time to an interaction task in many different ways when mobile. The boxes represent 30-s time windows, within which the black bars are glances to the interaction task (mobile browsing) and the white bars to the environment. (A) Sporadic, short glances at irregular times; (B) alternating, approximately equally long glances to the device and the environment; (C) concentrated attention to the device predominates. Adapted from the dataset of Oulasvirta et al. (2005).

Gray and Boehm-Davis 2000). In the following, we summarize a number of problems that have been observed and their consequences to interaction.

First, *long interruptions* may result in slowdowns in user response times to events, or missing them entirely and thus to even errors. Moreover, attentional displacement (looking off the target when returning to the main task (Wikman et al. 1998), and memory interference (Glanzer et al. 1981) may occur. On the other hand, *too frequent switches* lead to a poor level of sampling/processing quality due to build-up of switch costs (see Monsell 2003). Third, *too long shifts* away from a task result in long periods of unawareness over other events, increasing uncertainty over them. In driving, for instance, long glances away from the primary task are risky and may lead to fatal consequences (Wikman et al. 1998). Finally, and related to the third one, investing *all time* to just one task ensures maximal resources for its processing, but compromises other tasks.

Thus, the duration of uninterrupted time dedicated to a task, the relative proportion of such periods, and their frequency are all associated with different consequences to the user's ability to interact. In our work, we have assessed four related metrics as characteristics of time-sharing.

1.2 Existing and envisioned adaptations

Noting the importance of the problem, several papers have been published recently that present adaptations presuming sensor-based information about the user's time-sharing behavior (although this term is rarely used). A distinction is here made between four categories of potentially useful adaptations:

1.2.1 Optimization in presentation

First, optimization in presentation means changing the format or style of a UI (but not the content or functions) according to capacities implied by the user's current time-sharing pattern. One explored example in this category is the facilitation of visual search when the user's time is scarce (Mäntyjärvi and Seppänen 2003). Text on display is summarized and enlarged when the user is "in a hurry". A more speculative example is adaptively provided task-resumption cues to overcome displacement after interruptions (Altmann and Trafton 2004). Here, longer glances away from the application could be taken as an indication of being interrupted, and the cues could be presented to help the user to remember where she was left. (However, experiments have shown that the design of efficient cues is not trivial; e.g., Cutrell et al. 2001.) It might also be appropriate to make presentations richer and more detailed when the user is not interrupted but is predicted to allocate more time to the task.

1.2.2 Preparation of resources

Second, preparation of resources means the execution of resource-intensive preparatory operations (network, memory, or computational) when the user is *not* paying attention to the device (Salovaara and Oulasvirta 2004). For example, being able to predict breaks in interaction during mobile browsing would enable the device to identify suitable moments for proactive pre-caching of web pages in order to reduce the delays caused by a slow network connection.

1.2.3 Adaptation of functionalities and timing

Third, adaptation of functional properties like timing could provide new opportunities for interaction or inhibit potentially disruptive ones. Timing of functions is a form of adaptation that is about “saying the right thing at the right moment” (Fischer 2001). Mixed-initiative UIs, for instance, could use information on the user’s allocation of processing time to decide when to take turns in negotiation (Horvitz 1999a). Communication applications could use information on the user’s time-sharing as an index of availability, and postpone or redirect messages accordingly (Horvitz et al. 1999; Fogarty and Lai 2004; Fogarty et al. 2005). The attentive UIs enterprise has envisioned functionalities being launched according to the user’s concentration on a specific target (Vertegaal 2003). Finally, interruptions (e.g., pop-ups and messages) could be presented between tasks rather than during them (Ho and Intille 2005), or abrupt changes in time-sharing behavior could be utilized for delivering messages.

1.2.4 Social awareness cue

Fourth, in awareness systems, information about time-sharing could be used as a cue indicating availability. In general, awareness cues are representations of a remote user’s state or situation based on automatic interpretations of sensor data. They enable the users to orient to a remote person in order to align ongoing activities and trigger new ones (Dourish and Bellotti 1992). For example, the work by Begole and colleagues looked at visualizations of a worker’s rhythms in time as a means for informing colleagues in the office of one’s presence (Begole et al. 2003). In mobile awareness systems, even crude cues are known to be interpreted flexibly and creatively according to situational demands (Oulasvirta et al. 2007). For example, the manipulation history cue (“the user has/has not used the phone during the last 15 minutes”) is often interpreted as an indication of availability for communication and messaging, feedback on whether or not the other has received a message, proximity to the phone, interruptability, inability to respond to messages and being asleep. We believe that more detailed cues such as “user is not paying attention to the phone at all” or “user is concentrating on the phone” might be useful as well, informing communication decisions and in general supporting understanding of what a particular person is doing at the moment. (See also Fogarty and Lai 2004; Fogarty et al. 2004.)

1.3 Approach: a wizard-of-oz feasibility study

The practical motivations of this study are based on the idea of a *service* running on a computer, here a mobile device, which would provide applications with information about the user’s time-sharing. The service would receive data from a variety of sensors as input, and respond to queries from applications by returning either its single best guess of the value of a time-sharing variable or a probability distribution over all possible values.

As will be argued below, numerous factors affect time-sharing besides those that could be monitored with an attainable collection of sensors. Since people can interleave tasks in many different ways, also across modalities, the problem is far from trivial. However, it seems plausible that there might be systematic dependencies between observable variables and time-sharing, and capturing such dependencies successfully in a statistical model might enable us to predict time-sharing with sufficient accuracy to meet the needs of relevant applications.

We approach this challenge by means of a *wizard-of-oz feasibility study* (Hudson et al. 2003). The idea is to take realistic data and build predictive models on the basis of sensors that are partially *simulated* by human codings of the data (thus the term “wizard-of-oz”). The type of prediction addressed in this paper is that of unobtrusive “keyhole prediction” where the user does not actively contribute to the prediction process (Carberry 2001). Both the specific findings and the improved overall understanding resulting from this kind of a study provide valuable guidance to developers interested in the feasibility of various adaptations to time-sharing.

1.4 Related research: online detection of users’ interruptability and attentional state

The importance of attention as a limiting factor has been recognized, and several previous studies have focused on modeling its relation to observable data. The operationalized variable varies from one study to another. Although the results are not directly comparable to ours, the previous studies illustrate complementary perspectives and clarify the relationship of our work to other related efforts.

Fogarty and colleagues (2005) explored the interruptability of programmers working with desktop PCs in an office setting, aiming to predict the level of interruptability based on software sensors that monitored low-level input events at the user interface. They first gathered training data by logging the actions of the users and observing their response times to randomly presented notifications. A collection of sensors was then created to extract higher-level features from the log data, and the response times were clustered in three groups representing interruptability in the corresponding situations. Interruptability was defined in terms of the response time to an abrupt notification. The resulting data set was used for creating a classifier that predicted the level of interruptability on the basis of the observable actions of the users, and the predictive performance of the classifier was evaluated to assess the overall feasibility of optimizing the timing of interruptions. In terms of the general approach, the study is similar to ours, but the activities of the users, the environment in which they took place, and the conceptual approach to quantifying the “available resources of the user” are different.

Ho and Intille (2005) investigated the interruptability of mobile users based on the idea that certain moments are more appropriate for delivering messages than others, and an adaptive application could try to identify the appropriate moments rather than tracking the degree of interruptability continuously. The hypothesis was that users are more receptive to interruptions upon a transition in posture or movement. The participants carried a PDA equipped with accelerometers while performing the normal activities of a workday. Messages were delivered to the device both at activity transitions and at random times, and the participants rated the perceived burden of each interruption. The results indicate that messages delivered at transitions were in fact considered somewhat less disturbing. The study is similar to ours in the sense that the participants were interacting with a mobile device in a variety of natural settings. On the other hand, the interaction involved in acknowledging a short message is quite different in nature from mobile browsing, and the idea of concentrating interactions at activity transitions is not directly applicable to the needs of our research.

Other interesting studies concerned with the adaptation of notification flow for mobile users have been performed by Kern and colleagues (Kern and Schiele 2003; Kern et al. 2004). They proposed that a distinction should be made between personal and social interruptability. The former is the perceived cost of an interruption to the user, while the latter represents the cost to other people present in the social situation. The levels of personal and social interruptability define a two-dimensional space, which was mapped directly to a corresponding grid that specified the appropriate notification modality (including the possibility of omitting immediate notification altogether). In the first study (Kern and Schiele 2003), certain stereotypical situations (e.g., “walking in the street” and “conversation in a restaurant”) were assigned to specific regions of the space representing personal and social interruptability, and the feasibility of recognizing the situation based on the auditory environment, movement, and location of the user was evaluated with a working prototype. Interruptability was not modeled *directly* in terms of the sensor data, but was assumed to be fully determined by the situation. The second study (Kern et al. 2004) avoided this assumption by relying on explicit ratings of interruptability obtained from a sample of users, and presented a number of technical improvements in the construction of the model and the hardware platform.

Jameson and colleagues (2006) conducted a laboratory experiment where the participants were required to (a) speak quickly versus not (as an indicator of time pressure) and (b) navigate through a simulated airport terminal versus stand still. The objective was to assess the feasibility of detecting the resource limitations of the user from the speech signal. 70–80% accuracy was achieved (chance level 50%) in the presence of background noise, and the most useful sensor counted the number of syllables in an utterance.

Vertegaal (2003) explored the possibilities for adapting to the user’s attention in indoor environments with ubiquitous computers. In this setting, the multitude of devices results in conflicting demands, and modeling the user’s attention could enable more natural and convenient interactions. The proposed system evaluates the overall interruptability of the user, prioritizes the demands for attention, and chooses an appropriate device and modality for presenting notifications. In some cases, information about the user’s attention could also control the operation of a device directly, for example pausing a video automatically when the user is not watching it. Compared to our work, the problems addressed are somewhat different. Mobile users (at least currently) only interact with a single device, but the interactions take place in environments that are so complex and dynamic that resource limitations need to be considered.

Some of the most elaborate models to date for adapting to the attentional states of the user have been constructed at Microsoft Research (Horvitz 1999b; Horvitz et al. 1999, 2003; Horvitz and Apacible 2003). The work attempts to provide a foundation for both enhancing existing applications and creating new kinds of applications based on mixed-initiative computing. Although mobile devices are considered as part of a larger system, the focus is on office and home environments equipped with desktop computers. The proposed models rely on a wide variety of sensors monitoring the user’s activities, including gaze deployment, posture and movement, location, and interactions with computing devices. In addition, the goals and interests of the user as well as the contents of the messages being delivered are relevant in some applications. Several different constructs are proposed for describing the attentional states of the user. Some of these reflect the availability of attention or its specific target, while others are stereotypical situations that are assumed to determine the appropriate behavior of the application. The model computes a probability distribution for either the current attentional state or a future state, and a detailed utility function determines the trade-offs involved in each possible adaptation. In other words, the approach takes into account both the uncertainty in a particular interpretation of the situation and the potential costs and

benefits of the adaptations. Our work is also based on probability models, but we do not consider the needs of individual applications in detail. However, if the prediction of the time-sharing of mobile users turns out to be feasible and appropriate, the utility-based approach represents an important direction for future research.

The most essential distinguishing characteristic of our work compared to the studies presented above is the conceptualization of the user's available time as a *resource*, which constitutes a necessary prerequisite for the progress of interactions with the mobile device. This view, inspired by work in cognitive sciences (e.g., Simon 1971), maintains that the user's ability to handle interactions of varying duration and complexity depends to a large extent on the other demands of the situation. Successful prediction of variables describing time-sharing would enable adaptive applications to ask whether or not the user is capable of performing a particular task under the observed circumstances. This is a different question than whether or not the user feels interruptible. Therefore, our approach complements the studies that have modeled the interruptibility of mobile users independently of the required interactions, and is relevant to a wide variety of adaptations.

