Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

Int. J. Human-Computer Studies 68 (2010) 525–547

International Journal of Human-Computer Studies

www.elsevier.com/locate/ijhcs

In situ evaluation of recommender systems: Framework and instrumentation

M. Funk[a,*], A. Rozinat[b], E. Karapanos[c], A.K. Alves de Medeiros[b], A. Koca[c]

[a]*Faculty of Electrical Engineering, Eindhoven University of Technology, The Netherlands*
[b]*Department of Information Systems, Eindhoven University of Technology, The Netherlands*
[c]*Faculty of Industrial Design, Eindhoven University of Technology, The Netherlands*

Received 30 December 2008; received in revised form 15 November 2009; accepted 17 January 2010
Available online 29 January 2010

Abstract

This paper deals with the evaluation of the recommendation functionality inside a connected consumer electronics product in prototype stage. This evaluation is supported by a framework to access and analyze data about product usage and user experience. The strengths of this framework lie in the collection of both *objective* data (i.e., "What is the user doing with the product?") and *subjective* data (i.e., "How is the user experiencing the product?"), which are linked together and analyzed in a combined way. The analysis of objective data provides insights into how the system is actually used in the field. Combined with the subjective data, personal opinions and evaluative judgments on the product quality can be then related to actual user behavior. In order to collect these data in a most natural context, *remote data collection* allows for extensive user testing within habitual environments. We have applied our framework to the case of an interactive TV recommender system application to illustrate that the user experience of recommender systems can be evaluated in real-life usage scenarios.

Keywords: Use experience; Recommender systems; Experience sampling; Observation; Process mining

1. Introduction

In a world of information overload and vast entertainment possibilities, complex information services enter also the home. While consumer electronics products in the living room nowadays offer immersive multimedia experiences, content provision works still according to a very simple and traditional broadcasting paradigm. Yet, the choice of content is soaring. Hence, products are increasingly equipped with recommendation functions that guide users to relevant content. Still, the actual usage of recommender systems in the home is a largely unknown domain. Especially from the recommendation point of view, the setting and context differs much from more traditional domains of recommender systems like business, e-commerce, search engines and social networks. Examples of this new type of recommender systems can be found in interactive TV and IPTV applications, but also in music players, e.g., Apple iPod^{TM}'s Genius feature that allows for music recommendations based on the personal listening history. Rather than offering precise recommendations, the main goal in this domain is to create a pleasant user experience, to meet users' expectations. Therefore, an assessment of strengths and shortcomings of such systems and the impact of recommendation is crucial.

Much research in the evaluation of recommender systems has employed metrics that assess the effectiveness of the recommendation service such as its *accuracy*, the degree to which the recommendations cover the entire set of items, or how often the recommendation service leads users to wrong choices (see Herlocker et al., 2004; McNee et al., 2006 for an extensive overview). Such metrics assume that the motivation of recommender system usage is to find specific information, and therefore the extent to which the task of finding this specific information is completed

[*] Corresponding author. Fax: + 31 40 243 3066.
E-mail addresses: m.funk@tue.nl (M. Funk), a.rozinat@tue.nl (A. Rozinat), e.karapanos@tue.nl (E. Karapanos), a.k.medeiros@tue.nl (A.K. Alves de Medeiros), a.koca@tue.nl (A. Koca).

1071-5819/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijhcs.2010.01.002

successfully determines the success of a recommendation service. However, as recommendation services are expanding towards entertainment computing, the motivations that underlie the use of such services extend beyond the traditional goal-achievement paradigm (Hassenzahl and Ullrich, 2007; Karapanos et al., 2008). For instance, in the interactive TV context, people may use a recommendation service simply as a shuffle mechanism, as a means to address their curiosity. In this case, the success of the recommender system will not relate to the accuracy of the recommendation service in terms of correct search results, but instead to the overall experience during this prolonged interaction which might involve also the omission of unwanted recommendations. This impacts recommender systems evaluation research in at least two ways. First, metrics that are based in the goal-achievement paradigm will inevitably fail to capture the qualities that underlie such use cases and therefore new metrics will need to be developed that relate to the user's overall affective state, such as their satisfaction (Chin et al., 1988) or valence (Russell, 2003). Second, lab studies will fail to simulate the full spectrum of possible motivations that might underlie the use of a recommendation service. Recommender systems should thus be evaluated in the field.

Field studies are however hampered by several challenges. Beyond the requirement of a fully engineered system (Konstan and Riedl, 1999), the experimenter has less control over the usage of the system and the feedback episodes. While in a lab study the experimenter may ask the participant to complete a certain task and evaluate the system right after the completion of the task, in field studies the experimenter is bound to participant-initiated feedback. For instance, in Event-Contingent Diaries (Bolger et al., 2003) the participant reports on an event that she considers significant enough while the Day Reconstruction Method (Kahneman et al., 2004) asks the participant to reconstruct the events of the preceding day in a serial order and report on each one of them. Another interesting example is the Experience Sampling Method (Larson and Csikszentmihalyi, 1983; Csikszentmihalyi and Larson, 1992) which prompts the participants at random times to report on their feeling and current actions. Beyond the original implementation of a random prompting, several researchers have attempted to expand Experience Sampling with sophisticated algorithms for the calculation of the right moment to prompt a request for input from the participants (Intille et al., 2003; Froehlich et al., 2007; Khan et al., 2008). This leads to Experience Sampling methods that are triggered by aspects of people's behavior, the so-called event-based experience sampling methods. These might relate to users' interaction patterns with products, people's physical proximity to objects and locations, or any other aspects characterizing their physical and social behavior. However, none of the available approaches mentioned above is applicable to event-based prompting for the evaluation of recommendation services as a connection to a facility that provides events derived from user actions is required.

In this paper, we propose a *framework for the in situ field evaluation of recommender systems*. We present an observation framework that supports the experimenter in identifying interaction patterns in the field and in *dynamically* defining the algorithm that prompts the participants for feedback. A specialty of this framework is that it allows for dynamic changes to the observation logic, i.e., the amount of data and the way the data is logged can be adapted while the remote observation is running. Furthermore, the framework creates use-log data with an inherent semantic structure, where the semantic annotations are preserved and leveraged in the analysis phase (Funk et al., 2009a). Through the case of a recommendation service for interactive TV applications, we illustrate an evaluation procedure that combines the analysis of interaction patterns with users' subjective reports. More specifically, we compare the users' evaluations in three distinct usage modes, namely "browse", "search", and "explorative search" (cf. Fig. 9). Furthermore, users can initiate (positive or negative) feedback at any point in time, and we show how interaction patterns that lead to particular types of feedback can be identified. These interaction patterns are identified by using *process mining* techniques (Alves de Medeiros et al., 2008; van der Aalst et al., 2007a), which are techniques specially suitable for the analysis of temporal log data from a process perspective. Our contributions are the following:

- Building on a general observation framework which is previous work, we present a specific implementation for recommender systems that can be used to observe the interactions of a user with a recommender system in the field.
- This framework is extended to incorporate both participant-initiated feedback and event-based experience sampling. In event-based experience sampling, the participant is prompted for an evaluation of the system based on particular patterns of actions.
- We developed multiple ontologies to enable semantic linking of remotely collected objective and subjective data items, and to facilitate different, even orthogonal, stakeholder views on the acquired usage information.
- We illustrate the opportunities of combining objective data, i.e., information about user actions (the use patterns), and subjective data, i.e., user feedback (user perceptions), through a real case study with an interactive TV application. Furthermore, we provide a detailed discussion on the outcome of this study, the importance of the experience sampling method for the evaluation of highly interactive recommender systems, and the role that our proposed framework can play in this evaluation as a tool.

The remainder of this paper is organized as follows. First, we elaborate on the relationship between the user actions and the user feedback in Section 2. Then, an example case is presented in Section 3. In Section 4 we describe our approach and, in Section 5, its realization. In Section 6, the results of the case study are presented. In Section 7, we reflect on the results of the study and discuss the benefits and limitations of our approach. Finally, the paper is concluded in Section 8. Due to the multi-disciplinary nature of our work it is difficult to present all relevant related publications in a single place. We therefore chose to discuss related work in close connection to the corresponding aspects of our approach throughout the paper rather than in a separate section.

2. Correlating user behavior and user perceptions—an evaluation framework for recommender systems

When evaluating the usability of a product, we are interested both in how exactly the product is being used, and how it is perceived by the user. In a typical lab experiment, we would thus *observe* the actions of the participants (e.g., using a camera) and *ask* them for their opinions (e.g., using a tape recorder). However, a drawback of such a lab experiment is that users may not behave naturally enough in an artificial environment. Furthermore, the possible sample size of manual data collection is limited.

To address these drawbacks, we advocate the use of *remote product usage monitoring*. This approach allows participants to use the products in their habitual environment and—due to its high degree of automation—enables experiments on a larger scale. Fig. 1 illustrates a user interacting with a product which has been instrumented by an observation module to obtain the relevant information. On the left, one can see that the product transmits logs of actual *user actions* (e.g., sequences of used product functions, button presses, etc.) to the remote observer. Furthermore, the user can provide *feedback* to the remote observer (e.g., by filling out electronic forms that appear on the product). The actual product usage logs can be seen as the *objective data*, while the information obtained via user feedback is *subjective data*. These two types of information are complementary and need to be linked to obtain a comprehensive and unbiased understanding of the suitability of the product in the given usage context.

