Capturing, Analyzing and Utilizing Context-Based Information about User Activities on Smartphones

Wolfgang Woerndl, Florian Schulze
Technische Universitaet Muenchen
Boltzmannstr. 3, 85748 Garching, Germany
{woerndl, schulze}@in.tum.de

Abstract
In this paper, we present some of our work in mobile user modeling following the three steps in a general user modeling process. First, we outline a framework for mobile user activity logging. The framework integrates various hardware and software sensors on smartphones. Second, we have worked on learning relevant user locations for personal information management and recognizing user activities from sensor data to analyze the collected data. Third, the user model can be used to adapt mobile information access, for example in mobile recommender systems. The paper also outlines some requirements for an Activity Context Representation and Exchange Language from the perspective of mobile user modeling.

Introduction
Smartphones and other mobile devices are becoming more and more powerful but still suffer from restricted input capabilities, display sizes and other limitations. Therefore, it is desirable to tailor information access on mobile devices to the current situation. To adapt information access, three main steps can be identified in a general user modeling process (Brusilovsky and Maybury, 2002):
1. Collecting data about the user
2. Analyzing the data to build a user model
3. Using the user model to adapt information access

Mobile user modeling has additional challenges in comparison to learning about user activities in general. Most importantly, the context is very significant in mobile interaction. Context can be defined as any information that can be used to characterize the situation of entities (Dey, Abowd and Salber 2001). We can distinguish between physical and cognitive context. Physical context is location, time, temperature and other sensor data. Cognitive context includes preferences, mental states, tasks and social affinities of human users (Jung, Lee and Choi 2009). Limited attention spans of users while moving also add to the need of mobile personalization.

In this paper, we summarize our work on several aspects of the three steps of mobile user modeling. We first focus on the collection of data about user activities on smartphones in an appropriate granularity for user modeling. Then outline some of our previous and current work to analyze the collected data, building a user model and utilizing the model for personalization and recommendation. We finally outline some requirements for an Activity Context Representation and Exchange Language and conclude the paper.

Capturing User Activities and Sensor Data
For the first step of the user modeling process, the collection of user data, we have implemented a framework for mobile user activity logging on Windows Mobile smartphones (Woerndl, Manhardt and Prinz 2010). The goal is a unified approach for recording user actions on mobile devices in granularity appropriate for user modeling. The framework is based on the MyExperience project as a foundation for our implementation (cf. http://myexperience.sourceforge.net). MyExperience aims at collecting data for allowing early detection of subject compliance or technology issues of mobile software while running in the background and ensuring minimal impact on other programs and services carried out on the device. We extended this system with new hard- and software sensors in order to maximize the range of covered user activities. 27 new sensors were added to the eleven already existing that we used. Figure 1 gives an overview of implemented sensors, grouped into several categories (Woerndl, Manhardt and Prinz 2010).
The approach allows logging data about phone calls, messaging, peripheral devices (e.g. BT headsets), media players, GPS sensor, networking, personal information management (e.g. contacts and appointments), web surfing, system behavior (e.g. power status) and usage of arbitrary applications. It is possible to detect when, at which location and how a user uses an application or accesses certain information, for example.

In each category, we log the information that is needed for detailed and flexible analysis afterwards. For example, we do not just log the most common parameters of a phone call, like phone number, direction of connection, timestamp and duration. But we also try to acquire additional information by cross-referencing these parameters with other sources of information on the phone. A phone number can be looked up in the contact list to determine name and group membership of the caller or callee. As this kind of information is subject to change (a user might delete contacts), it is important to log it when a sensor is triggered instead of trying to make the connection during analysis of the user activity data.

In case of Web browsing, the system records not only websites visited by the user, but also the keywords of a Web search by dissecting the according URL as most search engines form a parameterized URL of a pattern similar to "/search?q=search+term".

One of the most important differences between mobile and non-mobile systems and applications is the relevance of the current user location. Therefore, we aimed for an elaborate and adjustable configuration of positioning sensors. To eliminate inaccuracies we used an algorithm based on the “dilution of precision” (DOP).