2 Towards modeling requirements: human strategies in time-sharing

This section presents empirical findings from cognitive psychology and human factors research. The objective is to gather "requirements" that will be addressed in the construction of the predictive models.

2.1 Internal and external constituents of time-sharing

Ideally, a person could perform several tasks simultaneously without additional costs. From a cognitive perspective, what leads to the sequential sharing of time is the presence of a resource competition situation where multiple tasks compete for limited resources. The *multiple resources theory* (Wickens 1984,2002) suggests that competition increases with the processing-difficulty and resource-similarity of the tasks. This notion is also relevant to mobile human-computer interaction where the competition is between *mobility tasks* (e.g., route planning, talking, waiting, estimating time-to-target, controlling personal space) and *interaction tasks*, which compete mainly for the visual and motor resources (Jameson and Klöckner 2005; Oulasvirta et al. 2005).²

Although one might easily think that external events are the root cause of diverted attention, time-sharing is actually largely driven by internal processes (Kushleyeva et al. 2005). Satisfactory time-sharing requires the ability to create and schedule future intentions, the facility to remember, maintain and prioritize them, and the ability to switch from carrying out one intention to another when needed (Burgess 2000). Internal control is necessary also because environmental feedback is not always available or reliable (Fu and Gray 2004; Salvucci 2005). In addition to time and resource costs of internal operations (Vera et al. 2004), a general *switch cost* poses perhaps the most important internal limitation to time-sharing (Pashler 1993). This cost ranges from tenths of seconds to a few seconds, the exact cost depending on many factors (Monsell 2003), and are arguably caused by two mental events: reconfiguration of the task set and interference from previous tasks. As will be discussed later, avoiding the accumulation of switch costs is an important aspect of time-sharing strategies.

² Jameson et al. (1999) uses the terms *environment-related* and *system-related* basically in the same meaning as our mobility and interaction tasks.

External constraints imposed by the task environment obviously play a role as well (Vera et al. 2004). Simply because of different constraints, differences in fragmentation should follow. In addition, orienting responses to abrupt events in the environment can break the top-down control of attention (Näätänen 1992). In mobile interaction, due to the presence of multiple tasks and events requiring attention, the span of continuous attention allocated to a single task is typically in the order of ten to few tens of seconds (Oulasvirta et al. 2005). By contrast, an observational study revealed that office workers' time is divided into spans of three minutes per task on average (González and Mark 2004).

2.2 Users' tactics and strategies in time-sharing

Despite serious cognitive limitations, people are able to do on-line interleaving of tasks fluently. To understand how, we review some tactics and strategies of time-sharing. Table 1 summarizes some of the findings discovered by the authors based on the data of Oulasvirta et al. (2005).

There are several reasons why these are at least locally rational strategies given the internal and external constraints. First, as argued, more switching leads to poorer overall quality of processing due to the accumulation of switch costs (Monsell 2003). Thus, there is a qualitative difference (in processing quality) between allocating time in a frequent and erratic manner versus in a continuous manner with few switches. In order to counteract this effect, people exhibit a tendency to continue performing a lower-priority task longer than optimal. This tendency implies that time-sharing must also be sensitive to *goal and task hierarchies* (Salvucci 2005). Related to this, task boundaries are natural places to switch (Adamczyk and Bailey 2004; Ho and Intille 2005; Miyata and Norman 1986) and people often resist switch-aways just before task boundaries. Moreover, due to the prioritization of tasks, not all tasks need to be immediately executed but are more easily postponed than others, this tends to happen when the workload increases. It is worth noting that time-sharing accuracy increases with practice. People can interleave their tasks with the maximum accuracy of 50 ms after extensive training (Pashler 1993).

Second, countering and reducing costs due to increasing uncertainty (over the events of the task environment) during long interruptions is important. *Elapsed time* in a task is found to be a good predictor of urgency to switch away to another task (Kushleyeva et al. 2005). Third, *pre-knowledge* of what is to be expected, in semantic memory, is used as a source for longer-term calibration of time-sharing. For example, when a metro train leaves the station, experienced travelers "preprogram" themselves to what is the end signal of the task, the announcement of the destination station or its visual characteristics observable from the windows. After this, only brief sampling is required, and unnecessary devotion of processing time to irrelevant stimuli can be avoided (Oulasvirta et al. 2005). The crucial role of longer-term tactics, strategies and plans suggests that time-sharing emerges in a longer-term span that is distinct from ephemeral actions and reactions to external events.

Given all task and cognitive constraints, users choose one strategy from the *space* of possible strategies determined by the constraints (Eng et al. 2006). Because of the psychological reality of situation-independent strategies and the contiguity of similar situations in the world, it is reasonable to consider the temporal dependencies between successive time slices in the predictive model. Taken together, these findings speak for looking at long enough time spans, implementing time-dependent sensors (e.g., elapsed time) and, somehow, modeling the user's pre-knowledge of the task.

Table 1 Time-sharing strategies, examples, and possible rationales

#	Strategy / Tactic	Example (from Oulasvirta et al. 2005)	Possible rationale
1	Withdrawing resources from a task of secondary importance	A participant slows down walking when interacting with the mobile browser application.	The secondary task taxed the resources needed for interaction (the main task).
2	Postponing task switch when the workload increases	A participant stops walking entirely when interacting with the mobile browser application	The secondary task taxed the resources needed for interaction (the main task), and the secondary task can be performed later on.
3	Avoiding frequent task-switching	A participant keeps gazing at the mobile device despite various environmental distractions.	Minimizes the accumulation of switch costs.
4	Resisting switching just before the end of a task or a subtask	In order to finish an almost completed interaction task, a participant slows down walking as he gets closer to the escalator that marks the end of the walking task.	Uniform tasks and subtasks are better cognitively managed than fragments.
5	Switching to tasks that have been on hold for a long time	A participant makes a short glance to the browser to see if the page loading state has finally changed. (Page loading typically took about 16 s in the experiment.)	Long periods of unawareness of the progress of page loading decrease overall task performance even though page loading is only a waiting subtask and does not directly contribute to achieving the task goal.
6	Calibrating switching to expectancies of future events	A participant sits down in bus and looks out of the window to estimate when it arrives to a given destination.	Enables preprogramming of attention to recognize only the end signal and to better ignore irrelevant stimuli in the environment. Reduces resources needed for monitoring the environment. Requires previous experience of the situation.
7	Resisting switching from tasks when a nodal event (an event breaking the current task and signaling the change of context or upcoming of a new task) is expected soon	A participant in a metro car approaching the target station keeps looking out of the window and prepares to leave the car.	Unawareness of a nodal event increases uncertainty. Preparing for the upcoming task switch is necessary for fluent action.

3 The prediction task

At a general level, we define the prediction task as follows:

Given the information provided by simulated sensors, compute the probability distributions of certain variables describing the user's gaze deployment pattern during the next 30 seconds.

In the rest of this section we define the prediction task in detail and explain the underlying rationale.

3.1 Targets of time-sharing

First, for reasons of simplicity, we chose to analyze the sharing of time between *interaction tasks* (implied by mobile browsing) and *mobility tasks* (implied by goal-oriented activities in the environment). By contrast, others before us have analyzed time-sharing between multiple tasks or targets, but the analysis has been limited to well-defined tasks the structures and modalities of which are known in advance to the researchers with significant accuracy in controlled conditions (Jameson and Klöckner 2005; Jameson et al. 2006; Vera et al. 2004). Our data was collected in non-controlled environments and thus the tasks were not known with comparable accuracy to us researchers—a situation typical of studies concerned with mobile devices that are used in various circumstances not known in advance.

3.2 Time window

Second, time-sharing is a phenomenon taking place over a period of time. Most of the adaptations discussed in Sect. 1.2 are concerned with the ability of the user to interact with the mobile device in the immediate future. Therefore, we chose to predict time-sharing for a time window extending from the present onwards. As additional experiments we will also consider the cases where the time window is entirely in the past or centered around the present.

Several reasons supported choosing a time window of 30 s: (1) the relatively *short-term* nature of events and actions in mobile use situations; (2) the relatively *long* spans of time needed for effective time-sharing to become manifest even there, and (3) the envisioned utility to the adaptations discussed in Sect. 1.2. However, we will also report the predictive performance achieved with time windows of 15 and 60 s.

3.3 Time-sharing variables

Third, echoing points made by others before us (Fogarty et al. 2005; Horvitz and Apacible 2003), we believe that there is no *single* measure of “available time” that would support all adaptations. We will therefore use the following four variables for describing complementary aspects of time-sharing within the predicted time window:

- **Total** refers to the total amount of time spent on looking at the interaction task.
- **Longest** refers to the duration of the longest uninterrupted glance to the interaction task.
- **Average** refers to the average duration of glances to the interaction task.
- **Frequency** refers to the number of glances to the interaction task. (Unlike the other variables, larger **Frequency** does not necessarily mean more processing time for the interaction task due to the accumulation of switch costs.)

These variables are best suited for optimizing the presentation of information or the timing of interactions, or being conveyed as a social awareness cue (see Sect. 1.2). Providing task resumption cues or proactive preparation of resources would involve predicting glances *away* from the interaction task. The corresponding set of variables for these adaptations is basically the reverse of the list presented above, and is omitted from our analysis for the sake of simplicity. On the other hand, the time-sharing variables may represent only a subset of the information that would be needed for adaptation of functionalities. Explicit modeling

of the *tasks* performed by the users might also be required, but in this paper we focus on time-sharing as a general phenomenon relevant for several kinds of applications.

4 The data set

Building predictive models of time-sharing requires a realistic data set, which we acquired from Oulasvirta et al. (2005). The data set covers mobile Web browsing carried out in an urban setting where the users were traveling and visiting several different kinds of places. Urban mobility is characteristic of mobile HCI, and due to the complex and dynamic nature of the situations, it represents a suitable acid test for predictive models. We here report how the data was gathered and what assumptions about the phenomenon are implied.

4.1 Experimental method and data collection

The method used by Oulasvirta et al. (2005) is called a *semi-naturalistic field study* because of the partial control over the events in the experiment, particularly as determined by the tasks the subjects performed and the locations in the city they were performed in. In this context, observing user behavior required full capture and recording of events with a four mini-camera setup.

4.1.1 Participants

Twenty-eight subjects participated in the study; 15 of them were 20 to 26 and 13 of them 41–47 years old. Half of the participants were male, half female. They were experienced in using mobile phones ($M = 7.5$ years) and browsing the Web with a PC ($M = 6.7$ years). They were also familiar with the Helsinki area ($M = 24.1$ years) and its public transportation system ($M = 6.2$ years). None of them had prior experience with mobile browsers.

4.1.2 Tasks and materials

The subjects performed 25 assigned information retrieval tasks using an Opera browser on a Nokia 6600 (see Appendix A). They were taken to nine situations in a city center (busy street, escalator, quiet street, bus, metro platform, railway station, cafeteria, metro car, laboratory) and, while performing the tasks, they were either explicitly asked to do something typical of the situation (e.g., walking, drinking coffee in a café) or the activity was implicit in the situation (e.g., getting off the bus). The total recording time per subject was about 1 h. Appendix B shows a time-annotated example of the progression of an individual experiment.

Each task was performed in one of three Instructed Time Pressure (ITP) conditions: (1) in the *hurry* condition, the instruction was to “Do as many tasks as you can as quickly as possible.” (2) In the *baseline* condition, a single task was performed within a given (4 min) or implicit time frame (e.g., “You can continue doing the task until we arrive to the Sörnäinen metro stop”). The time frame was sufficient to perform the task, but if exceeded, the experimenter stopped the task and instructed the subject to move on to the next task. (3) In the *waiting* condition, the participants waited for something, and were told that they had plenty of time to carry out a single task: “We’ll be waiting for a call from my colleague, you have plenty of time.” The presence of the ITP manipulation is beneficial for ecological validity, because not all tasks were therefore carried out “as quickly as possible”.