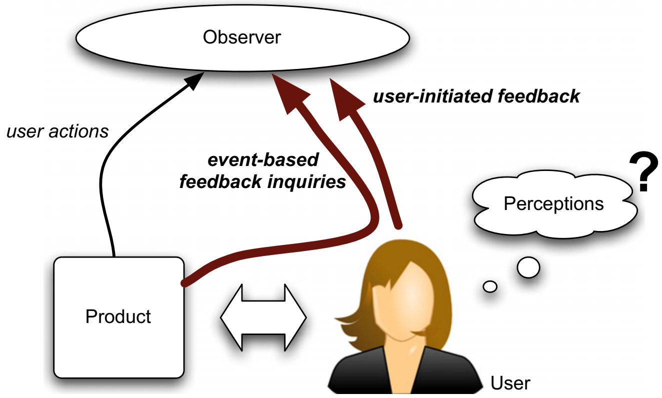


Fig. 1. Insights into both product usage and perceptions of the user while operating the product are necessary to evaluate the user experience of a product in a comprehensive way.

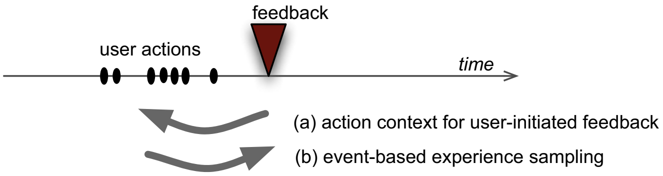

feedback
user actions
time
(a) action context for user-initiated feedback
(b) event-based experience sampling

Fig. 2. Information about preceding user actions can (a) provide the context for user-initiated feedback and (b) trigger event-based experience sampling.

Fig. 1 shows that two types of subjective data are collected by the system: (a) feedback that can be initiated by the user at any given point in time, and (b) feedback that is inquired from the user in situations where her actions match a certain usage pattern. Their relation to the objective usage data can be described as follows.

User-initiated: We can assume that often users provide their feedback in a specific usage context, the context that triggered them to initiate this feedback. This can be of a positive nature (“Nice, I like this feature!”) or criticism (“I only wanted to do *X*, but I had to do it in a very complicated manner.”). In both situations it is interesting to analyze the context in which the feedback was submitted (“What was the user doing before?”), since it may help to gain a deeper understanding of why the user is feeling in a certain way. This situation is illustrated in Fig. 2(a).

Event-based: We are often interested in the opinion of the user with respect to a specific part of the product functionality, or in a particular usage context (“Are users operating the product in this way happier than the users operating the product in that way?”). Furthermore, during analysis of the user actions we may encounter unexpected usage patterns, in which case we can instruct the observation module inside the product to trigger a feedback request to the user to gain more insight into her motives (“Why was the user behaving that way?”). This situation is illustrated in Fig. 2(b).

As can be envisioned from the above, correlating user behavior and user perceptions can be very beneficial for a deeper analysis of product quality (Koca et al., 2008). Previously, we have focused on objective data and presented a framework to monitor the user actions of instrumented products (Funk et al., 2009a). *In this paper, we integrate the collection of subjective data and focus on analyzing the relationship between the feedback* (*user-initiated and event-based*) *and the actual product usage.*

3. Example: interactive TV application

We use the example of an interactive TV (ITV) application to first illustrate our framework (Sections 4 and 5), and then demonstrate the applicability of this framework using a small-scale experiment using prototypes of a consumer electronics product containing this ITV application (Section 6).

The ITV application incorporates content- and knowledge-based recommendation of video content as a core feature. The prototype focuses on video recommendations that are based on prior content-classification obtained from an external recommendation service provider. However, the recommendation functionality is limited compared to more advanced approaches (Ardissono et al., 2004; Ardissono and Maybury, 2004; Ali and van Stam, 2004) since only the current search or watched video is taken into account for a subsequent retrieval of video recommendations from the service provider. No personalization takes place, that is, different people querying the same search terms or watching the same video would also get the same recommendations. The actual video content is provided via access to popular sites like YouTube and MySpace or news sources like Reuters. An essential aspect is the newly developed user interface that provides recommendations and allows for searching and browsing of video content. Furthermore, there is an additional, novel, remote pointing device for operating the ITV application.

Consider Fig. 3, which depicts a schematic view of the user interface of the ITV application. In the upper part of the screen in Fig. 3(a) one can see the video that is currently played. The video playback can be paused and resumed, and the playback window can be maximized to be displayed in fullscreen mode and brought back to the normal mode. In the lower part of the screen a number of recommendations related to the current video are displayed (using the right or the left arrow more related recommendations can be explored). Any of these recommendations can be viewed in more detail by moving the mouse pointer over it (as can be seen for the right-most recommendation) and selected for playback, after which it is displayed in the upper part of the screen. New recommendations are then retrieved and displayed according to the selected item.

The ITV application also has a search function that allows to search for video content by name and categories, which is shown in Fig. 3(b). The user can type letters of a particular search term she has on mind in the text entry field at the top. For example, in Fig. 3(b) the current search string is *ML*. With every change to the search string the ITV application updates a set of related videos (at the bottom) and a set of suggested text recommendations to refine the search (keywords below the entry field), so as to display potentially relevant search terms and search results. For example, among the displayed videos in Fig. 3(b) is a piece about *Martin Luther King* from YouTube, and a suggested keyword is *MLB Baseball*.

Fig. 4 depicts the core action space of this ITV application. The user starts off in the *Play Video* mode, where she can explore further recommended videos (*Pick Suggestion*), as illustrated in Fig. 3(a). Alternatively, the

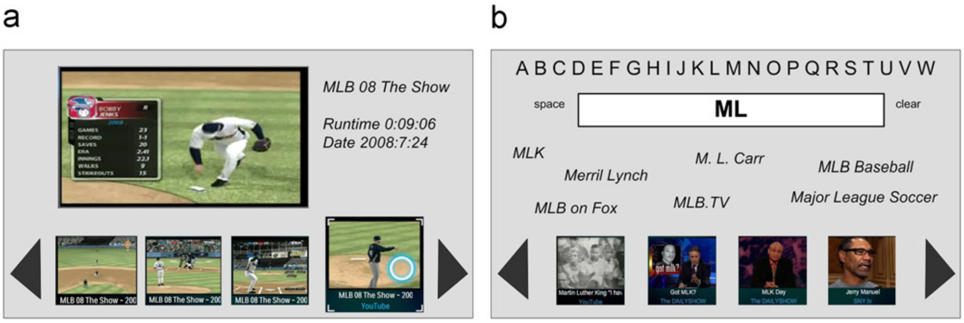


Fig. 3. Schematic user interface of an interactive TV application. (a) During video playback, recommended videos are displayed at the bottom. (b) While searching for video content, text recommendations and search results are provided.

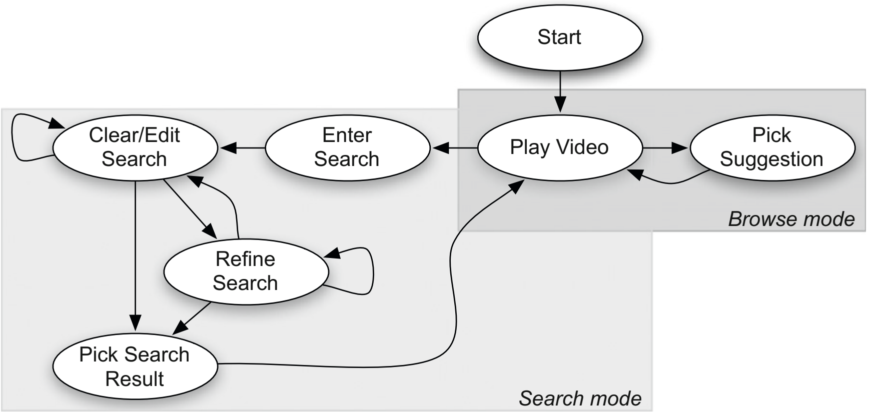


Fig. 4. The core action space of the interactive TV application. The application allows for two basic usage modes: *searching* and *browsing*.

search mode shown in Fig. 3(b) can be activated (*Enter Search*). Then, the search term can be entered (*Clear/Edit Search*). Now, there is the option to either directly select one of the videos at the bottom (*Pick Search Result*), or to click on one of the suggested keywords (*Refine Search*). Picking one of these text recommendations will replace the current search string by the selected keyword. For example, the currently entered *ML* would be replaced by the string *MLB Baseball*. As a result, only videos relating to baseball would be displayed among the search results (so, the video about *Martin Luther King* would not be offered anymore). Furthermore, more specific text recommendations appear, which enable the user to refine the search even further within this category. As soon as a concrete video is picked from the search results, the application changes into the playback mode again.

We are interested in how people will actually use this ITV application. For example, will they use the application mainly to *browse* the available video content in a playful manner, guided by the recommendations? Or are they usually looking for specific things, i.e., using the *search* mode more? Such information about the actual usage of the application can generate valuable insights. Accordingly, it may help to concentrate on the relevant product functionalities during early phases of development. For example, we might assume that those functionalities of the product that are used often are more important than the less used ones. Hence, they should be as stable and reliable as possible.