Sensors in the category of system state are also very valuable sources of information for a user model. In combination with additional information from other sensors we can use them to infer patterns in user behavior to a great extent. Determining, for example, the situational context that prompts the user to mute her phone could be a fundamental goal to automatically support the user in her everyday activity by shutting down other possibly disruptive or unused functions while the situation lasts.

All logged sensor data is stored in an SQL database. The MyExperience project offers a so-called Analyzer tool. Contrary to the name, however, it is not intended for analysis and interpretation of the data but rather for roughly checking the collection. For evaluation we tried our framework in several usage scenarios and were able to validate that it was able to collect meaningful information about the user without obstructing her too much. Figure 2 shows an excerpt of the logged data, at corresponding user locations. Our framework allows for connecting user activities, system state and features, and sensor data easily.

![Figure 1. Implemented sensors](image1)

![Figure 2. Visualization of logged user activities on map](image2)
relations between resources. To do so, our application assists the user as much as possible.

One area we investigated was analyzing user position logs to be able to suggest relevant locations for inclusion in the personal ontology to the user. We have applied a solution based on a time-based clustering algorithm that drops smaller clusters, which represent places with only a short length of stay (Kang et al. 2005). Each newly logged GPS coordinate is compared to the cluster of its immediate predecessors. If the discrepancy is above a certain threshold, the user is assumed to move away from her previous location. Therefore, a new cluster is created (Figure 3).

![Figure 3. Time-based clustering (Kang et.al. 2005)](image)

However these clusters still lack semantics as they are not associated with real places in terms of a notation that is comprehensible and common to human beings. To map (clusters of) raw GPS coordinates to addresses we used reverse geocoding by means of Google’s Reverse Geocoder, which is part of the Google Maps API family. Locations identified that way can serve as a basis for the recommendation of points-of-interests (POI). Furthermore, we also established a hierarchy of location by extracting the common content of multiple mappings, e.g. the name of a street. By repeating this procedure successively, we were able to gain several ontological layers of different granularity. Following the paradigm of the Semantic Desktop, we created our own ontology to use location information among others for personal information management (Woerndl, Schulze and Yordanova 2010).

We have evaluated the approach with good results in a small user study. The approach is able to recognize locations on different levels, e.g. “city” and “street address”. It is important to note that all the collection and interpretation is done locally on the smartphone. This is significant with regard to privacy because the personal and potentially sensitive information about the user never leaves her mobile device.

### Recognizing User Activities From Sensor Data

Previous work also includes an approach to generate high-level context from sensor data on smartphones (Woerndl, Schueller and Rottach 2007). Thereby, we have developed a process model to transform raw sensor data into aggregated and interpreted information. The logging application on the smartphone was designed to be easily extensible with sensors such as GPS and acceleration, and offers the opportunity to annotate the log data with meta information regarding the user’s current activity. We then used supervised learning algorithms for the interpretation of the raw data. In a case study, we were able to recognize pre-defined user activities such as “standing”, “running” or “biking” from acceleration sensor data with good accuracy.

### Utilizing the User Model for Mobile Adaptation and Recommendation

#### Adapting System Behavior on Mobile Devices

To utilize the context-based information about user activities on smartphones, application examples include the adaptation of mobile applications to the current user situation (e.g. automatically muting the phone in certain situation), the personalization of mobile web searches (Woerndl and Yousef 2007) or the suggestion of items (e.g. other mobile applications) for the user. The latter is the domain of mobile recommender systems.

As far as the analysis of the data logged by the mobile user activity logging framework presented in the second section of this paper is concerned, several applications scenario are reasonable to exploit the user model learned from mobile user activity. It is possible to apply association rules mining to discover rules like “IF wlan=“connected“ AND location=“work place“ THEN Mailprogram.Open”.

The already mentioned Semantic Desktop application SeMoDesk is another example for utilizing information about user activities on smartphones. SeMoDesk allows for managing personal resources like tasks, contacts and appointments using a personal ontology. Resources can be associated with locations that have been learned (see section “Learning Relevant User Locations” above). The ontology can also be utilized to recommend additional
items that are not explicitly managed by the user (Woerndl and Hristov 2009). For this purpose, we have extended the personal ontology by a POI (point-of-interest) concept with sub concepts such as “cinema”, “restaurants”, “shop” etc. The user can then relate tasks or any other resources to POI types. The map feature of SeMoDesk displays information about relevant POIs on a map (Figure 4), together with the location of upcoming appointments (Woerndl and Hristov 2009).