4.1.3 Design

The subjects in both age groups were randomly assigned (1) route direction (normal or reverse) and (2) task order (normal or reverse). However, it should be noted that the relationship between locations and tasks was not fully random. We will return to this limitation of the data set later when evaluating the usefulness of place-related and task-related sensors.

The Instructed Time Pressure (“hurry”, “wait”, or an implied “deadline”) conditions were assigned to natural reference situations (although some of them could not be assigned to certain situations, e.g., the “wait” ITP to walking situations). With repetitions of the situations (e.g., there were several escalator, metro, and walking situations) within a set, a different ITP was administered each time, if possible. Thus, the order of the ITPs was only partially counterbalanced, and the ITPs could not be entirely separated from the nine locations.

4.1.4 Recording and analysis

Four 30 g Watec WAT 230A minicams were used for recording the trials. The video streams were sent to a receiver in the participant’s backpack and backed up onto a tape carried by the experimenter. Figure 2A shows the camera setup.

From the video tapes, deployment of visual gaze was manually coded at a granularity of 1 s. The other coded variables included: task id (25 different values), location (9), posture and mode of movement (4), crowdedness (4), page-loading state (3), the ITP condition (3), and interaction with the device (2). Later on, these codings were augmented with background information about the subjects (to simulate user profiles), locations, and tasks (see Sect. 5). A total of 33 h of video was analyzed in this manner. Figure 2B shows an example of video output.

The experimenter’s shadowing posed a possible source of distraction to the participants, despite the fact that they were instructed not to talk to or look at the experimenter. Indeed, the data does contain occasional glances to the experimenter, but their relative frequency is low compared to glances to other targets in the environment. Moreover, since the shadowing was always done in the same way, the experimenter following one or two footsteps behind the participant, we believe that its effect is uniform across the situations. Measuring the impact of an experimenter’s presence to multitasking behavior remains to be examined rigorously, but at the time this study was conducted, there was no alternative to a human experimenter recording the environment of the participant.

4.2 Time-sharing variables: descriptive statistics

While Oulasvirta et al. (2005) reported only the gaze deployment behavior observed during page loading, we here include all the data in our analysis. The purpose of this section is to introduce the reader to the data by giving examples of the relationships between observable conditions and the four time-sharing variables. By contrast to the prediction task (see Sect. 3) where we compute the probability distributions of the time-sharing variables for the immediate *future*, this analysis looks at dependencies between pairs of variables within the *same* time window.

As can be seen from Fig. 3, different places were associated with different time-sharing behavior. Moreover, recent change of place was a factor. For example, when the *place had changed* within the last 30 s, *Average* dropped from 20.9 s to just 13.2 s. These regularities hint that there are qualitative differences in how processing time is shared in different situations.

Fig. 2 (A) The mini-camera setup used to record attention, action, and context. (B) Video integrated on the fly from the four streams.



Second, *walking speed* had a strong (Pearson) correlation with all of the time-sharing variables ($|0.260| < \text{all } r\text{'s} < |0.360|$). Correlation with *crowdedness*, for example, was notably weaker ($|0.019| < \text{all } r\text{'s} < |0.085|$). Third and not surprisingly, more interaction with the browser, as measured by segments of interaction started in a time window, was associated with more glances, average *Frequency* rising from 1.67 to 2.08 to 2.46 for 0, 1–2, and 3–4 glances, respectively. Average was lower in situations when the performance of the task had already continued for over 60 s (17.1 s vs. 18.2 s). Similarly, a change in the page loading state within the last 30 s was associated with increasing *Frequency* as well (1.71 for no change and 2.23 for change). Instruction to hurry or to wait had little effect on time-sharing. For example, while the difference between the hurry condition and the wait condition was statistically significant ($p < 0.01$ in a post hoc LSD test), in practice the difference remained small, about 3% for *Total*.

Finally, variables describing the user's familiarity with the city were associated with the time-sharing variables, more experienced participants being able to better concentrate on the interaction task (e.g., *Frequency* being 2.32 for those who had lived in Helsinki under

5 years, and 1.97 for those with 5 or more years). On the other hand, these variables coincided with the age of the participant.

5 Constructing the predictive models

At this stage, we are concerned with preliminary assessment of the feasibility of predicting time-sharing. Some of the inputs from the simulated sensors are different in nature from the information that would be obtained from real sensors, and the models therefore do not address the full set of issues involved in creating a working system. On the other hand, we examined the potential usefulness of a wide variety of sensors and several complementary formulations of the prediction task, attempting to identify fruitful directions for further efforts. A simple and straightforward approach was appropriate.

The task of the predictive models was formulated as a classification problem: computing the value or posterior distribution of a binary time-sharing variable on the basis of the observed values of the sensor variables. The procedure used for defining the classes is described in Sect. 5.4. There were three main reasons to use classification models, although the original time-sharing variables are numeric:

- *Almost all of the adaptations presented in Sect. 1.2 are discrete.* The only exceptions are the adjustment of timing and the provision of social awareness cues, in which the level of interruptability or availability could be represented by either a discrete or a continuous variable. Applications would typically choose between a small number of alternative courses of action based on the prediction supplied by the model. This means that classification accuracy is the appropriate performance criterion, although the relevant distinctions may vary from one application to another.
- *The results are intuitively understandable.* The meaning of classification accuracy is obvious and relatively easy to relate to the needs of an application. This would not have been the case for mean square error or other metrics used for evaluating the performance of regression models. On the other hand, it should be noted that we assume the costs of all misclassifications to be equal. Using explicit cost matrices would have complicated the

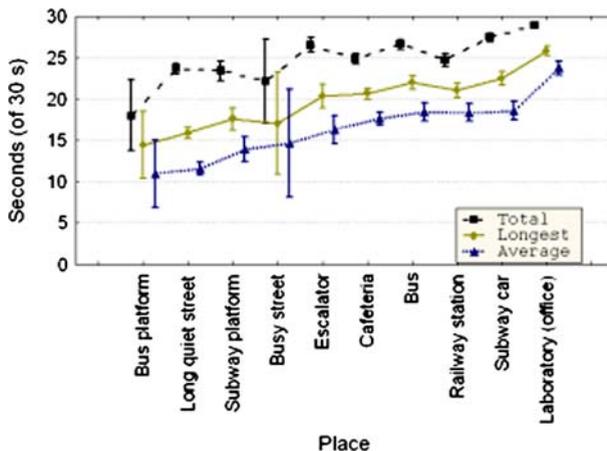


Fig. 3 The relationship between place and three indicators of time-sharing in the original human-coded data. The vertical bars represent 95% confidence intervals

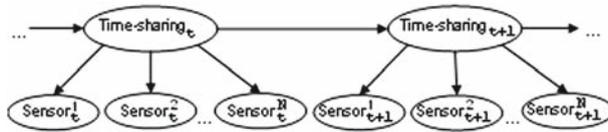


Fig. 4 Graphical representation of the structure of the model

interpretation of the results, but it may be appropriate in more specific work on individual applications.

- *Choosing a suitable model was easier in classification.* The kind of classifier discussed in the next section is widely understood and known to perform reasonably well on a wide variety of data sets. Since many of the conceptual issues involved in defining the problem itself are non-trivial, it seemed appropriate to keep the technical aspects of the work simple and focus on assessing the overall predictability of time-sharing from multiple perspectives.

5.1 General approach and procedure

As a generic model structure, we used the type of *Bayesian network classifier* illustrated in Fig. 4. For each time window, the model computes the posterior probability distribution of the time-sharing variable on the basis of the previous distribution and the current values of the sensor variables.³ The result could consist of the most probable value, the most probable value together with its probability, or the entire distribution. The ability to associate probabilities with the predictions could be useful for applications, enabling e.g., a utility-based approach to adaptation (Horvitz 1999a).

The model relies on a number of assumptions reflecting both practical considerations and our understanding of the underlying phenomenon. The temporal dynamics of time-sharing were modeled as a *first-order Markov process*, in which the current state depends only on the immediately preceding state, and the possible effect of the more distant past is ignored (for arguments see Sect. 2). We also assumed the underlying process to be *stationary*, which means that possible changes in the process itself were not considered. It is conceivable that some users may have learned better strategies for time-sharing during the course of the experiment, or it may have simply taken them some time to become familiar with the mobile device. To account for the possibility of such effects, the stage within the experiment was included as a sensor variable. Finally, we assumed the sensor variables to be *conditionally independent*, given the value of the time-sharing variable. Despite being unrealistic, this assumption is very common in Bayesian classifiers, because it simplifies the model, lowers the computational demands, and often gives excellent performance with real-world data sets. The resulting model is sufficiently simple for real-time prediction in modern mobile devices like smartphones and PDAs.

The original data matrix was coded from video by adding a new row of values whenever one or more changes were observed (Sect. 4). In the *preprocessing* of the data, the resulting sequence was mapped to fixed-duration time windows, and the entries contained in each time window were passed on to *feature extraction* to create the data vectors used for training and testing the classifier. We ended up programming the feature extractors by hand, relying heavily on our understanding of the semantics of the underlying variables (see Sect. 5.2).

³ See e.g., Russell and Norvig (2003) for a technical description of the relevant computations.

Learning a Bayesian classifier from data involves estimating the statistical dependencies between the class variable and the predictor variables. All of the values, including those of the time-sharing variable, were present in the training data, and the standard algorithm for calculating the maximum-likelihood parameters was applied. During validation, however, the *actual* value of the time-sharing variable was not made available to the model, and the temporal dependency between successive time windows was taken into account by *marginalizing* over the possible values. In other words, more information was obtained from the experiment (with additional equipment and human labor) than would be available to the mobile device in real use. In Sect. 6.4, we examine a variation, in which an eye-movement camera is assumed make past time-sharing directly observable.

The choice of predictor variables has a significant effect on the performance of a Bayesian classifier (Guyon and Elisseeff 2003). We applied a wrapper method (Kohavi and John 1997), which involved searching the space of possible sensor combinations and evaluating the models by means of cross-validation. With 179 candidate sensors, the space was far too large to be searched exhaustively, and we employed *simulated annealing* and *greedy* algorithms in an attempt to find good (but not necessarily optimal) classifiers efficiently.⁴

During the search, models with different sensors had to be compared to each other somehow. The most straightforward way to do this was to observe the predictive accuracy directly. The procedure, known as *cross-validation*, is based on splitting the available data repeatedly into two independent samples. One of the samples is used for training the classifier, and the other for validation. In the case of time series data, random splits often lead to overoptimistic estimates of performance because of spurious dependencies between the samples (Hjorth 1994). Furthermore, the structure of the model required the data to be presented in the original temporal order during prediction. For these reasons, we used the individual *tasks* performed by the subjects as the basis for creating the splits. Each task in turn was used as validation data for a model trained on the remaining tasks, and the performance was measured in terms of the overall classification accuracy. The results reported in Sect. 6 are based on the same procedure.

5.2 Information sources

Our assortment of simulated sensors can be characterized in terms of two dimensions. On the one hand, we relied on certain *information sources*, which included aspects of the environment, actions and characteristics of the user, and properties of the task that the user was performing. These were basically determined by the design of the experiment, the contents of the resulting videos, and the procedure used for coding the raw data from the videos (see Sect. 4.1.4). On the other hand, we employed four different *sensor types*, each of which processed the available information in a different way. Significant effort was put into capturing the *relative timing* of events and actions: *Indicators* focused on the present, *EventTrackers* and *HistoryTrackers* on the past, and *FuturePeekers* anticipated things that would happen in the near future.⁵ The general idea was to provide a wide variety of potentially useful sensors to the learning algorithm and identify good subsets by means of search in the space of sensor combinations.