In order to obtain a deeper understanding of how users *perceive* the product (as opposed to how they *operate* it), we need additional user input. Therefore, we want to be able to ask users for their opinions or feelings. For example, it would be good if we could ask them how satisfied they are *when we recognize that they are currently using the product in browse mode*, i.e., perform event-based experience sampling. In the same way, we could then ask them for their satisfaction level *when we track that they are searching*, and then compare the results. Furthermore, we need to let users express their opinions whenever they want to. If we can connect their appraisals to the relevant actions that led them to issue the feedback, this can enhance and enrich the overall feedback and may help us to understand the context in which the feedback was provided.

In the following, we describe a framework that can be used to observe the actions of a user and, additionally, allows for user feedback. The framework captures the data in such a way that both the actions and the feedback can be correlated, and thus be analyzed in a combined fashion.

4. Approach

Direct product usage information as well as user feedback are potentially of use to a large group of professionals involved in the product development process: knowledge engineers, product managers, requirements engineers, developers, interaction designers, and other information stakeholders can all benefit from such information. Note that the members of this group, from hereon referred to as *domain experts*, have, in practice, only a rather modest influence during some phases of the product development process. However, especially for the development of innovative products, the expertise of such domain experts is needed. These experts are the target users of our approach: initially, they might have only a vague understanding about, e.g., what should be observed during the use of the product to answer open questions, or about when participants should be prompted for additional feedback, but iteratively it is possible to map issues to observable items within the product, and to finally obtain comprehensive and reliable information.

Uses of remote product monitoring have been reported before (Hartson and Castillo, 1998; Hilbert and Redmiles, 1998; Kabitzsch and Vasyutynskyy, 2004; Shifroni and Shanon, 1992). However, these approaches assume information stakeholders capable of programming and willing to use programming paradigms to achieve the sought-after data. In contrast, our approach aims at means to specify observation in a way that is doable by actual stakeholders of the collected information. Besides that, our approach towards product observation emphasizes the integration of observation functionality into the target system by using a software engineering process which is, in our opinion, necessary for widespread use. While previous work (Funk et al., 2008a, 2008b) describes our product observation approach in more detail, this paper focuses on the novel incorporation of user feedback (i.e., subjective data) by submitted surveys.

In the remainder of this section, we first provide an overview of our product usage monitoring approach (Section 4.1) and then elaborate on the role of ontologies as a semantic link between the different phases of observation and analysis in our approach (Section 4.2).

4.1. Overview

Consider Fig. 5, which depicts an overview of our approach. The system we propose is a *combination of a logging framework and a process mining tool*. On top of that, ontologies are used to link collected data items, hence, to connect observation and analysis on the information level. The figure shows that ontologies are connected to all three steps of the flow, namely, specification, observation, and analysis. Therefore, the definition and maintenance of one or more ontologies should be a concurrent task that accompanies the depicted flow.

In Fig. 5 one can see that the product to be observed is equipped with an *observation module* which has access to the so-called *hooks*. These hooks and the observation module have to be initially built into the product. Regarding the ITV application described in Section 3, the prototype machines were equipped with such an observation module before they were given to testers at home. Especially the user interface had to be instrumented with

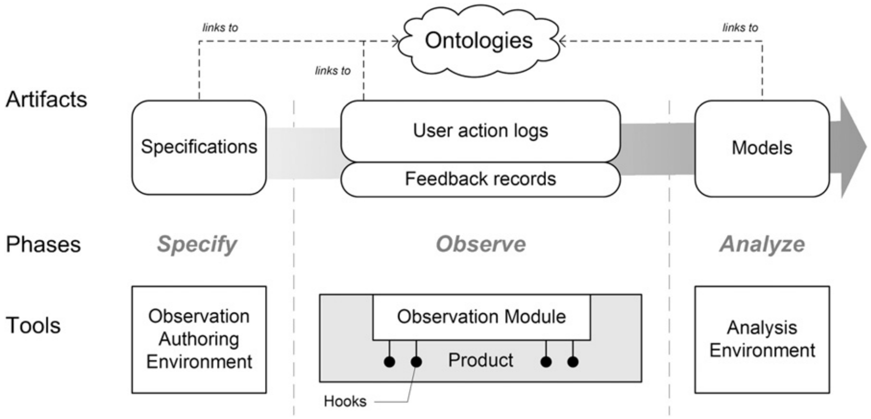


Fig. 5. Overview of our approach for product usage monitoring and analysis. Although not explicitly indicated in this figure, the phases *Specify*, *Observe* and *Analyze* are performed in an iterative way, where artifacts can be refined, added, or removed based on analysis results.

hooks that are triggered, e.g., when a video playback is started, or a recommendation is selected.

During an actual test, the following three phases are performed in an *iterative manner*: (1) During the specification phase, domain experts visually define what objective information should be observed in the product and how this information relates to the concepts from the given ontologies. Furthermore, they can specify in which situations the participants should be asked for additional feedback (i.e., to enable event-based experience sampling), and how the structure of these feedback forms should be. This task is done via an easy, but formal visual language. (2) The outcome are specification artifacts used to automatically and remotely instruct the observation modules in the products. These modules collect field data during product usage depending on their current configuration and then send it to a central data storage. Furthermore, the observation modules prompt feedback forms according to the pre-specified action patterns. Semantic links inside the specification artifacts, via the use of ontologies, enable the observation module to categorize the captured data on-the-fly. (3) In the third phase, namely the data analysis phase, collected data are processed using various (semantic) process mining techniques, which provide different views on the aggregated data. This last step offers the possibility to extract the essence out of a potentially huge data set. Furthermore, it helps to present this information in a comprehensive and directly usable way to information stakeholders.

The whole process is of a strongly iterative nature. Iterations between the three phases are not only expected but also encouraged to finally achieve the most reliable and accurate picture of product usage. For instance, during the observation phase, the domain expert might recognize an interesting usage pattern which should better be included in the set of patterns that currently trigger the event-based experience sampling. Furthermore, one could come across unexpected information that needs special treatment and hence can lead to the extension of the connected ontology to cover new concepts, both in objective and subjective data. Due to the highly flexible framework, these changes can be carried out directly and thus lead to an immediate improvement on the quality of the collected data.

4.2. Ontologies

Although the automatic processing chain from observation to analysis consists of several independent parts, a common connection is feasible by using ontologies for a semantic content structure. Ontologies (Gruber, 1993) define the set of shared concepts necessary for the analysis, and formalize their relationships and properties. Ontology elements are organized in a directed graph and there are several formalisms to build ontologies such as OWL and WSML (Bruijn et al., 2006). In our approach, ontologies bridge the gap between raw objective or subjective data that is collected from products and information that features a high-level semantic structure and is thus usable by information stakeholders. On the one hand, ontologies may represent conceptual models of the relevant product features for data collection. That is, they capture functional semantics of the product. Therefore, ontologies allow instrumenting the data collection process on the information level, abstracting from data collection mechanisms, infrastructure and synchronization issues. On the other hand, we use ontologies also to conceptualize user satisfaction and dissatisfaction levels and reasons, which can in turn be linked with the functional features of the product via means of data collected from our user-initiated surveys. As a result, using ontologies, it becomes possible to analyze subjective user feedback in connection with objective product log data.

In Fig. 6(a), an excerpt of a product-specific *ontology representing user actions* on the ITV application is shown. Note that in our approach we use the existing WSMT Toolkit, which is an editor for WSML ontologies (see also Section 5.1). One can see that concepts are organized in a hierarchical way, i.e., concepts may have one or more

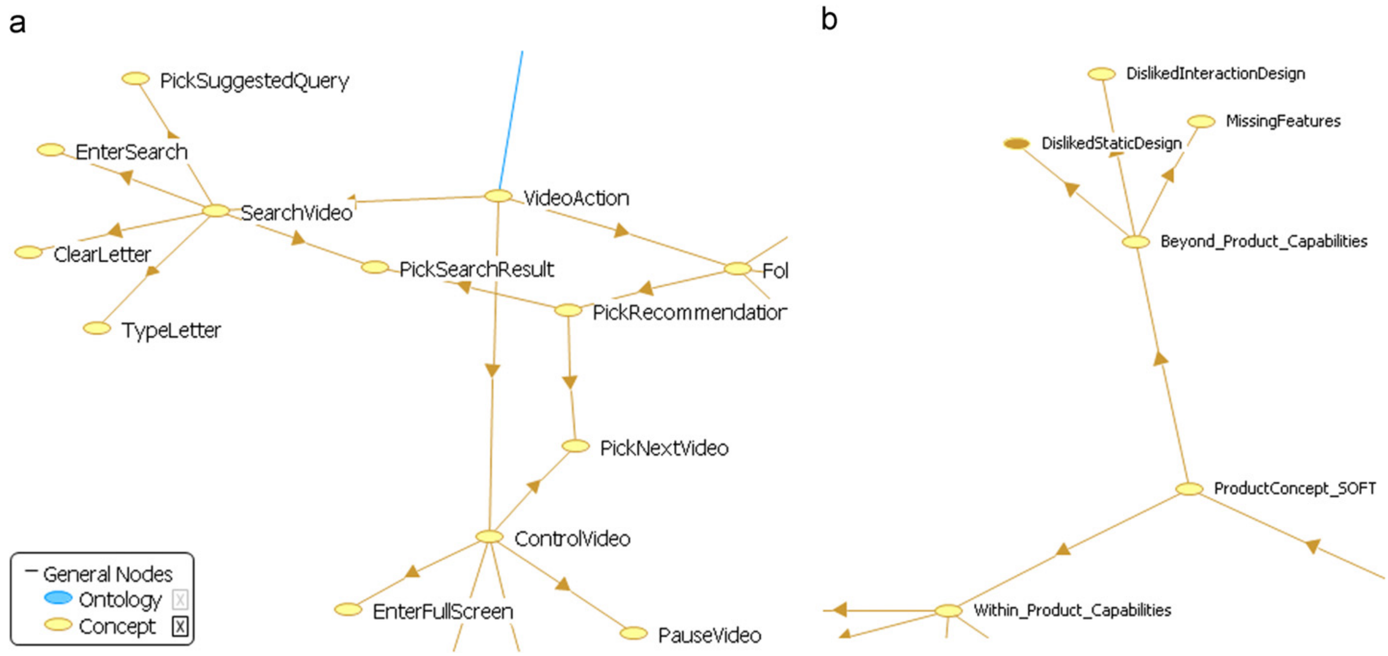


Fig. 6. Example fragments of two ontologies for the ITV example. (a) Ontology representing user actions in the ITV application (see also Section 3). (b) General consumer appraisal ontology for *user-initiated* feedback.