Mobile Recommender Systems

One area we are working on is mobile, context-aware recommender systems. Examples include the recommendation of points-of-interests in the vicinity of a user with a smartphone (Woerndl, Moegele and Prinz 2011), or the recommendation of mobile applications that may be useful for the user in the given context.

The basic idea of a context-aware application recommender is to suggest applications that have been used by other users in a similar context (Woerndl and Schlichter 2008). An example is to suggest a train table application when the user is near a train station, because other users have used this application in a comparable location.

We have worked on a score model to compare different context and other information with regard to the usefulness of an item in a given situation. By doing so, it is possible to integrate collaborative filtering methods. An algorithm would predict the rating of items, and then combine these collaborative filtering item scores with other scores to find relevant items. An example is a mobile restaurant recommender, which evaluates nearer and better – according to the predicted rating – restaurants with higher combined scores, while ruling our restaurants that are closed today altogether. In a mobile setting, good and precise recommendations are essential because of intrinsic obstacles of mobile usage environments (Woerndl, Moegele and Prinz 2011). Therefore, it is important and useful to capture, model and interpret user activities with associated context data on smartphones.

Requirements for an Activity Context Representation and Exchange Language

In this section, we outline some requirements for an Activity Context Representation and Exchange Language, which originated from the work presented in this paper. From our perspective, the identified sensors in the explained mobile user activity logging framework is a good starting point for such a language (cf. Figure 1). It includes all the information about user activities that is available on a smartphone. We think it is useful to treat hardware (i.e. data from hardware sensors on a mobile device, e.g. GPS) and software (i.e. information from applications about what the user is doing on the device) sensors in a unified way. The implementation details may vary from platform to platform, but the basic information as shown in Figure 1 is available on all mobile platforms. An Activity Context Representation and Exchange Language should abstract from platform specific implementation details and allow for easy exchange and aggregation of user activity information accumulated on several devices.

In mobile scenario, the context of user activities is very important. Location is one important example of mobile context. It is necessary to log and manage information on different levels of abstraction, for example with location:

1. GPS logs
2. Important locations and addresses for a user
3. Location with context, e.g. “work place”

The latter two may be derived from log data but also possibly entered or annotated by users. In the mentioned Semantic Desktop project SeMoDesk, we have designed a location and sensor ontology as an extension to the existing personal ontology. By doing so, applications can relate user activities and other information to places in a comprehensive and flexible manner. The location ontology models different spatial entities such as “Room”, “PartOfBuilding”, “City” and so on. In our project, we experimented with an RFID infrastructure to automatically discover the exact location of a user – respectively her
mobile device – indoors, while using GPS outside building. A language for describing context should be capable of integrating various positioning technologies.

An ontology is not only useful when describing location and other context data, but also user activities. In the given example of recognizing user activities from sensor data, we noticed that it is sometimes difficult to distinguish between somewhat related activities such as “walking” and “running”. In this case, it would be useful to organize activities in a hierarchical manner. An algorithm could then infer a more general user activity (such as “moving”), if available sensor data does not allow a clear classification. Some applications areas may require more specific activity models than others. In addition, an activity ontology could also formalize more detailed relationships between concepts, such as activity A is the opposite of activity B.

The overall goal in user modeling and personalization is to achieve a user model as accurate and extensive as possible. Therefore, an Activity Context Representation and Exchange Language should not only contain user activities and the corresponding context, but also be able to integrate other facets of user models such as user needs, preferences and tasks. An example is the inclusion of ratings for activities, which can be exploited in mobile, collaborative recommender systems later. This information may not be modeled on a core Activity Context Representation and Exchange Language, but a corresponding standard should be easily extensible and provide interfaces to other standards.

**Conclusion**

In this paper, we have outlined work on several aspects of the three steps of a (mobile) user modeling process. The focus is on logging, analyzing and utilizing information about user activities including the associated context to allow adaptation of information access in mobile scenarios. Note that in the presented work, the processing of information about user activities and sensor data is mostly performed on smartphones.

**References**