⁴ Each sensor can be either included in the model or left out, which means that the number of possible models (with the given model structure) is 2^{179} . An exhaustive search is clearly infeasible, and even very large amounts of processing time would not change the nature of the problem.

⁵ Implementing *FuturePeekers* in a working system would involve constructing additional predictive models, but in this wizard-of-oz feasibility study we take them as given.

In the human-made coding of the data, the *environment* was represented by 13 variables. One of the variables identified the place where the user had been located at a particular moment, and 10 others provided information about the characteristics of the place. Distinctions were made e.g., between indoors and outdoors, vehicles and buildings, and whether or not there were cars and other pedestrians present. The overall level of crowdedness was rated on a scale from 1 to 4. Furthermore, the time of day was also included as an environment related variable, as it could e.g. provide information about lighting conditions.

The *posture and movement* of the user was encoded in a single variable with values for sitting, standing still, walking slowly, and walking at a normal speed. Another variable indicated whether or not *interaction* with the mobile device was happening at a particular moment, and the *state of the page* in the browser was classified as fully loaded, loaded except for images, or unreadable.

Certain aspects of the *user background* were also included in the data. The users represented two different age groups, approximately half of them being 20–30 years old and the rest 40–50 years old. The background information also included the gender of the user, as well as his or her experience (in years) of using the Web. Finally, the number of years the person had been living in the city and using public transportation served as rough indications of overall familiarity with the environment.

The general properties of the *tasks* were included in the data in an attempt to capture the goal-oriented nature of time-sharing. Each task was performed under one of three different kinds of *Instructed Time Pressure* (see Sect. 4.1.2). The difficulty of the navigation required for finding the desired piece of information was represented by a binary variable (easy/demanding), and the number of criteria for identifying the information varied between 1 and 4. The tasks also differed in terms of the type of input required from the user. About 17 of the 25 tasks were based on navigation along hyperlinks, five on scrolling and searching on a longer page, and the remaining three involved defining a query to a search engine. Furthermore, the size of the Web site and its presumed familiarity to the user were rated on a three-point scale.

5.3 Sensor types and feature extraction

Figure 5 illustrates the temporal relationships among the sensor types. The current moment is denoted by t , and the time-sharing variable being predicted extends from t to $t + 30$ s. An *Indicator* simply reports the state of a particular variable, telling e.g., that the user is currently sitting. An *EventTracker*, in contrast, is triggered by a certain event (e.g., the start of interaction), and uses it as a reference point for computing a relational feature (e.g., the elapsed duration of the current interaction segment). *HistoryTrackers* summarize the recent past, telling e.g., the proportion of the time window taken by the browser to load a new page. Three versions of each *HistoryTracker* were made, with time spans of 15, 30 and 60 s. *FuturePeekers* are the complement of *EventTrackers*, computing a relational feature with respect to a future event (e.g. the amount of time left before the user needs to step out of a metro car).

Appendix C provides a comprehensive listing of the sensors. The *EventTrackers* were triggered either by any change in the value of a certain variable (e.g., movement) or a particular kind of change (e.g., a change from walking or standing to sitting). In both cases, the computed feature was the amount of time since the change, discretized into seven classes (0 s, 1–5 s, 6–10 s, 11–30 s, 31–60 s, more than 60 s, no change observed since the beginning of the task) reflecting the assumption that there is a nonlinear relationship between time and the user's time-sharing. The *FuturePeekers* were based on the same principles (including

the discretization), except that they detected changes that would happen in the near future. In addition to the absolute time difference, the *FuturePeekers* related to interaction, page loading and the ongoing task also computed the stage (as a percentage) within the relevant time period (see rows 56, 64 and 81 in Appendix C). The *HistoryTrackers* produced the largest variety of features. The majority of them computed a binary feature indicating whether or not a change (or a particular kind of change) was observed within the time window. Other *HistoryTrackers* reported the proportion of the time window spent in a crowded environment (row 25 in Appendix C), in a particular posture (rows 28–30), interacting with the device (row 53) or waiting for a new page to become available in the browser (row 63). In addition, the number of changes in crowdedness (row 20) and walking speed (row 40), and the number of separate segments of interaction with the device (row 54) were observed. The ranges of these two kinds of numeric values were discretized to 3–4 equal width intervals with the extremes (e.g., 0% and 100%) separate. Finally, the *HistoryTrackers* monitoring movement also reported the average walking speed (row 38) and detected the presence or absence of accelerating and slowing speed (rows 46 and 50).

There were 179 sensors in total (see Table 2 and Appendix C). The purpose of the sensors associated with the environment was to provide information about the overall demands of the mobility tasks affecting the user’s time-sharing, and to account for the user’s cognitive strategies (including *calibration* and *brief sampling*) for coping with changes in the environment (see Table 1).

There were 13 *Indicators* monitoring the environment, which just replicated the manual codings of the variables in the original data (rows 1–11, 18 and 26 in Appendix C). For each variable except for the time of day, the time from the last change and the time to the next change were reported by an *EventTracker* and a *FuturePeeker*, respectively (rows 13–14, 16–17, 21–22). 33 of the *HistoryTrackers* monitored changes in the variables related to place (rows 12, 15). Three of them indicated whether or not the place had changed within the observed time window (15, 30 or 60 s), and the rest reported particular kinds of changes (e.g. a move from indoors to outdoors or vice versa) in each of the 10 characteristics

Fig. 5 Temporal relationships among the sensor types

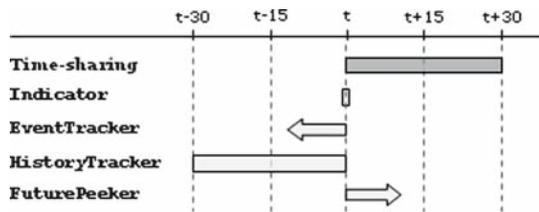


Table 2 The distribution of the sensors by information source and sensor type

Source	Indicators	EventTrackers	HistoryTrackers	FuturePeekers	Total
Environment	13	12	48	12	85
Movement	2	5	36	5	48
Interaction	1	1	9	2	13
Page state	1	2	12	3	18
User	5	0	0	0	5
Task	7	1	0	2	10
Total	29	21	105	24	179

of the place. The remaining 15 `HistoryTrackers` monitored crowdedness (rows 19–20, 23–25). In addition to detecting the presence or absence of a change, they indicated the frequency of changes, increases and decreases in crowdedness, and the proportion of the time window with a high level of crowdedness.

The actions of the user were also assumed to provide information about the division of resources between mobility and interaction tasks. It seems plausible that more resources might be available for interaction when the user is sitting or standing rather than walking, and changes in walking speed could indicate *resource withdrawal* from one or the other of the tasks (see Sect. 2). Furthermore, past interactions provide direct evidence of past allocation of resources towards the mobile device, and the state of the page visible in the browser affects the timing of the interactions.

The momentary posture and movement of the user was reported by an `Indicator` that just replicated the manual encoding (row 27 in Appendix C), and another `Indicator` represented the associated speed of movement (row 37). The timing of specific changes was monitored by pairs of `EventTrackers` and `FuturePeekers`, which were triggered either by any change in speed, an increase or a decrease in speed, or a switch from a standing to a sitting position or vice versa (rows 32–33, 35–36, 41–42, 48–49). The same set of changes was also monitored by `HistoryTrackers` indicating the presence or absence of the change in a specific time window (rows 31, 34, 39, 43, 47). Other `HistoryTrackers` computed the frequency of changes in speed, the average speed, and the proportion of the time window that the user had spent sitting, standing still or walking (rows 28–30, 38, 40). Slowing and accelerating speed, defined in terms of the difference between the end points of the time window, were also detected (rows 46, 50).

Interactions with the mobile device were monitored by 13 sensors. Once again, an `Indicator` replicated the manual encoding of the data, telling whether the user was interacting at a particular moment (row 51). An `EventTracker` and a `FuturePeeker` computed the time from the end of the last interaction segment and the time to the start of the next segment (rows 55, 57). Another `FuturePeeker` indicated the *stage* within an ongoing interaction segment as the percentage completed (row 56). The remaining nine sensors were `HistoryTrackers` observing sequences of past interactions (rows 52–55). They reported the presence or absence of interaction, the relative proportion of interaction, and the number of separate interaction segments.

Due to the relatively slow Internet connection of the mobile device, the loading of new pages in the browser also seemed likely to affect the sharing of time between mobility and interaction tasks. An `Indicator` reported the state of the page at a particular moment, and an `EventTracker` and a `FuturePeeker` computed the time difference relative to the closest change in the past or the future (rows 58, 60–61). In case the page was loaded, another `EventTracker` reported the amount of time that it had been available (row 67). Similarly, a pair `FuturePeekers` anticipated the amount of time left before an unloaded page would become available (rows 64–65). One of them gave the result in absolute terms and the other as the stage in the loading process. `HistoryTrackers` also monitored several closely related, but somewhat different aspects of page loading (rows 59, 62–63, 66). They checked if the state of the page had changed during the time window, and more specifically, whether or not a new page had become available. A complementary set of `HistoryTrackers` indicated if the observed time window contained page loading, and computed its relative proportion.

The variables describing user background remained stable throughout the experiment. They were represented by `Indicators` that just replicated the original encoding of the data (rows 68–72).

The properties of the assigned information retrieval tasks were assumed to reflect the overall demands of the interaction. In addition, tracking progress within a task seemed crucial for capturing the cognitive strategies of the user (see Sect. 2). At the beginning of a task, *calibration* might increase the relative amount of resources allocated to interaction tasks, and *task finalization* might have the same effect in the end.

The properties of the tasks were represented by *Indicators* (rows 73–78 in Appendix C). An *EventTracker* computed the time from the start of the task, and *FuturePeekers* gave the remaining time both in absolute terms and as the proportion completed (rows 79–81). In addition, the stage within the experiment was tracked by an *Indicator* in order to account for the possibility that the users developed better strategies during the course of the experiment (row 82).

5.4 Thresholds of the time-sharing variables

Each of the original time-sharing variables was discretized into two classes. As described above, we were interested primarily in the overall feasibility of predicting time-sharing, and wanted to compare the predictability of the four variables to each other. The thresholds defining the boundary between the two classes were chosen in such a way that as little as possible was known a priori about the value of the variable. In other words, the frequencies of the two classes were made (roughly) equal. Defining the classes in this way gave a better indication of the overall predictive power of the models than working with a strongly biased prior distribution would have given, and applying the same principle consistently across all of the variables enabled direct comparisons.

The thresholds resulting from this procedure were *Total* < 29 s, *Longest* < 22 s, *Average* < 14 s and *Frequency* < 2 s. These thresholds reflect the fact that the data contained mostly *concentrated interaction* with the mobile device. The participants seemed to prefer performing the tasks in a focused manner, as also indicated by the small effect of *Instructed Time Pressure* (see Sect. 4.2). The observed willingness to allocate as much time to interaction as the other demands of the situation permitted supports the idea of conceptualizing the user's time as a limited resource, the availability of which varies over time.

6 Validation results

In this section we report the predictive performance that was achieved under various assumptions about the available sensors and the prediction task itself. We start by presenting the best performing model for each time-sharing variable. The results give a general indication of the predictability of time-sharing and the feasibility of the adaptations. After that, we analyze in more detail the contributions of the sensors to predictive accuracy in order to assess the relative importance of the various factors affecting time-sharing. The proposed sensors differ significantly in terms of the amount of effort that would be required for implementing them, and the results therefore provide a basis for preliminary assessment of the potential costs and benefits. Finally, we experiment with a number of modifications to the prediction task. We assume that past time-sharing is observable to a hypothetical device equipped with an eye-movement camera, and change the size of the predicted time window as well as its temporal location with respect to the sensors.