superconcepts. For example, the concept "PickSuggestedQuery" is a subconcept of the "SearchVideo" category, which in turn is a subconcept of "VideoAction". These subsumption relationships are very useful because they enable the analysis of the data on different levels of abstraction.

The different levels of data abstraction are illustrated in Fig. 7, where process mining has been used to automatically create a process model from the data collected in the example scenario. These process models visualize the temporal relations of the observed steps in the usage process as directed graphs.[1] The model depicted in Fig. 7(a) contains steps related to lower-level user actions (i.e., the steps refer to the leaf concepts of the ontology in Fig. 6(a)). In contrast, the model in Fig. 7(c) only contains process steps relating to higher level user actions. This depicted model provides a highly abstract view by making use of the semantic information in the log data. For example, since all four user actions "EnterSearch", "TypeLetter", "ClearLetter" and "PickSuggestedQuery" shown in Fig. 7(b) are a "VideoAction" according to our ontology (cf. Fig. 6(a)), they are not differentiated in the model shown in Fig. 7(c). Note that although the model depicted in Fig. 7(c) may seem too general, the level of abstraction can be varied at will and without the need to modify the actual data itself. This way, varying models with even heterogeneous degrees of abstraction can be created easily. For example, we can create a model that provides a detailed view on "VideoActions" but fully abstracts from "MenuActions".

Next to capturing user actions, *ontologies are used to structure user feedback*. For example, the answers to a survey used in experience sampling are categorized by an ontology (not shown here), which helps both to query the results on a more abstract level (e.g., positive feelings, or negative feelings), and to relate them to the user's actions.

The use of multiple ontologies in representing answers to a survey is also possible and can help to analyze subjective user feedback data in connection with objective product logs. In Fig. 6(b), part of a general consumer appraisal ontology is depicted, which is used to conceptualize types of positive and negative user-initiated feedback (cf. Koca and Brombacher, 2008b for the entirety of the ontology we developed). When a user-initiated positive or negative feedback survey is submitted, data are collected both about the degree and reason for the satisfaction or dissatisfaction (i.e., mapping to concepts in the general consumer appraisal ontology partially depicted in Fig. 6(b)) and also about the product feature that led to it (i.e., mapping to concepts in the product-specific user action ontology). As the concepts of "consumer appraisal" and "user action" ontologies jointly represent answers to user-initiated surveys, it is possible to enhance and validate the semantic analysis of objective product log data, which is solely based on the "user action" ontology, by linking it with the subjective user-initiated feedback about perceptions regarding a product feature, which is based on both the "user action" and the "consumer appraisal" ontologies. For example, if a certain product feature was never used until a point in time, it may be possible to identify the reason for that via data provided by user-initiated surveys: i.e., if the reason

[1] Note that it is not necessary to understand these models in detail and we, therefore, do not elaborate on the type and parameters of the shown Fuzzy Models (Günther and Aalst, 2007).

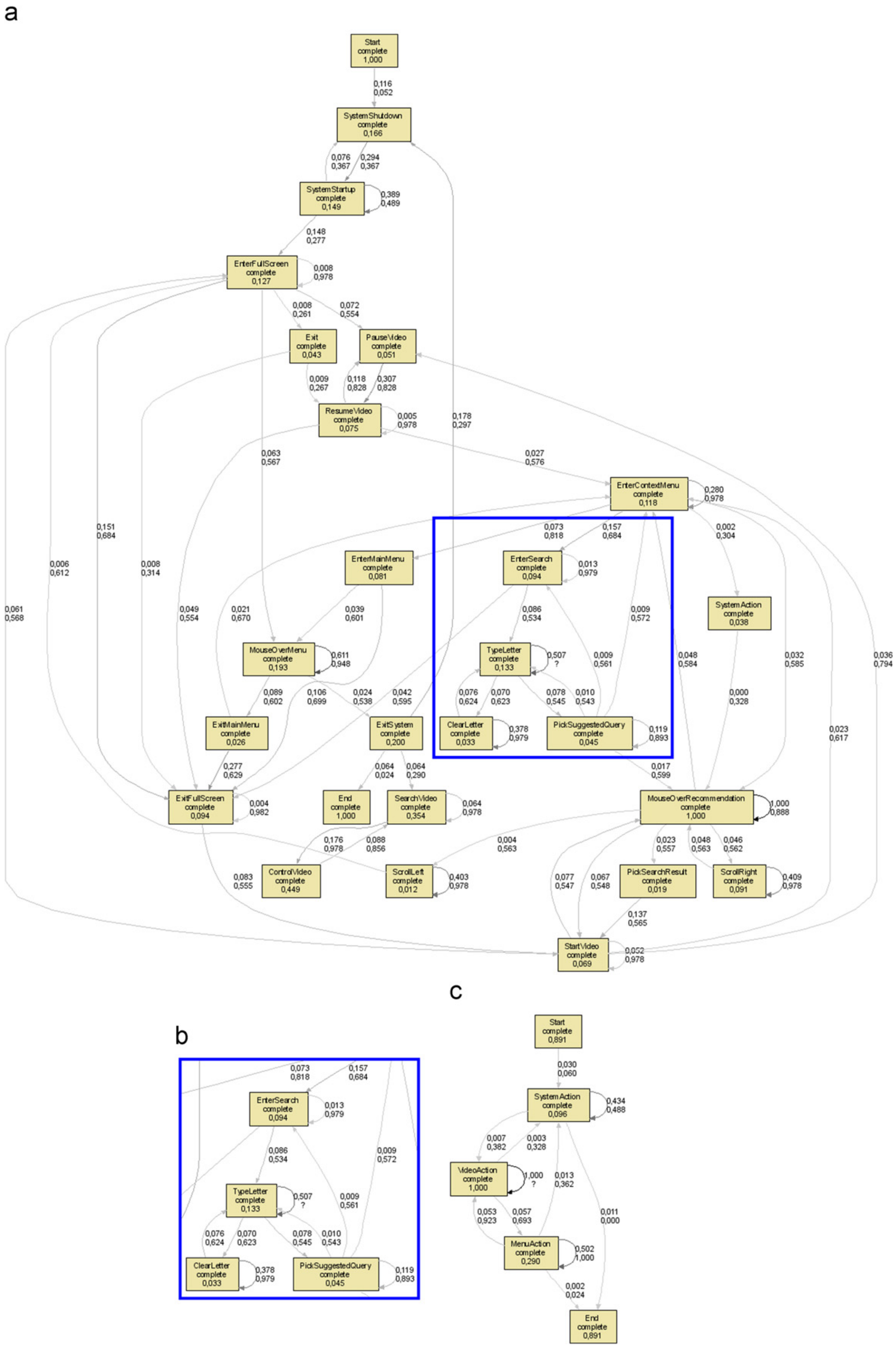


Fig. 7. Two models that were mined from the same log data, but using different abstraction levels. (a) Model mined when using all user action events in the log. (b) Zoomed-in view of framed part of the mined model in (a), showing user actions for *search mode*. (c) Model mined when using highly abstracted view on the same process.

was lack of awareness on behalf of the user, or if it was because the user was aware but was simply not motivated to make use of it, or if the user was both aware and motivated but could not get it working. Alternatively, if according to the data collected via user-initiated surveys it appears that a certain product feature is highly praised by many users, it would be important to check whether that feature is a heavily used one or not, in order to determine the impact of such a feature on the whole product.

Previously, we identified three types of ontologies that can be defined, namely *general*, *context*, and *expert* ontologies (Funk et al., 2009a). For example, the product-specific user action ontology shown in Fig. 6(a) is only applicable in the context of the considered IPTV prototype. In contrast, the consumer appraisal ontology shown in Fig. 6(b) can be re-used across experiments with different products as it relates to the domain expertise of the analyst evaluating the participants' feedback. Often, the created ontologies will be orthogonal to each other and can be maintained separately by the domain experts participating in the experiment. However, especially in larger teams working with more complex and inter-dependent ontologies the topic of ontology management can be expected to become more relevant. Questions such as "Who creates the ontologies?", "Who manages them?", and "How are changes made and propagated?" will need to be addressed by a collaborative methodology. However, this topic is beyond the scope of this paper and needs to be addressed by future research.