Table 3 Overall predictive performance for each of the time-sharing variables

Time-sharing variable	Accuracy	Gain	No. of sensors
Total	72.2	22.0	22
Longest	69.6	19.2	30
Average	72.3	17.2	36
Frequency	69.7	17.1	23

6.1 Overall predictive performance

Table 3 presents the overall performance of our best classifier for each time-sharing variable. The selection of the sensors was based on the cross-validation procedure described in Sect. 5.1. All sensors were available to the learning algorithm, and the number of sensors chosen in the model was not constrained explicitly. In addition to looking at the classification accuracy as an absolute number, it is useful to compare it to the proportion of the largest class (which is often called the *default*). We refer to the difference between the two numbers as *gain*, because it represents the benefit of relying on the model as opposed to a simple guess that ignores all sensor data.

The absolute classification accuracy is around 70% for all of the variables. In terms of gain, the result for *Total* (22%) is somewhat better than the others. All of the models rely on a fairly large and varied collection of sensors. In the case of *Total*, for example, 11 of the 22 sensors are related to the environment, three to the posture and movement of the user, two to interaction, two to the state of the page, one to the background of the user, and three to the ongoing task. The importance of the various kinds of sensors is examined in detail in the next two sections.

Table 4 shows the confusion matrices of the best models. For *Total*, *Longest* and *Average* the distribution of the errors is very similar. In about 60% of the misclassified instances the user allocated less time towards the mobile device than predicted, and in the remaining 40% the error is the reverse. In the case of *Frequency*, the errors are distributed evenly between the two classes. However, the total number of situations where the user’s time-sharing was fragmented ($Frequency \geq 2$) is somewhat higher, and the relative proportion of errors is therefore about 4% points lower.

The ROC curves of the models summarize the trade-off between the coverage in predicting particular kind of time-sharing and the associated error rate (see Fig. 6). The curves were produced by combining the predictions from all the test folds used in the cross-validation and ordering the resulting list on the basis of the probability of the predicted value. The certainty of the model about the correctness of the predictions decreases from left to right. Therefore, each point on the curve is associated with a threshold, which would result in a certain coverage of the true positives at the cost of a certain error rate. The shape of the curve is affected by both the inherent uncertainty involved in the prediction task and the quality of the model as an approximation. In particular, “easy” instances that are predicted correctly move the curve towards the upper left-hand corner and increase the area under the curve, which is denoted by A' .

An application in which the cost of inappropriate adaptations is significant could require a higher probability threshold for the positive class. However, errors could not be avoided with any threshold. In the case of *Total*, for example, requiring at least 75% probability (instead of 50%) would reduce the number of false positives only to 51.3%, while decreasing

Table 4 Confusion matrices of the best models

		Predicted	
		Total < 29 s	Total ≥ 29 s
Actual	Total < 29 s	965	485
	Total ≥ 29 s	317	1119
		Predicted	
		Longest < 22s	Longest ≥ 22 s
Actual	Longest < 22 s	885	546
	Longest ≥ 22 s	331	1124
		Predicted	
		Average < 14 s	Average ≥ 14 s
Actual	Average < 14 s	818	478
	Average ≥ 14 s	321	1269
		Predicted	
		Frequency < 2	Frequency ≥ 2
Actual	Frequency < 2	928	439
	Frequency ≥ 2	435	1084

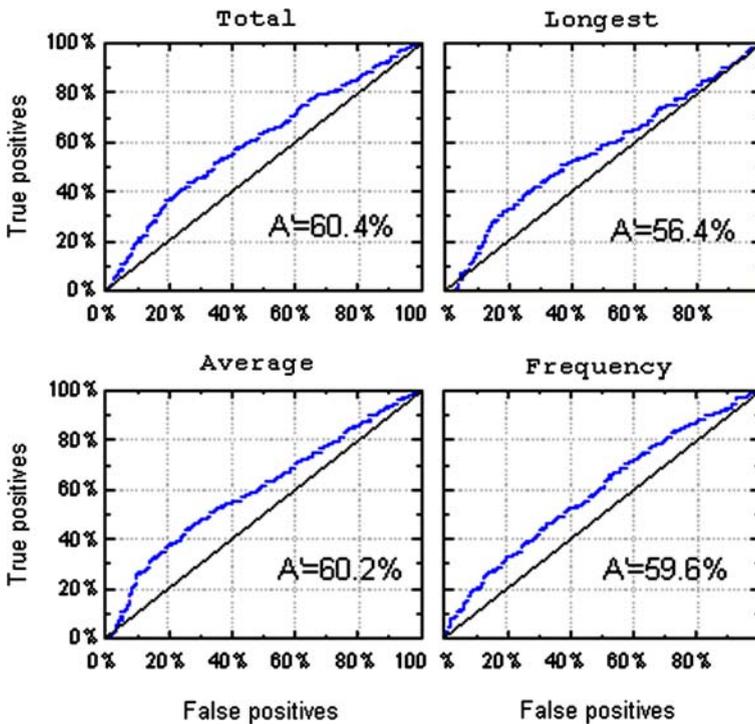


Fig. 6 ROC curves of the best models. The “positive” class is the one in which the user allocates more time to interaction with the device. A' is the area under the curve

Table 5 Predictive performance with each sensor group

Sensor group	Variable	Variable			
		Total	Longest	Average	Frequency
User background (5): age + gender + experience with web + familiarity with city	Accuracy	59.7	57.8	60.7	57.6
	Gain	9.5	7.4	5.6	5.0
	No. of sensors	4	3	3	2
Activity (79): movement + inter- action + browser state	Accuracy	67.3	66.1	68.6	65.3
	Gain	17.1	15.7	13.5	12.7
	No. of sensors	15	8	12	10
Environment (84): place + crow- dedness	Accuracy	67.7	66.4	68.0	64.7
	Gain	17.5	16.0	12.9	12.1
	No. of sensors	14	15	13	22
Level 1 (27): interaction + browser state + time of day – FuturePeekers	Accuracy	58.2	57.7	60.5	60.8
	Gain	8.0	7.3	5.4	8.2
	No. of sensors	7	9	8	9
Level 2 (87): Level 1 + user back- ground + place – FuturePeekers	Accuracy	68.1	65.4	67.8	65.6
	Gain	17.9	15.0	12.7	13.0
	No. of sensors	18	16	11	17
Level 3 (169): Level 2 + movement + crowdedness + FuturePeekers	Accuracy	71.6	69.3	72.0	69.6
	Gain	21.4	18.9	16.9	17.0
	No. of sensors	18	21	31	23
No. FuturePeekers (155): Level 3 + task – FuturePeekers	Accuracy	69.5	66.7	69.5	66.4
	Gain	19.3	16.3	14.4	13.8
	No. of sensors	26	26	25	19
All (179): Level 3 + task	Accuracy	72.2	69.6	72.3	69.7
	Gain	22.0	19.2	17.2	17.1
	No. of sensors	22	30	36	23

The number of sensors in the group appears in parentheses

the coverage of true positives to 64.4%. The situation is essentially the same with the other variables, despite minor variations in the shape of the curve.

6.2 Importance of the sensor groups

In order to evaluate the usefulness of the various kinds of sensors, we divided them into groups and computed the classification accuracy achievable within each group. Firstly, a distinction was made between the sensors related to the background of the user, the actions of the user, and the environment. We were primarily interested in the relative importance of these complementary information sources, and seeing if any of them would alone be sufficient for predicting time-sharing. The results are shown on the first three rows of Table 5. Secondly, we divided the sensors into five groups on the basis of the challenges that would be involved in implementing them. If particular kind of information was available, it would in most cases be relatively easy to create a variety of sensors computing different features based on the same information. Therefore, the nature of the underlying information source is more crucial for the difficulty of implementation than the specific computations performed by the sensors.

The first group, which we named *Level 1*, would require only stand-alone software without external infrastructure or data that might be difficult to acquire (see Table 5). These sensors

would provide information about the time of the day, the state of the page in the browser, and the user's interaction with the browser. *FuturePeekers* are not included, as implementing them would require prediction in itself. *Level 2* extends the first group with user background and place. The provision of these two information sources would require voluntary input from the user, relatively fine-grained location estimation, and a database about the characteristics of the locations. *Level 3* includes the sensors for tracking the movements of the user and the crowdedness of the place. The former would require additional hardware and the latter either a historical database or an infrastructure for real-time monitoring. *Level 3* is also the first group containing *FuturePeekers*. Finally, the last two groups include the sensors associated with the task being performed by the user. Acquiring such information automatically would involve both conceptual and technical difficulties. As the names suggest, *No FuturePeekers* excludes the sensors anticipating the future and *All* represents the entire collection.

The results are summarized in Table 5. Relying only on the background information about the user limits the gain within the range 5.0–9.5%. Monitoring either the activities of the user or the characteristics of the environment enables significantly better results. For all of the variables the achieved classification accuracies are 5% points or less below the best models, which appear on the last row.

The simple software sensors of *Level 1* give modest results with gains of 5.4–8.2%. A substantial improvement is achieved by adding the sensors for user background and place. For *Level 2*, the gains are within 13.0–17.9%, which is again less than 5% points below the best models. Further improvement results from adding the sensors for movement and crowdedness along with the *FuturePeekers*. The performance with this collection of sensors, labeled *Level 3*, is within one percentage point of the best models. Adding the task related sensors and removing all *FuturePeekers* lowers the gain to 13.8–19.3%. The *FuturePeekers* do not seem to be critically important, however, as the performance is still only about 3% points below the best models.

The predictability of the time-sharing variables decreases from left to right in Table 5. With every sensor group except for *Level 1*, the largest gain is achieved for *Total*. The gain for *Longest* is consistently better than for *Average*, and also better than for *Frequency* with the exception of *Level 1* sensors. The differences between *Average* and *Frequency* are relatively small, but on five of the eight rows *Average* has the larger gain.

Increases in the number and sophistication of the candidate sensors consistently result in better performance. Despite the general trend, the numbers do not fully correspond to our expectations. In particular, the sensors related to the ongoing task improve the performance by less than 1% point. It seems that taking the users' task-related strategies (associated with e.g., the beginning or the end of a task, as described in Table 1) into account is either not crucial or our sensors do not capture them adequately. As pointed out in Sect. 4.1.3, the relationship between locations and tasks is not fully random in the data, and therefore the place-related sensors could in principle dominate the task-related ones by providing information about both places and tasks. However, additional experimentation indicates that the contribution of the task-related sensors remains small even when no place-related sensors are present. Another surprising and possibly related result is that the *FuturePeekers* improve the performance by only about 3% points. While the time-sharing strategies of the users are likely to be anticipatory in the sense of reflecting expectations of the near future, such effects may not be so pervasive that foresight on part of the model would be necessary.

Table 6 The effect of removing an individual sensor from the model

Variable	Sensor	Decrease in accuracy
Total	Speed of movement	3.6
	Years living in the city	3.4
	Stage in ongoing interaction segment	2.9
	Place: indoors/outdoors	2.4
	Time to start of next interaction segment	1.4
Longest	Speed of movement	4.9
	Years living in the city	2.8
	Stage in ongoing interaction segment	1.7
	Place: indoors/outdoors	1.4
	Size of the web site	1.1
Average	Proportion of time walking (60s)	3.8
	Years living in the city	2.3
	Stage in ongoing interaction segment	1.8
	Place: pedestrians passing by?	1.5
	Proportion of "crowded" (15s)	1.4
Frequency	Stage in ongoing interaction segment	4.0
	Place: pedestrians passing by?	2.8
	Proportion of "crowded" (15s)	2.5
	Time to next change in page state	2.4
	Time of day	2.1

6.3 Importance of individual sensors

One way to evaluate the importance of an individual sensor is to remove it from the model and observe the decrease in classification accuracy. Table 6 shows the five most important sensors for each time-sharing variable, as determined by this procedure. The models used were the ones with the best predictive performance (see Sect. 6.1 and the last row of Table 5).