5. Realization

The approach outlined in Section 4 has been fully implemented by leveraging the interplay of multiple tools. Recall that the approach aims at reducing the effort that domain experts have to spend between an initial question and the acquisition of reliable data of a certain quality. Therefore, the data collection part is entirely automated and the phases of specification and analysis become the domain experts" only concerns. Besides that, structured and meaningful data, which is directly usable, can be collected in a much faster way than with traditional data collection methods.

Fig. 8 illustrates the architecture of our realized system. As can be seen, the architecture contains three main parts: (1) the D'PUIS component, (2) the actual database (DB) where the events relating to the observed usage of products are stored, and (3) the process mining analysis suite (Alves de Medeiros et al., 2008; van der Aalst et al., 2007a, 2007b; Günther and van der Aalst, 2006).

The idea of using semantics to perform analysis of processes is not new (Casati and Shan, 2002; Hepp et al., 2005; O'Riain and Spyns, 2006). Our analysis approach is based on previous work on semantic process mining techniques (Alves de Medeiros et al., 2007, 2008). Process mining techniques can provide valuable insights into a real-life process based on data registered in event logs and have been successfully applied in practice (van der Aalst et al., 2007a). *Semantic* process mining enhances the analysis by leveraging semantic information (Alves de Medeiros et al., 2007).

Since this paper focuses on explaining how to set up the link between the objective and subjective data, and how to use this connection to enhance the analysis, we will not provide a detailed explanation about the DB and the process mining analysis suite. The interested reader is referred to Funk et al. (2009a). The remainder of this section focuses instead on the D'PUIS component. Section 5.1 introduces the D'PUIS component, and Section 5.2 shows an example of how to create two specification artifacts for the collection of objective and subjective data: an observation specification and a linked survey.

5.1. D'PUIS

We have developed D'PUIS (Dynamic Product Usage Information System) (Funk et al., 2008a, 2008b) as a

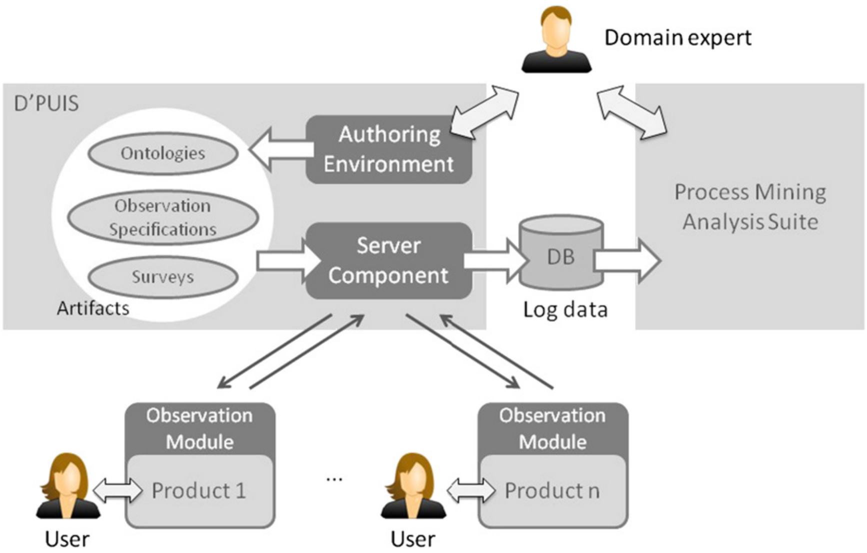


Fig. 8. Overview of tools and artifacts involved in the realization of the approach.

platform-specific realization of the specification and observation approach depicted in Fig. 5. Regarding the approach outlined in Section 4, the D'PUIS framework covers all tasks from the specification phase to the actual data collection phase. A precondition for the high degree of automation in the data collection phase is a set of specification artifacts that define what data are collected in the product during usage, and how these data are processed to acquire meaningful information (i.e., ontologies, observation specifications, and surveys). The D'PUIS framework provides means to create these artifacts during the specification phase, and it uses them to automatically collect data.

The framework is split up into three components that are connected via a network infrastructure: the authoring environment, the server component, and the observation module that is integrated into the system under observation (cf. Fig. 8).

The latter two components realize a flexible abstraction layer for low-level logging matter and data communication issues. In the authoring environment, observation specifications are created and then distributed via the server component to a number of products equipped with observation modules. These modules dynamically execute the specifications and, over time, collect data according to what has been specified. These data are transmitted back to the server component, where they are accessible to analysis and visualization tools.

The authoring environment is the common interface for information stakeholders to specify the data collection process in such a way that it leads to meaningful and structured data that can be analyzed by these domain experts. The environment is structured as an integrated development environment (IDE), using the renowned Eclipse platform.[2] As for the specification artifacts introduced earlier—ontologies, observation specifications, and surveys—the authoring environment provides tools to create and manage these artifacts in an integrated manner.

Ontology: Ontologies provide semantic structure (cf. Section 4.2), and are created by a third-party editor (WSMT), which is integrated into the authoring environment. This editor allows for visual and textual ontology creation in WSML (Bruijn et al., 2006) language.

Observation specification: The second class of specification artifacts—observation specifications—connects data sources inside the product to the semantic concepts given in the ontologies. The specification of items that will be observed and categorized by semantic concepts determines the richness of the obtained data. Besides that, patterns of user actions, which can be recognized as sequences of semantic events, can themselves be captured in semantics. This allows to combine different atomic events into more complex events. Note that complex events, again, can be combined into more abstract events, thus allowing for arbitrary levels of abstraction.

Survey: In contrast to observation specifications which are used to extract *objective* data from products, the third class of artifacts—surveys—are used to acquire *subjective* data from users. Surveys are designed by means of a conceptual description language, and are subsequently rendered and shown directly on the product. Again, ontology concepts add meaning to surveys and allow for a combined analysis of objective and subjective data items, which both have been structured by additional semantics.

Observation specifications and surveys are created within newly developed editors that also enable linking between these artifacts and ontologies (cf. Section 5.2). The outcome is a bundle of artifacts, composed of observation specifications and surveys, which are semantically linked by ontological concepts.

5.2. *Usage example*

Observation specifications are created with a visual language, supported by a newly developed editor. Language elements represent timing structures, abstract data sources within the product or the platform, filtering and processing blocks, and means for semantic annotation of data. In the context of supporting event-based experience sampling, we will focus on the visual language as the place where links between certain usage patterns and the surveys are initially constructed.

Consider the following example: shortly after the experiment started, we have analyzed the objective data collected so far and mined the process models shown in Fig. 7. Looking at the user action model fragment depicted in Fig. 7(b), we observe that sometimes participants continue to edit their search term (“TypeLetter”) after they already refined their search by suggested keywords (“PickSuggestedQuery”). We had expected that, after potentially refining the search multiple times, they would pick one of the search results (but not go back to edit the search term). This unexpected behavior could have two reasons: (a) the user is searching for something specific (and cannot find it) and (b) the user might simply enjoy browsing the keywords (without the actual goal to find something specific).

Previously, we have defined patterns to perform event-based experience sampling in situations where the user is *searching* and *browsing* (cf. Section 3). Figs. 9(a) and (b) illustrate these usage patterns in the context of the overall user action space. We are interested in situations where users pre-dominantly use the product on one of these interaction modes. Therefore, we chose to consider participants as “using the product in search mode” when they enter the search, type in a keyword, potentially refine the results, and then pick a result *three times after another*. Similarly, we assume that the “product is used in browse mode” when a video is played and another suggested video is picked subsequently, again, three times in a row. However, also single usage sequences or more flexible characterizations could have been chosen.