As can be seen from Table 6, there is significant overlap between the top five sensors of each time-sharing variable, and the sensors represent the full diversity of the available information sources. Six of the 11 different sensors on the list appear two or more times. The most frequent ones are *stage in ongoing interaction segment* and *years living in the city*. The most frequent information sources are *environment* (seven sensors) and *interaction* (five sensors). *Posture and movement* and *user background* are each represented by three sensors, and *page state* and *task* by 1 sensor. In terms of sensor types, Table 6 is dominated by Indicators (11 sensors) and FuturePeekers (six sensors). There are only three HistoryTrackers and no EventTrackers at all.

A complementary way to assess the importance of individual sensors is to start from an "empty" model and add sensors one by one, observing the effect on classification accuracy. We used a greedy algorithm, which at each step added the sensor that gave the largest increase in performance. It should be noted that this method does not in general produce the best classifier

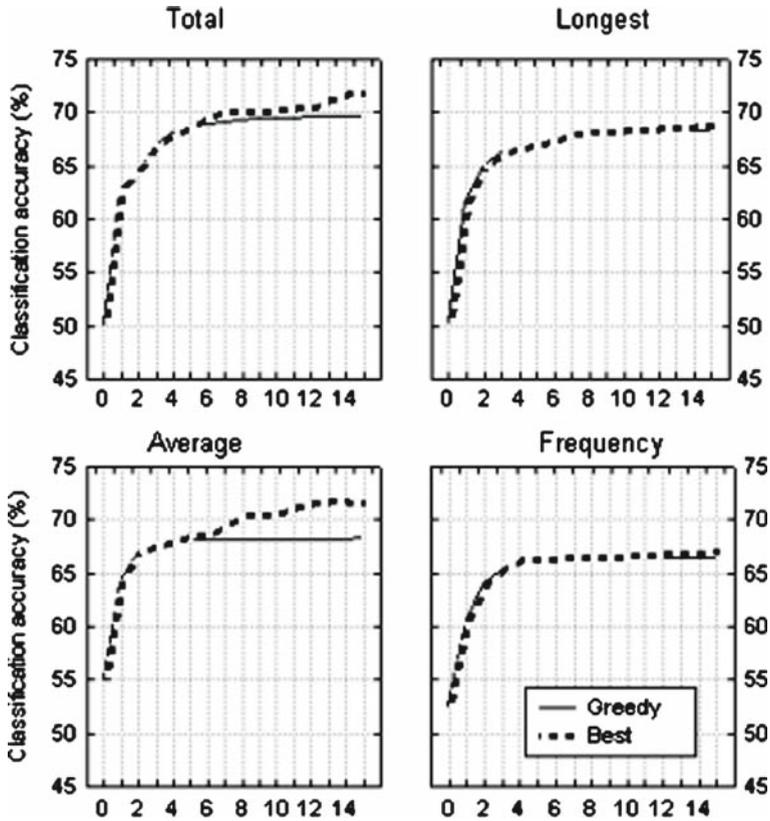


Fig. 7 Improvement in classification accuracy as the number of sensors is increased

with a given number of sensors, but despite that the results are illuminating—especially for small classifiers with only a few sensors.

The results are illustrated in Fig. 7. In each graph, the lower curve shows the performance achieved by the greedy algorithm, and the upper curve the performance of the best models that we were able to find in the absence of the restriction that previously added sensors could not be changed. With *Longest* and *Frequency*, the difference between the curves at the end of the range is less than one percentage point, whereas with *Total* and *Average* the greedy algorithm gets stuck in a local optimum much earlier and ends up 2–3% points lower. A clear trend in all of the graphs is that the performance improves rapidly as the first sensors are added, and levels off gradually. With 10 sensors, the upper curve is already within 1–3% points of the best results.

Table 7 shows for each time-sharing variable the first five sensors added by the greedy algorithm. Again, the list is well-balanced in the sense of representing all information sources except for *task*. The most frequent sensor types are *FuturePeeker* (nine sensors), *Indicator* (five sensors) and *HistoryTracker* (five sensors).

For *Total*, *Longest* and *Average*, the best model with only one sensor relies on a *HistoryTracker* or an *Indicator* monitoring the *posture and movement* of the user, and the first five sensors also include a *FuturePeeker* anticipating a particular kind of change in posture or movement. For *Average*, the fourth sensor is a *HistoryTracker*

Table 7 The effect of adding sensors incrementally in the model

Variable	Sensor	Increase in accuracy
Total	Proportion of time sitting (60 s)	12.6
	Years living in the city	1.8
	Stage in ongoing interaction segment	2.3
	Time from last change in crowdedness	1.2
	Time to next switch from sitting to standing	0.5
Longest	Proportion of time walking (15 s)	11.3
	Years living in the city	3.2
	Time to next increase in speed	1.3
	Place: pedestrians passing by?	0.5
	Time to start of next interaction segment	0.3
Average	Speed of movement	9.3
	Time to next increase in speed	2.6
	Time to next change in character of place	0.5
	Stage in ongoing page loading	0.3
	Switched from sitting to standing? (60 s)	0.2
Frequency	Place ID	8.0
	Time to next change in page state	3.2
	Time to start of next interaction segment	1.4
	Accelerating speed? (30 s)	0.9
	Moved on/off a vehicle? (15 s)	0.1

that recognizes accelerating walking speed. All of the sequences also contain at least one sensor related to the *environment*. In the case of *Total*, it reports the time from the last change in crowdedness, and is the only *EventTracker* appearing in the table. The *Indicator* that identifies the place is the best individual sensor for predicting *Average*, and three other sensors in the table provide information about the characteristics of the place. *Interaction* is represented by three sensors and *page state* by two sensors, all of which are *FuturePeekers*. The number of years the person has been living in the city, an *Indicator* related to *user background*, appears as the second sensor for *Total* and *Longest*.

6.4 Variations to the prediction task

The results presented above are based on the assumption that time-sharing cannot be observed directly during prediction. However, if the mobile device was equipped with a crude eye-movement camera, *past* time-sharing would become at least partially observable. The output of the eye-movement camera would enable an additional sensor, which would tell the *correct value* of the time-sharing variable in the previous time window. This turns the model structure presented in Fig. 4 into a Naive Bayes classifier, in which the values of all sensor variables affecting the time-sharing variable are known. The results of this variation are shown on the second row of Table 8. For every time-sharing variable, the classification accuracy improves, but not dramatically. (The results for the original version of the prediction task are repeated on the first row for convenience.) The largest improvement, 5.2% points, is achieved for *Frequency*, and the improvements for *Total*, *Longest* and *Average*

Table 8 Results for modified versions of the prediction task

Variation		Variable			
		Total	Longest	Average	Frequency
Original formulation	Accuracy	72.2	69.6	72.3	69.7
	Gain	22.0	19.2	17.2	17.1
	No. of sensors	22	30	36	23
Eye-movement camera available	Accuracy	75.8	72.3	74.9	74.9
	Gain	25.6	21.9	19.8	22.3
	No. of sensors	17	19	29	25
Detection of past time-sharing	Accuracy	72.5	71.3	72.7	69.4
	Gain	22.2	20.4	17.3	16.9
	No. of sensors	14	27	25	20
Detection of present time-sharing	Accuracy	71.3	70.9	72.3	69.3
	Gain	20.9	20.6	17.2	16.5
	No. of sensors	18	26	42	17
15 s time window	Accuracy	69.0	–	–	69.6
	Gain	18.3			5.5
	No. of sensors	36			18
60 s time window	Accuracy	74.0	72.2	73.2	71.6
	Gain	21.9	21.7	22.9	21.3
	No. of sensors	21	25	19	25

are 3.6, 2.7 and 2.6% points, respectively. All of the models have fewer sensors than the ones appearing on the first row, but also in this respect the effect of the variation is relatively modest.

The next two variations concern the temporal location of the time-sharing variable with respect to the sensors. For some applications it might be relevant to know the past ($t - 30:t$) or the present ($t - 15:t + 15$) time-sharing rather than the immediate future ($t:t + 30$). Intuitively, the more direct evidence provided by the sensors should make the problem easier, but this turns out not to be the case. The results are practically the same as in the original formulation, with only the three largest deviations (in gain) within 1.0–1.5%.

Finally, we changed the duration of the predicted time window from 30 s to 15 and 60 s. New thresholds were determined with the procedure discussed in Sect. 5.4. In the case of 15 s time window, *Total*, *Longest* and *Average* all got a threshold of 15 s, which means that all three models would predict whether or not the user would be looking at the mobile device for the entire duration of the time window. The results are reported in the column labeled *Total* in Table 8. For *Frequency*, the threshold with 15 s time window is 2. Applying the procedure to 60 s time window gave thresholds of 57, 34 and 19 s for *Total*, *Longest* and *Average*, and 3 for *Frequency*.

The results appear on the last two rows of Table 8. With 15 s time window, the achieved performance is in both cases lower than in the original formulation. Although the absolute classification accuracy for *Frequency* is almost the same, the gain is much smaller due to higher default. Doubling the duration of the time window to 60 s improves the gains for *Longest*, *Average* and *Frequency* by 2.5, 5.7 and 4.2% points, respectively. As a result, the differences between the time-sharing variables are smaller than in the original formulation.

7 Summary and conclusions

We were set out to assess the feasibility of sensor-based prediction of the time-sharing of mobile users. The situations in which mobile devices are used constitute a complex and dynamic task environment, and the presence or absence of systematic dependencies in the data was of significant theoretical and practical interest as such.

Drawing from the literature on time-sharing in cognitive psychology and human factors, we constructed 179 simulated sensors providing information about the environment, posture and movement, interaction with the mobile device, state of the page in the browser, user background, and task. One important aspect of our work was to operationalize time-sharing as a phenomenon taking place over a period of time. Rather than asking whether or not the user is looking at the device at a particular moment, we characterized time-sharing in terms of behavioral patterns long enough to reflect regularities arising from the top-down control of attention. The patterns were defined in terms of the number of glances (*Frequency*), the duration of the longest glance (*Longest*), and the total (*Total*) and average (*Average*) durations of glances to the interaction task within 30-s time windows. Momentary tracking of gaze deployment would be straightforward with an eye-movement camera, and difficult or impossible without it.

With this approach, the absolute classification accuracy was found to be around 70% for all of the variables, with the best variable, *Total*, reaching 72% with 22 sensors (22% gain). In about 60% of the misclassified instances the user allocated less time to interaction with the mobile device than predicted, and in the remaining 40% the error was the reverse. We also explored a number of variations to the prediction task. Even if the mobile device was equipped with an eye-movement camera observing the *past* time-sharing directly, the accuracy of the predictions would increase only by 3–5% points. It was also surprising to discover that moving the time window of the time-sharing variable from the future ($t:t+30$) to the past ($t-30:t$) did not make the problem easier, but the results were practically the same as in the original formulation. Decreasing the size of the time window to 15 s gave lower performance, and doubling it to 60 s somewhat higher performance.

Our explorations with the various sensor groups give an indication of the effort and infrastructure that would be required for a working implementation of the adaptations. Relying only on the background information about the user or the simplest software sensors did not give satisfactory results, but with all the other sensor groups the achieved performance was within 5% points of the best models (see Table 5). Information about either the activities of the user or the characteristics of the environment enabled 3–5% points lower performance compared to the best models. Similar results were achieved when relying on the combination of the simplest software sensors, the background information about the user and the characteristics of the place, all without the *FuturePeekers* (Level 2 in Table 5). Adding the sensors for movement and crowdedness along with the *FuturePeekers* improved the performance to a range within one percentage point of the best models. The *FuturePeekers* would be particularly difficult to implement, but they did not seem critically important, as omitting them decreased the performance by only about 3% points. Another interesting and somewhat surprising finding was that the sensors related to the ongoing task, arguably the most difficult ones to implement, improved the performance by less than 1% point.

Application developers are also interested in identifying the individual sensors that give the largest improvements in prediction accuracy. When sensors were added one by one using a greedy algorithm, the improvement in performance was surprisingly rapid with the first few sensors. Only 2–4 sensors were needed to get within 5% points of the best results (see Table 7). These sensors provided information about *posture and movement* (4 occurrences),

environment (3), *interaction* (2), *user background* (2), and *page state* (1). While getting reasonably close to the best achievable performance with just a few sensors seems encouraging on the one hand, it should be noted that almost all of the various kinds of information were needed. Furthermore, 5 of the 12 sensors were *FuturePeekers*, which would require another level prediction in a working application.

Predicting the time-sharing of mobile users seems like a hard problem. The most obvious explanation for this is that the relevant factors are only partially observable. In particular, the cognitive processes responsible for the top-down control of time-sharing are inaccessible to sensors and likely to remain so in the foreseeable future. Furthermore, most of the information about the environment that we assumed to be available is statistical by nature, and does not include the *specific events* that may draw the user's attention. Comprehensive monitoring of the surroundings and automatic real-time interpretation of the resulting data streams is way beyond the reach of the current technology. Due to these limitations, we believe that dramatic improvements in the predictive performance are not achievable, even though larger data sets would enable the construction of more sophisticated models.

Accepting this seemingly pessimistic view bears implications for further efforts in the area of user modeling. Adaptations that do not tolerate a substantial number of erroneous predictions are probably not realistic, and the efforts should therefore focus on ideas that can be implemented in a "forgiving" form. In order to maximize the performance of such adaptations, the collection of sensors that we relied on could be extended to several directions. The number of years that the user had lived in the city turned out to be one of the most useful sensors we had, and the acquisition of more fine-grained information about the user's familiarity with particular city districts, as well as other kinds of knowledge and skills, might well be worthwhile. Some of the information might be possible to acquire automatically by monitoring the user's activities and mobility patterns for an extended period of time. Access to personal information such as the user's calendar might facilitate the recognition of certain kinds of social events, like meetings and lectures. As mentioned above, it is probably not realistic to expect mobile devices to acquire the kind of *comprehensive* awareness of the environment that humans have, but this does not preclude the possibility that *some* fairly specific events and situations might be possible to recognize on the basis of audio and video streams or explicit signals received from ubiquitous computing devices. The benefits of adapting to the time-sharing of mobile users may not be large enough to motivate the development of a complex technological infrastructure, but even a single "killer application" with similar needs could expand the possibilities significantly.

Acknowledgements The Academy of Finland funded this work within the framework of the *Prima* and *ContextCues* projects, and in the final stages the first author received financial support from The Finnish Work Environment Fund. We thank Anthony Jameson, Nicky Kern, Heikki Summala and the anonymous reviewers for valuable comments.

Appendix A: List of information retrieval tasks used in the experiment

1. Report the latest news heading from *Iltalehti* [a newspaper].
2. Report the hours of today's spinning lessons at *Helsinki Fitness Center*.
3. Report today's special flight offer from *Finnair* [an airline].
4. Report the time of departure for the next train to *Lappeenranta* [a city].
5. Report any library that carries the movie *Pahat Pojat* in DVD.
6. Report today's menu at the *Unicafe* restaurant in *Porthania* [a building on the university campus].
7. Report the current song playing on *GrooveFM* [a radio channel].

8. Report the latest culture news from Helsingin Sanomat [a newspaper].
9. Report the TV shows shown on MTV3 and Nelonen [TV channels] today at 20.30.
10. Navigate through Google to the Web site of the University of Art and Design Helsinki.
11. Report the opening hours of the Arabianranta [a city district] library.
12. Report the current value of the HEX [Helsinki stock exchange] index.
13. Report the weather forecast for Helsinki from Foreca [a meteorological service].
14. Report the quickest route from the Parliament to Otaniemi [a city district] using Reittiopas [a journey planner].
15. Report the time and price of the next Pikku G [a band] concert from Lippupalvelu [a ticket seller].
16. Report the description of the movie Pirates of the Caribbean from Makuuni [a video rental company] and the average rating from the viewers.
17. Report today's menu at the Otaniemi [a city district] student restaurant.
18. Report the ticket price to the first night club event in Helsinki listed on Klubitus.org [a Web site].
19. Report the wish list created on the Sokos [a department store] Web site by your friend who is getting married.
20. Report the cross-country skiing routes available at the moment in Helsinki.
21. Report the next time and place for seeing the movie Under the Tuscan Sun.
22. Report the open jobs in the field of marketing from the Web site of the Ministry of Labour.
23. Report the movie theater that shows Levottomat 3 [a movie] and the showing times today.
24. Report how long Kiasma [a modern arts museum] is open today.
25. Acquire the Estonian-English dictionary to your phone.

Appendix B: An example of the progression of an experiment

The table below shows a time-annotated listing of the tasks and places included in an individual experiment. The subject was a 47-year-old female secretary working for a marketing company. Please note that the time *between* the tasks was not effective, and the durations of the tasks cannot therefore be calculated directly from the first column. Moreover, because the subject in question could not perform all of the tasks within the given time, only 17 tasks out of the 25 were tried out.

Time	IR Task	ITP condition	Place	Mobility task
0:00	24	Wait	Long quiet street	Walk to the end of the street to the metro station
12:52	2	Implied deadline	Metro station	Walk to the escalator
13:51	2	Implied deadline	Escalator	Go to the metro platform
15:22	2	Implied deadline	Metro platform	Wait for the metro
18:04	4	Wait	Metro car	Get off at the Sörnäinen station
21:56	7	Wait	Metro car	Get off at the Sörnäinen station
24:07	7	Wait	Metro platform (at destination)	Go upstairs
25:02	7	Wait	Escalator	Go upstairs

Appendix B: continued

Time	IR Task	ITP condition	Place	Mobility task
25:48	7	Wait	Metro station (upstairs)	Walk to the cafeteria outside
31:50	5	Wait	Cafeteria	Eat a bun and drink coffee
36:56	6	Wait	Cafeteria	Eat a bun and drink coffee
46:19	9	Hurry	Busy street	Walk outside to the bus stop
56:41	10	Implied deadline	Bus stop / Bus	Get off at the central railway station
64:58	10	Implied deadline	Railway station square	Walk to the ticket vending machines
65:44	11	Hurry	Railway station hall	Stand in the middle of the hall
71:00	13	Wait	Metro station	Sit down and wait a few minutes for the metro
75:50	15	Hurry	Metro station	Catch the next metro back
76:04	15	Hurry	Escalator	Catch the next metro back
76:56	15	Hurry	Metro platform	Catch the next metro back
81:06	17	Implied deadline	Metro car	Get off at the last station
83:51	18	Hurry	Metro platform (at destination)	Go upstairs
84:04	18	Hurry	Escalator	Go upstairs
86:04	18	Hurry	Metro station (upstairs)	Walk back to the starting place
87:12	18	Hurry	Long quiet street	Walk back to the starting place
96:44	21	Hurry	Laboratory	Sit down in front of a table
97:38	22	Hurry	Laboratory	Sit down in front of a table
102:42	23	Hurry	Laboratory	Sit down in front of a table
104:21	1	Hurry	Laboratory	Sit down in front of a table

Appendix C: List of sensors

Row	Information source	Sensor	Sensor type
1	Environment	Place ID (10 values)	Indicator
2	Environment	Place: urban/suburban	Indicator
3	Environment	Place: indoors/outdoors	Indicator
4	Environment	Place: vehicle?	Indicator
5	Environment	Place: open place (i.e. no surrounding buildings nearby)?	Indicator
6	Environment	Place: heated/cold	Indicator
7	Environment	Place: public / semipublic / private	Indicator
8	Environment	Place: for passage/stay	Indicator
9	Environment	Place: cars passing by?	Indicator
10	Environment	Place: pedestrians passing by?	Indicator
11	Environment	Place: cars or pedestrians passing by?	Indicator
12	Environment	Place changed?	HistoryTracker ^a
13	Environment	Time from last place change	EventTracker
14	Environment	Time to next place change	FuturePeeker

Appendix C: continued

Row	Information source	Sensor	Sensor type
15	Environment	Characteristic of place changed?	HistoryTracker ^b
16	Environment	Time from last change in place characteristic	EventTracker
17	Environment	Time to next change in place characteristic	FuturePeeker
18	Environment	Crowdedness (1–4)	Indicator
19	Environment	Crowdedness changed?	HistoryTracker
20	Environment	Number of changes in crowdedness	HistoryTracker
21	Environment	Time from last change in crowdedness	EventTracker
22	Environment	Time to next change in crowdedness	FuturePeeker
23	Environment	Crowdedness increased?	HistoryTracker
24	Environment	Crowdedness decreased?	HistoryTracker
25	Environment	Proportion of “crowded” (<i>crowdedness</i> ≥ 3)	HistoryTracker
26	Environment	Time of day (9–11/11–14/14–16/16–18)	Indicator
27	Posture/movement	Sitting/standing/walking slowly/walking normally	Indicator
28	Posture/movement	Proportion of time sitting	HistoryTracker
29	Posture/movement	Proportion of time standing	HistoryTracker
30	Posture/movement	Proportion of time walking	HistoryTracker
31	Posture/movement	Switched from sitting to standing?	HistoryTracker
32	Posture/movement	Time from last switch from sitting to standing	EventTracker
33	Posture/movement	Time to next switch from sitting to standing	FuturePeeker
34	Posture/movement	Switched from standing to sitting?	HistoryTracker
35	Posture/movement	Time from last switch from standing to sitting	EventTracker
36	Posture/movement	Time to next switch from standing to sitting	FuturePeeker
37	Posture/movement	Speed of movement (0–2)	Indicator
38	Posture/movement	Average speed	HistoryTracker
39	Posture/movement	Speed changed?	HistoryTracker
40	Posture/movement	Number of changes in speed	HistoryTracker
41	Posture/movement	Time from last change in speed	EventTracker
42	Posture/movement	Time to next change in speed	FuturePeeker
43	Posture/movement	Increase (temporary or permanent) in speed observed?	HistoryTracker
44	Posture/movement	Time from last increase in speed	EventTracker
45	Posture/movement	Time to next increase in speed	FuturePeeker
46	Posture/movement	Accelerating speed ($speed(start) < speed(end)$)?	HistoryTracker
47	Posture/movement	Decrease (temporary or permanent) in speed observed?	HistoryTracker
48	Posture/movement	Time from last decrease in speed	EventTracker
49	Posture/movement	Time to next decrease in speed	FuturePeeker
50	Posture/movement	Slowing speed ($speed(start) > speed(end)$)?	HistoryTracker
51	Interaction	Interacting currently?	Indicator
52	Interaction	Interaction observed?	HistoryTracker
53	Interaction	Proportion of interaction	HistoryTracker
54	Interaction	Number of interaction segments	HistoryTracker
55	Interaction	Time from start of ongoing interaction segment	EventTracker
56	Interaction	Stage in ongoing interaction segment	FuturePeeker

Appendix C: continued

Row	Information source	Sensor	Sensor type
57	Interaction	Time to start of next interaction segment	FuturePeeker
58	Page state	Page state (loading/loaded without images/fully loaded)	Indicator
59	Page state	Page state changed?	HistoryTracker
60	Page state	Time from last change in page state	EventTracker
61	Page state	Time to next change in page state	FuturePeeker
62	Page state	Page loading observed?	HistoryTracker
63	Page state	Proportion of page loading	HistoryTracker
64	Page state	Stage in ongoing page loading	FuturePeeker
65	Page state	Time before page being loaded becomes available	FuturePeeker
66	Page state	New page became available?	HistoryTracker
67	Page state	Time current page has been available	EventTracker
68	User	Age group (20–30/40–50)	Indicator
69	User	Gender	Indicator
70	User	Experience of web browsing (1–4 years/more)	Indicator
71	User	Years living in the city (1–4 years/more)	Indicator
72	User	Years using public transportation (none/1–4 years/more)	Indicator
73	Task	Difficulty of navigation (easy/difficult)	Indicator
74	Task	Number of criteria identifying the information (1–4)	Indicator
75	Task	Type of input required (navigation/scrolling/query)	Indicator
76	Task	Size of the web site (1–3)	Indicator
77	Task	Familiarity of the web site (1–3)	Indicator
78	Task	Instructed time pressure (hurry/baseline/waiting)	Indicator
79	Task	Time from start of task	EventTracker
80	Task	Time to end of task	FuturePeeker
81	Task	Stage in ongoing task	FuturePeeker
82	Task	Running number of ongoing task set (1–5)	Indicator

^a There were three instances of each *HistoryTracker*, with time spans of 15, 30 and 60 s.