[2] Eclipse Platform—www.eclipse.org

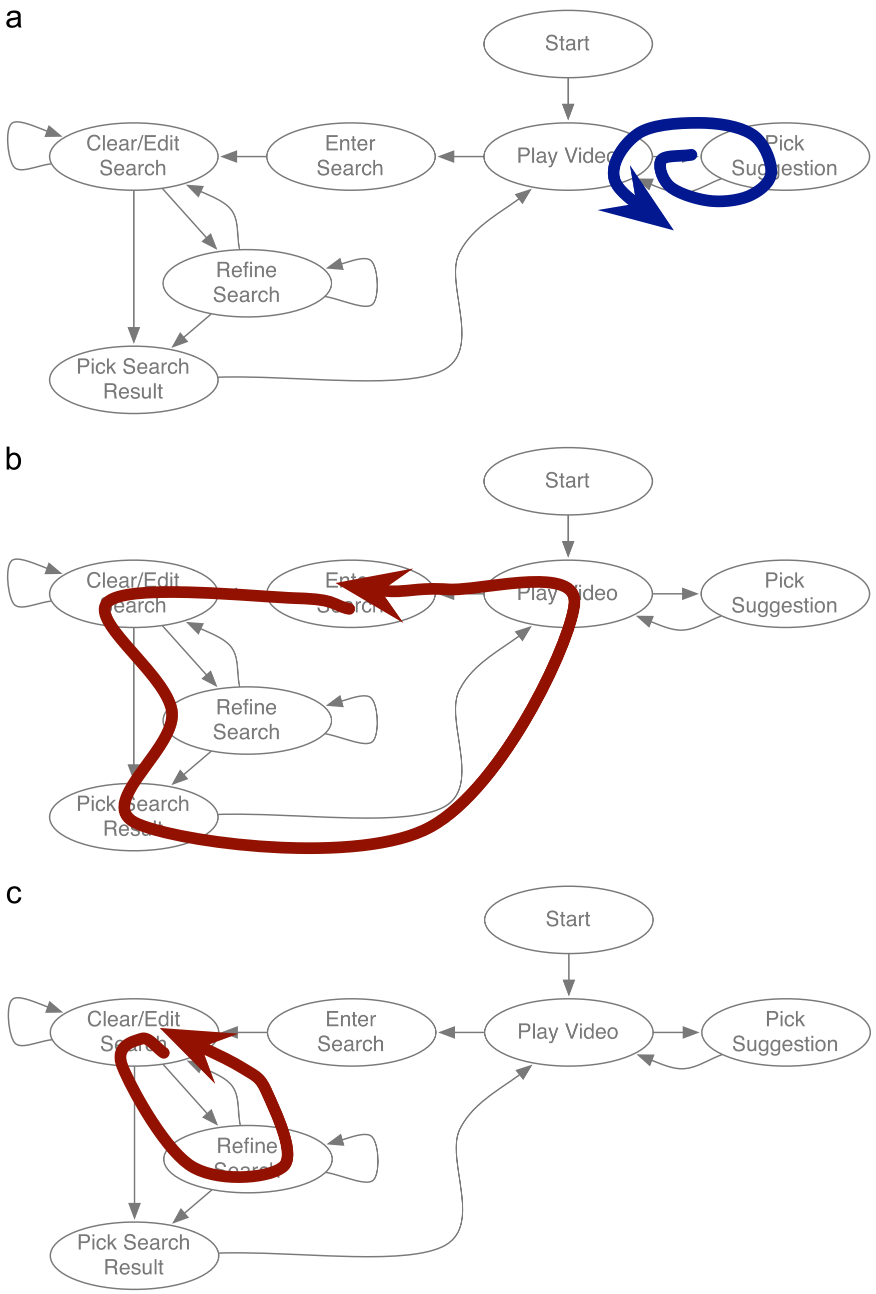


Fig. 9. Illustration of the three usage patterns identified for the ITV example. (a) "Browse" pattern: videos are watched by browsing recommended videos only. (b) "Search" pattern: new videos are searched for explicitly, watched, the next video is searched, etc. (c) "Explorative Search" pattern: search terms are provided and refined by suggested keywords, but none of the search results is actually selected to watch the video. Instead, the search words are continuously edited. We want to know: Is the user looking for something that cannot be found? Or are the suggested keywords browsed for fun?.

Now, we are also interested in this newly discovered interaction pattern and want to find out why users might *continuously* return to edit the search term *after having refined their search but without actually picking a search result*. See Fig. 9(c) for an illustration of this usage pattern. That is, we are interested in whether they remain in the search area of the application to look for something specific (i.e., indeed more in a search mode), or whether they might be exploring the search results for different keywords in a playful manner (i.e., in a rather explorative way). Therefore, we want to extend the system by this new pattern, which we call *explorative search*. Using our framework, this can be done without interrupting the experiment (i.e., at run time). This means that after specifying and deploying the new pattern, the survey used for experience sampling appears also in situations where the explorative search pattern matches.

In the following, we explain how such a pattern for event-based experience sampling can be formulated using the example of the newly discovered pattern "explorative search". To be able to connect such a pattern to a survey, the visual language has been extended to allow for

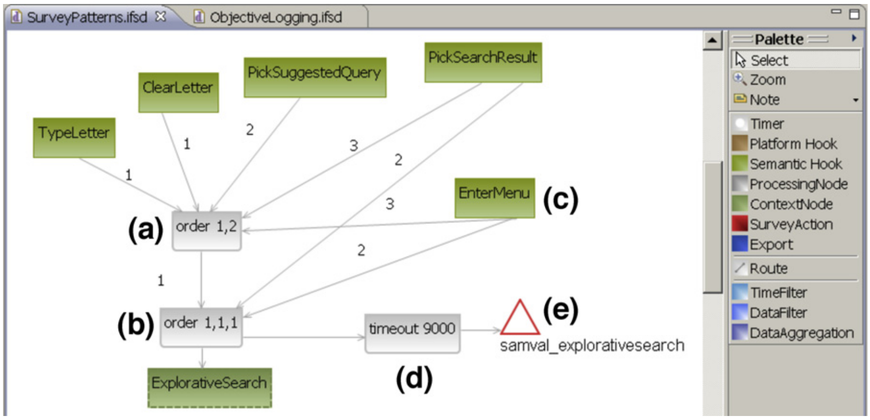


Fig. 10. Visual editor for observation specification with an example showing the specification of the *explorative search* pattern used for event-based experience sampling.

triggering of surveys based on the occurrence of certain data events. Fig. 10 shows a screenshot of the editor together with an observation specification. This example shows five semantic event sources (green boxes at the top) that are initially obtained by mapping low-level product events to ontology concepts (not shown). A pattern of semantic events is then created by connecting the event sources to *order* processing blocks (grey boxes). These simply specify the order of events coming in via one of the numbered routes. Once the given pattern is completed, the processing block sends an event to its successors in the event chain.

In the example shown in Fig. 10, the pattern of (a) first "TypeLetter" or "ClearLetter", then "PickSuggestedQuery" (but not "PickSearchResult"), (b) three times in a row, and (c) without an interrupting "EnterMenu" event is defined. The occurrence of the events "PickSearchResult" and "EnterMenu" *break* the pattern because they are connected via routes whose numbers do not appear in the corresponding *order* processing blocks. To avoid a survey popping up every time this pattern occurs, a *timeout* processing block (d) is inserted in the event chain and set to 15 minutes (i.e., 9000 seconds). This element works like a gate that blocks events for a certain amount of time. It passes one event and closes again until the timeout is over. At the end of the event chain, a survey trigger block (e) is connected (red triangle on the right) to the respective survey. Fig. 11 shows how this survey looks like in our example. A detailed description of the execution semantics of the visual language is beyond the scope of this paper and it can be found in Funk et al. (2008b).

The definition of a survey is achieved with a simple textual markup language. This language provides common survey elements such as multiple choice or single choice fields, text entry and check boxes, and—more importantly—also ways to connect the acquired information to semantic concepts from one or more ontologies. By connecting the survey contents to semantic concepts, semantic events are triggered upon answering the survey. This way, *subjective survey answer events can then be used as event sources for observation specifications in the same way as the objective user action events* discussed before (cf. green boxes at the top of Fig. 10). Surveys defined in this information-centric language are rendered against a customizable survey template into a HTML page. However, given a different template, specialized representations of data collection tools can easily be realized. On the front-end, the surveys can readily be viewed and adapted at will if and whenever necessary (see Section 6.2 for screenshots of surveys created by the D'PUIS editor).

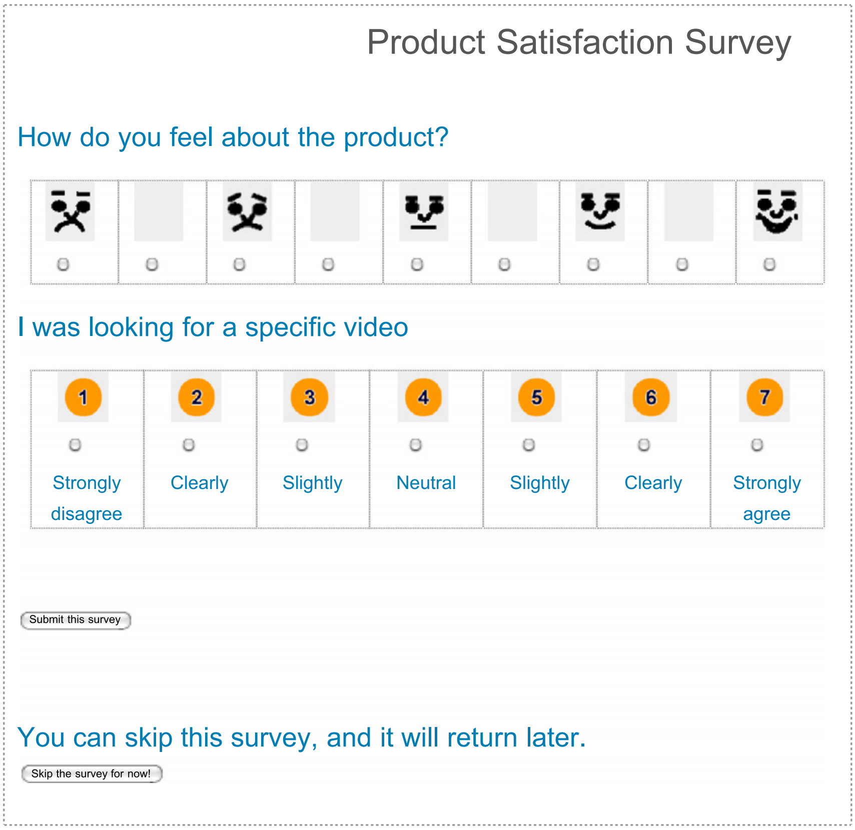
Finally, note that any defined pattern can be re-used in another specification. This way, arbitrary layers and combinations of objective and subjective patterns can be created. For example, whenever the *explorative search* pattern defined in Fig. 10 is matched, a new semantic "ExplorativeSearch" event is triggered (besides the timeout node). So, we could now use this higher-level behavioral pattern and include it in a new specification as a *source event*. For example, we could define that only in situations where participants use the product in explorative search mode *and* they provide feedback of a certain type, then we want to present them with an additional questionnaire.