^b There were 10 instances of this and the following two sensors, one for each of the 10 characteristics (urban/suburban, indoors/outdoors,...) listed above.

References

- Adamczyk, P.D., Bailey, B.P.: If not now, when? The effects of interruption at different moments within task execution. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2004), pp. 271–278. ACM Press, New York (2004)
- Altmann, E.M., Trafton, J.G.: Task interruption: resumption lag and the role of cues. In: Proceedings of the 26th Annual Conference of the Cognitive Science Society, pp. 42–47. Lawrence Erlbaum Associates, Hillsdale, NJ (2004)
- Begole, B., Tang, J., Hill, R.: Rhythm modeling, visualizations and applications. In: Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology, pp. 11–20. ACM Press, New York (2003)
- Burgess, P.W.: Strategy application disorder: the role of the frontal lobes in human multitasking. *Psychol. Res.* **63**, 279–288 (2000)

- Carberry, S.: Techniques for plan recognition. *User Model. User-Adapt. Interact.* **11**(1–2), 31–48 (2001)
- Card, S.K., Henderson, A.: A multiple, virtual-workspace interface to support user task switching. In: *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 1987)*, pp. 53–59. ACM Press, New York (1987)
- Cutrell, E., Czerwinski, M., Horvitz, E.: Notification, disruption, and memory: effects of messaging interruptions on memory and performance. In: *Proceedings of Interact 2001: IFIP Conference on Human-Computer Interaction*, pp. 263–269. IOS Press, Amsterdam (2001)
- Dourish, P., Bellotti, V.: Awareness and coordination in shared workspaces. In: *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW 1992)*, pp. 107–114. ACM Press, New York (1992)
- Eng, K., Lewis, R.L., Tollinger, I., Chu, A., Howes, A., Vera, A.: Generating automated predictions of behavior strategically adapted to specific performance objectives. In: *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2006)*, pp. 621–630. ACM Press, New York (2006)
- Fischer, G.: User modeling in human-computer interaction. *User Model. User-Adapt. Interact.* **11**(1), 65–86 (2001)
- Fogarty, J., Ko, A.J., Aung, H.H., Golden, E., Tang, K.P., Hudson, S.E.: Examining task engagement in sensor-based statistical models of human interruptibility. In: *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2005)*, pp. 331–340. ACM Press, New York (2005)
- Fogarty, J., Lai, J.: Examining the robustness of statistical models. In: *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2004)*, pp. 207–214. ACM Press, New York (2004)
- Fogarty, J., Lai, J., Christensen, J.: Presence versus availability: the design and evaluation of a context-aware communication client. *Int. J. Hum.-Comput. Stud.* **61**(3), 299–317 (2004)
- Fu, W.T., Gray, W.D.: Resolving the paradox of the active user: stable suboptimal performance in interactive tasks. *Cogn. Sci.* **28**(6), 901–935 (2004)
- Glanzer, M., Dorfman, D., Kaplan, B.: Short-term processing in the processing of text. *J. Verbal Learn. Verbal Behav.* **20**, 656–670 (1981)
- González, V.M., Mark G.: Constant, constant, multi-tasking craziness: managing multiple working spheres. In: *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2004)*, pp. 113–120. ACM Press, New York (2004)
- Gray, W.D., Boehm-Davis, D.A.: Milliseconds matter: an introduction to microstrategies and to their use in describing and predicting interactive behavior. *J. Exp. Psychol. Appl.* **6**(4), 322–335 (2000)
- Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. *J. Mach. Learn. Res.* **3**, 1157–1182 (2003)
- Hjorth, J.: *Computer Intensive Statistical Methods: Validation, Model Selection, and Bootstrap*. Chapman & Hall, London (1994)
- Ho, J., Intille, S.S.: Using context-aware computing to reduce the perceived burden of interruptions from mobile devices. In: *Proceedings the ACM Conference on Human Factors in Computing Systems (CHI 2005)*, pp. 909–918. ACM Press, New York (2005)
- Horvitz, E.: Principles of mixed-initiative user interfaces. In: *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 1999)*, pp. 159–166. ACM Press, New York (1999a)
- Horvitz, E.: Uncertainty, action and interaction: in pursuit of mixed-initiative computing. *IEEE Intell. Syst.* **14**(5), 17–20 (1999)
- Horvitz, E., Apacible, J.: Learning and reasoning about interruption. In: *Proceedings of the Fifth International Conference on Multimodal Interfaces (ICMI 2003)*, pp. 20–27. ACM Press, New York (2003)
- Horvitz, E., Jacobs, A., Hovel, D.: Attention-sensitive alerting. In: *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence (UAI'99)*, pp. 305–313. Morgan Kaufmann, San Francisco, CA (1999)
- Horvitz, E., Kadie, C.M., Paek, T., Hovel, D.: Models of attention in computing and communications: from principles to applications. *Commun. ACM* **46**(3), 52–59 (2003)
- Hudson, S., Fogarty, J., Atkeson, C., Avrahami, D., Forlizzi, J., Kiesler, S., Lee, J., Yang, J.: Predicting human interruptibility with sensors: a wizard of oz feasibility study. In: *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2003)*, pp. 257–264. ACM Press, New York (2003)
- Jameson, A., Kiefer, J., Mueller, C., Grossmann-Hutter, B., Wittig, F., Rummer, R.: Assessment of a user's time pressure and cognitive load on the basis of features of speech. Technical report, German Research Institute for Artificial Intelligence, Saarbrücken, Germany (2006)
- Jameson, A., Klöckner, K.: User multitasking with mobile multimodal systems. In: Minker, W., Bühler, D., Dybkjær, L. (eds.) *Spoken Multimodal Human-Computer Dialogue in Mobile Environments.*, pp. 349–377. Springer, Dordrecht (2005)

- Jameson, A., Schaefer, R., Weis, T., Berthold, A., Weyrath, T.: Making systems sensitive to the user's changing resource limitations. *Knowledge-Based Syst* **12**, 413–425 (1999)
- Kern, N., Antifakos, S., Schiele, B., Schwaninger, A.: A model for human interruptibility: experimental evaluation and automatic estimation from wearable sensors. In: *Proceedings of the Eighth International Symposium on Wearable Computers (ISWC'04)*, pp. 158–165. IEEE Computer Society, Washington, DC (2004)
- Kern, N., Schiele B.: Context-aware notification for wearable computing. In: *Proceedings of the 7th IEEE International Symposium on Wearable Computers (ISWC'03)*, pp. 223–230. IEEE Computer Society, Washington, DC (2003)
- Kobsa, A.: Generic user modeling systems. *User Model. User-Adap. Interact.* **11**(1–2), 49–63 (2001)
- Kohavi, R., John, G.: Wrappers for feature selection. *Artif. Intell.* **97**(1–2), 273–324 (1997)
- Kushleyeva, Y., Salvucci, D., Lee, F.J.: Deciding when to switch tasks in time-critical multitasking. *Cogn. Syst. Res.* **6**, 41–49 (2005)
- Mäntyjärvi, J., Seppänen, T.: Adapting applications in handheld devices using fuzzy context information. *Interact. Comput.* **15**(4), 521–538 (2003)
- McFarlane, D.C., Latorella, K.A.: The scope and importance of human interruption in human-computer interaction design. *Hum. Comput. Interact.* **17**(1), 1–61 (2002)
- Miyata, Y., Norman, D.A.: Psychological issues in supporting multiple activities. In: Norman, D.A., Draper, S.W (eds.) *User Centered Design: New Perspectives on Human-Computer Interaction*, pp. 266–284. Lawrence Erlbaum Associates, Hillsdale, NJ (1986)
- Monsell, S.: Task switching. *Trends Cogn. Sci.* **7**(3), 134–140 (2003)
- Näätänen, R.: *Attention and Brain Function*. Lawrence Erlbaum Associates, Hillsdale, NJ (1992)
- Oulasvirta, A., Petit, R., Raento, M., Tiitta, S.: Interpreting and acting on mobile awareness cues. *Hum.-Comput. Interact.* **22**(1&2), 97–135 (2007)
- Oulasvirta, A., Tamminen, S., Roto, V., Kuorelahti, J.: Interaction in 4-second bursts: the fragmented nature of attentional resources in mobile HCI. In: *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2005)*, pp. 919–928, ACM Press, New York (2005)
- Pashler, H.: Dual-task interference and elementary mental mechanisms. In: Meyer, D., Kornblum, S. (eds.) *Attention and Performance XIV*, pp. 245–264. MIT Press, Cambridge, MA (1993)
- Russell, S., Norvig, P.: *Artificial Intelligence: A Modern Approach* (2nd edn). Pearson Education, Upper Saddle River, NJ (2003)
- Salovaara, A., Oulasvirta, A.: Six modes of proactive resource management. In: *Proceedings of NordiCHI 2004*, pp. 57–60. ACM Press, New York (2004)
- Salvucci, D.: A multitasking general executive for compound continuous tasks. *Cogn. Sci.* **29**, 257–292 (2005)
- Simon, H.: Designing organizations for an information rich world. In: Greenberger, M. (ed.) *Computers, Communications, and the Public Interest*, pp. 37–72. Johns Hopkins University Press, Baltimore, MD (1971)
- Tamminen, S., Oulasvirta, A., Toiskallio, K., Kankainen, A.: Understanding mobile contexts. *Pers. Ubiquitous Comput.* **8**(2), 135–143 (2004)
- Vera, A., Howes, A., McCurdy, M., Lewis, R.L.: A constraint satisfaction approach to predicting skilled interactive cognition. In: *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2004)*, pp. 121–128. ACM Press, New York (2004)
- Vertegaal, R.: Attentive user interfaces. *Commun. ACM* **46**(3), 31–33 (2003)
- Wickens, C.D.: Processing resources in attention. In: Parasuraman, R., Davies, R. (eds.) *Varieties of Attention*, pp. 63–102. Academic Press, New York (1984)
- Wickens, C.D.: Multiple resources and performance prediction'. *Theor. Issues Ergon. Sci.* **3**(2), 159–177 (2002)
- Wikman, A.S., Nieminen, T., Summala, H.: Driving experience and time-sharing during in-car tasks on roads of different width. *Ergonomics* **41**, 358–372 (1998)

Authors' vitae

Miikka Miettinen is a Ph.D. candidate in Computer Science at the University of Helsinki. His primary interest lies in the design, development and evaluation of experimental application software. Many of his academic research efforts, including the one presented in this article, have involved user modeling with the objectives of improving awareness in collaborative systems and enabling personalized information retrieval.

Antti Oulasvirta, Ph.D. is a postdoctoral researcher working in the area of human–computer interaction. He completed his Ph.D. in 2006 in the Department of Psychology at the University of Helsinki, investigating working memory requirements relevant in everyday interruptions and multitasking. His primary interest lies in understanding the way mobile and ubiquitous technologies place new demands to attention and memory and how people develop skills and practices to be able to manage the temporal and spatial fragmentation of their work.