6. Case study

To validate the applicability of our framework, we have performed a case study using a set of prototypes equipped with the ITV application described in Section 3. In the remainder of this section, we describe the concrete goals (Section 6.1), the setup and methods applied (Section 6.2), and the results (Section 6.3) of this experiment.

6.1. Goals of the study

As already mentioned, one major goal is to apply our framework in a real experiment to test the applicability of the tool in a natural context. More specifically, we are interested in the opportunities of combining the analysis of objective data (i.e., user actions) and subjective data (i.e., explicit feedback of the participants). For this, we have



Product Satisfaction Survey

How do you feel about the product?

I was looking for a specific video

You can skip this survey, and it will return later.

Fig. 11. Survey measuring the pleasure and the goal-orientedness as perceived by a participant while using the product either in *browse*, *search*, or *explorative search* mode.

applied experience sampling based on particular action contexts (Section 6.1.1) and enabled user-initiated feedback at any point in time (Section 6.1.2).

6.1.1. *Event-based experience sampling*

The prototype as described in Fig. 4 provides two main access mechanisms for ITV content, *Browsing*, where a set of recommendations is being offered and updated when the user clicks for a different video item, and *Searching*, where the user searches for a specific video item.

Prior literature has suggested that, within the entertainment computing domain, the success of a recommendation service will not relate as highly to the accuracy of the recommendation, but rather to the overall affective state of the user (Hassenzahl and Ullrich, 2007; Leong et al., 2008), in that users may use recommendation services merely as a shuffle mechanism. Prior to conducting the field study we formulated a number of hypotheses. We expected that the *browse* and *search* functionalities will induce distinct modes of usage; while searching, users' behavior would be dominated by their need to achieve their goal, i.e., to find the specific video item. On the contrary, browsing would induce much more explorative behaviors that are driven by curiosity. We expected that while searching would result in higher effectiveness in searching for specific content, it would result in less pleasant affective states. Secondary, we expect that the search functionality would be used as a means to explorative interaction, also referred to as serendipitous behavior (Leong et al., 2008). In this case, search functionality is used as a form of browsing, explorative behavior that may result in useful findings. Thus, explorative searching is expected to be related with lower levels of goal-orientedness and effectiveness.

6.1.2. *User-initiated feedback*

The richest source of use context information tends to be qualitative and subjective as it involves individual users, their expectations prior to use, and their perceptions during use. Despite the various ways of getting this kind of information from users in the user-centered engineering domain, it is notoriously known to be challenging to systematically extract and quantify user feedback, and even more so to scale this process up within industrial settings where many users, many products, and many complaints or feedback co-exist (Koca and Brombacher, 2008a; Koca et al., 2007). An important goal we aim at with the use of user-initiated feedback forms, or the so-called “Thumbs Up” and “Thumbs Down” surveys, is to systematically elicit meaningful subjective information from users as to their satisfaction and dissatisfaction levels and causes related to the use of the product that may affect its long-term use and adoption. These surveys are structured primarily by use of the generic consumer appraisal ontology that we had developed earlier on Koca and

Brombacher (2008b), which serves to capture the *conceptual misalignments between explicit or implicit product capabilities and user expectations*.

In achieving our stated end goal, we formulated some hypotheses. We expected that the concepts in the consumer appraisal ontology can be embedded in the "Thumbs Up" and "Thumbs Down" surveys, such that users can, by filling out the surveys, accurately and consistently use these concepts to encode their subjective judgements regarding their use experience. To validate and verify their encoding, we also asked for detailed textual explanation of their feedback. Another reason for the textual description of feedback was to not lose the richness of information they provide, in case of requiring further analysis, e.g., during joint analysis with objective use patterns. A potential expectation was also to identify the recurring types of (encoded) feedback in relation with time, i.e., when different phases of use of a product is taken into account. However, due to the short duration of this study, this was deemed not feasible. Last but not least, we expected to enable deeper insights into subjective product perception in relation to objective product use, by explicitly asking users to *subjectively* interlink their feedback (i.e., concepts from the generic consumer appraisal ontology) with concepts from the specific user action ontology (Fig. 6(a)).

6.2. *Overall setup*

The case study is based on the use of a novel ITV application that features a recommendation functionality (cf. Section 3) at the participants' homes. The prototypes hosting the ITV application are realized based on off-the-shelf hardware and Windows Vista operating system. At home, the product prototype is connected to the TV screen, and it is operated with the supplied wireless keyboard and a novel pointing device. Before shipping the prototypes to the participants, we instrumented these prototypes with our evaluation framework described in Sections 4 and 5. The prototype application providing a user interface for browsing and searching internet videos is a Flash application running locally on all test machines. This application was instrumented for observation by placing few function calls to log the generation of relevant events or data, and by connecting the application to a reconfigurable observation module running on the same machine via local sockets. Leveraging the availability of this observation module and of the application source code, the instrumentation of the prototype application was achieved within four hours. Under similar conditions, comparable instrumentation efforts can be expected.

During the experiment, extensive data about users' interaction with the recommendation functionality is collected and analyzed. At the same time, surveys enable users to express individual opinions during product use. Overall, eight such pre-configured prototypes for home use were sent out to participants and were used during 10 days. All family members were allowed to use the prototypes and participate in the study. The participants also received instructions on how to set up and use the device, and how to provide feedback via two different kinds of surveys:

On the one hand, feedback could be initiated by the participants themselves. For this purpose, two buttons on the keyboard that accompanies the prototype have been highlighted by a "Thumbs Up" and a "Thumbs Down" sticker. When one of these buttons is pressed, a corresponding survey appears on the screen of the prototype, prompting the positive or negative feedback. On the other hand, surveys assessing the participant's degree of satisfaction were raised automatically in the case that a particular use pattern was recognized from the participant's actions (i.e., event-based experience sampling). To avoid bothering the participant with too many of these surveys, a maximum of three automatic surveys was raised within 15 minutes (cf. Section 5.2 for an example of the participant's action pattern leading to an automatic survey).

In the following, we describe these surveys for event-based experience sampling (Section 6.2.1) and user-initiated feedback (Section 6.2.2) in more detail.

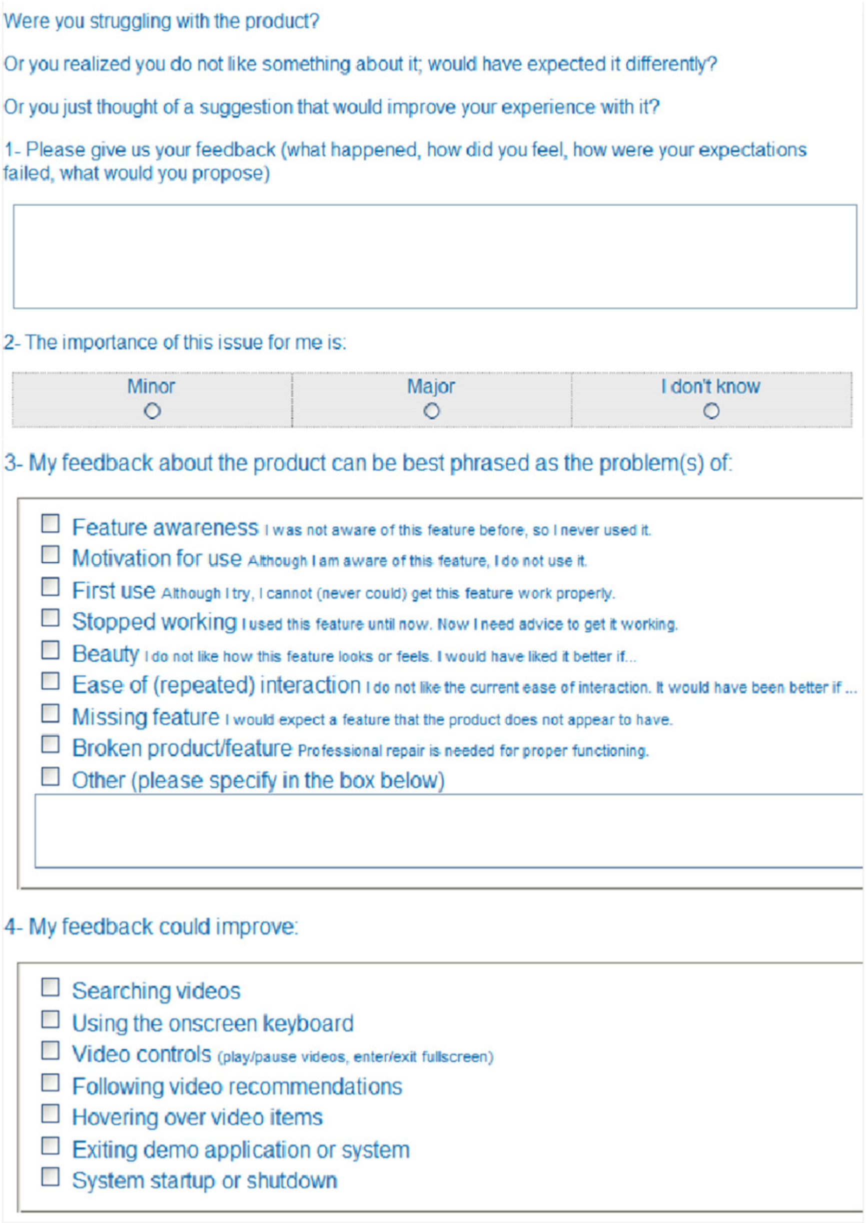
6.2.1. *Survey for event-based experience sampling*

Using the visual editor of D'PUIS, we have formulated three interaction patterns resembling *search*, *browse*, and *explorative search* behavior, respectively (see Section 5). In each of these situations a survey is triggered, which measures the two aspects that were described in Section 6.1.1: (a) Emotion-Valence (Bradley and Lang, 1994) to measure the overall affective state within that specific interaction encounter, and (b) goal-orientedness, an index of goal oriented behavior using a 7-point Likert scale. Fig. 11 depicts a screenshot of this survey.

6.2.2. *Surveys for user-initiated feedback*

During product creation, domain experts typically want to know about both the weaknesses of a product as well as the strengths of it so as to identify its unique selling proposition. Therefore, we employed two different surveys that appear on the product's screen as soon as positive (i.e., "Thumbs Up") or negative (i.e., "Thumbs Down") feedback is initiated by the user. A screenshot of the "Thumbs Down" survey is depicted in Fig. 12. The "Thumbs Up" survey is very similar, but tailored to capture positive feedback.

Both user-initiated surveys for collecting positive and negative feedback have the following four parts: (1) description of the user's feedback about a product feature, (2) degree of satisfaction or dissatisfaction with the feature described by the user in his/her feedback, (3) reason of the satisfaction or dissatisfaction with the feature described by the user in his/her feedback, and (4) actions on the product that get directly influenced by the feature described by the user in his/her feedback. While part (1) of user-initiated surveys is important to capture rich descriptive data that may be needed for detailed analysis later on, parts (2) and (3) capture categorical data that map to various concepts in



Were you struggling with the product?

Or you realized you do not like something about it; would have expected it differently?

Or you just thought of a suggestion that would improve your experience with it?

1- Please give us your feedback (what happened, how did you feel, how were your expectations failed, what would you propose)

2- The importance of this issue for me is:

Minor	Major	I don't know
○	○	○

3- My feedback about the product can be best phrased as the problem(s) of:

- □ Feature awareness I was not aware of this feature before, so I never used it.
- □ Motivation for use Although I am aware of this feature, I do not use it.
- □ First use Although I try, I cannot (never could) get this feature work properly.
- □ Stopped working I used this feature until now. Now I need advice to get it working.
- □ Beauty I do not like how this feature looks or feels. I would have liked it better if...
- □ Ease of (repeated) interaction I do not like the current ease of interaction. It would have been better if ...
- □ Missing feature I would expect a feature that the product does not appear to have.
- □ Broken product/feature Professional repair is needed for proper functioning.
- □ Other (please specify in the box below)

4- My feedback could improve:

- □ Searching videos
- □ Using the onscreen keyboard
- □ Video controls (play/pause videos, enter/exit fullscreen)
- □ Following video recommendations
- □ Hovering over video items
- □ Exiting demo application or system
- □ System startup or shutdown

Fig. 12. Screenshot of "Thumbs Down" (user-initiated) survey.

the general "consumer appraisal" ontology partially depicted in Fig. 6(b). The last part of the survey captures also categorical data, but that data maps to concepts in a product-specific ontology, namely the "user action" ontology depicted in Fig. 6(a), which represents user actions of interest on the ITV application. As the concepts of "consumer appraisal" and "user action" ontologies jointly represent answers to (the last three parts of) user-initiated surveys, it is possible to enhance and validate the semantic analysis of objective product log data, which is solely based on the "user action" ontology, by linking it with the subjective user-initiated feedback about *perceptions* regarding a product feature, which is based on both the "user action" and the "consumer appraisal" ontologies.

6.3. Results

Over a period of 10 days, the eight machines generated 15 328 events in total. This includes low-level system events relating to system startup and shutdown, as well as user action events and feedback events (both from event-based experience sampling and user-initiated feedback). Each of these events is linked to one or more concepts in an ontology. Fig. 13 visualizes the level of activity of the participants over time. Each row corresponds to one particular machine in the experiment, and each event is reflected by a dot at the corresponding time (many dots can be placed on top of each other if events were generated in close time proximity). One can see that the level of usage varied from a single-use to regular usage (almost every

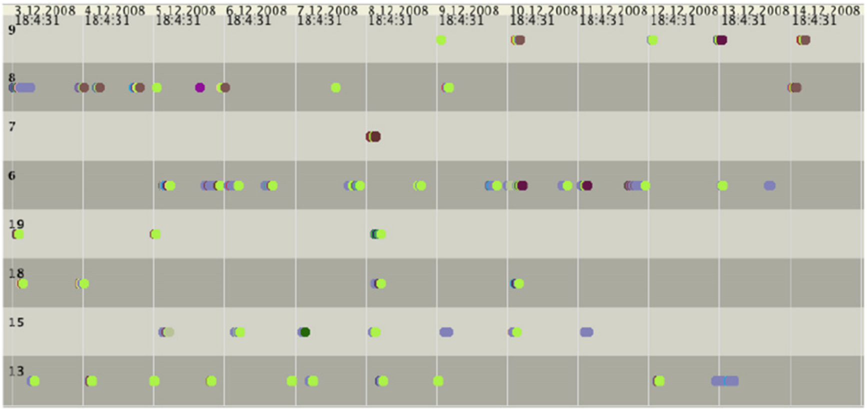


Fig. 13. Overview of the activity of the participants (each row corresponds to one particular machine used in the experiment) over time.

day). An amount of 377 events was generated by the least active participant, whereas 4040 events were generated by the most active participant.

Already in Section 4.2 (see also Funk et al., 2009a) we have shown that—based on this data—we can discover process models showing typical usage patterns on varying levels of detail. In the following, we describe the results of the event-based experience sampling (Section 6.3.1) and the results of the user-initiated feedback (Section 6.3.2) in more detail.

6.3.1. *Results of event-based experience sampling*

Figs. 14(a) and (b) respectively illustrate the average pleasure (i.e., valence) and goal-orientedness in the three different usage modes: browsing, searching, and explorative searching. *The initial hypothesis that browsing and searching will induce distinct usage modes was confirmed*, as one can note in Fig. 14(b) that browsing scores significantly lower on *goal-orientedness. No significant difference on goal-orientedness was found however between searching and explorative searching, as shown by the* 95% *confidence intervals on the mean goal-orientedness.*

Overall, *browsing of recommendations was the least preferred interaction mode*, with explorative searching the most preferred. This is contrary to our initial expectation; while we expected users' behavior to be dominated by a need for curiosity and exploration, users scored significantly less on pleasure when browsing than when searching for specific video items. To some extent, the dislike of the provided browsing experience can be attributed to its limited functionality as mentioned before: only by searching the full set of movies can be explored. *Explorative search*, initially thought of as a means to explore diverse content through a search functionality, *appeared to be enhancing the effectiveness of the search functionality*. Fig. 14(b) illustrates a (non-significant) increase in goal-orientedness, eventually resulting in more pleasant affective states (cf. Fig. 14(a)).

6.3.2. *Results of user-initiated feedback*

During the study, a total number of 23 surveys were submitted by users of seven (out of eight) machines. From these 23 surveys, 18 reported negative feedback (i.e., "Thumbs Down" survey) and five reported positive feedback (i.e., "Thumbs Up" survey). In Fig. 15(a), the distribution of the numbers of surveys submitted via each machine over time is shown. Each particular survey can, if needed, be further evaluated at the desired abstraction level in the context of the user activities performed around the time of the submission (cf. Fig. 13). Based on user responses to part (4) of the surveys, the "Thumbs Down" surveys were 44% about the "SearchVideo" feature, 28% about the "ControlVideo" feature, and 28% about the "TypeLetter" feature. While based on event-based experience sampling results, searching appeared to be an activity stimulating more pleasure than browsing, based on "Thumbs Down" survey results, "SearchVideo" appeared to be the most problematic feature. This may be potentially important information for improvement of the search feature, given the fact that 28% of "Thumbs Down" surveys reported problems with "TypeLetter".

Fig. 15(b) is a close-up look on the submitted "Thumbs Down" surveys. Due to the relatively small set of data acquired from the study, it is not reliable to try to associate the submitted few problem types (on various phases-of-use) with time.[3] Therefore, the time factor is excluded in Fig. 15(b). Instead, the problem types on various phases-of-use (e.g., awareness problem, motivation-for-use problem, first use problem, adaptability problem, etc.), as they relate to concepts of the consumer appraisal ontology (cf. Fig. 6(b)) are depicted in relation to the user groups of all prototype machines. On the whole, surveys were submitted 3 times by teenagers (i.e., 13–20), 5 times by early adults (i.e., 20–45), 13 times by mature adults (i.e., 45–65), and 2 times by users from an unknown age group. Submissions were made 6 times by females, 15 times by males, and 2 times by users who have not

[3] Such an association is possible to do with a larger data set, as presented in Koca et al. (2009).

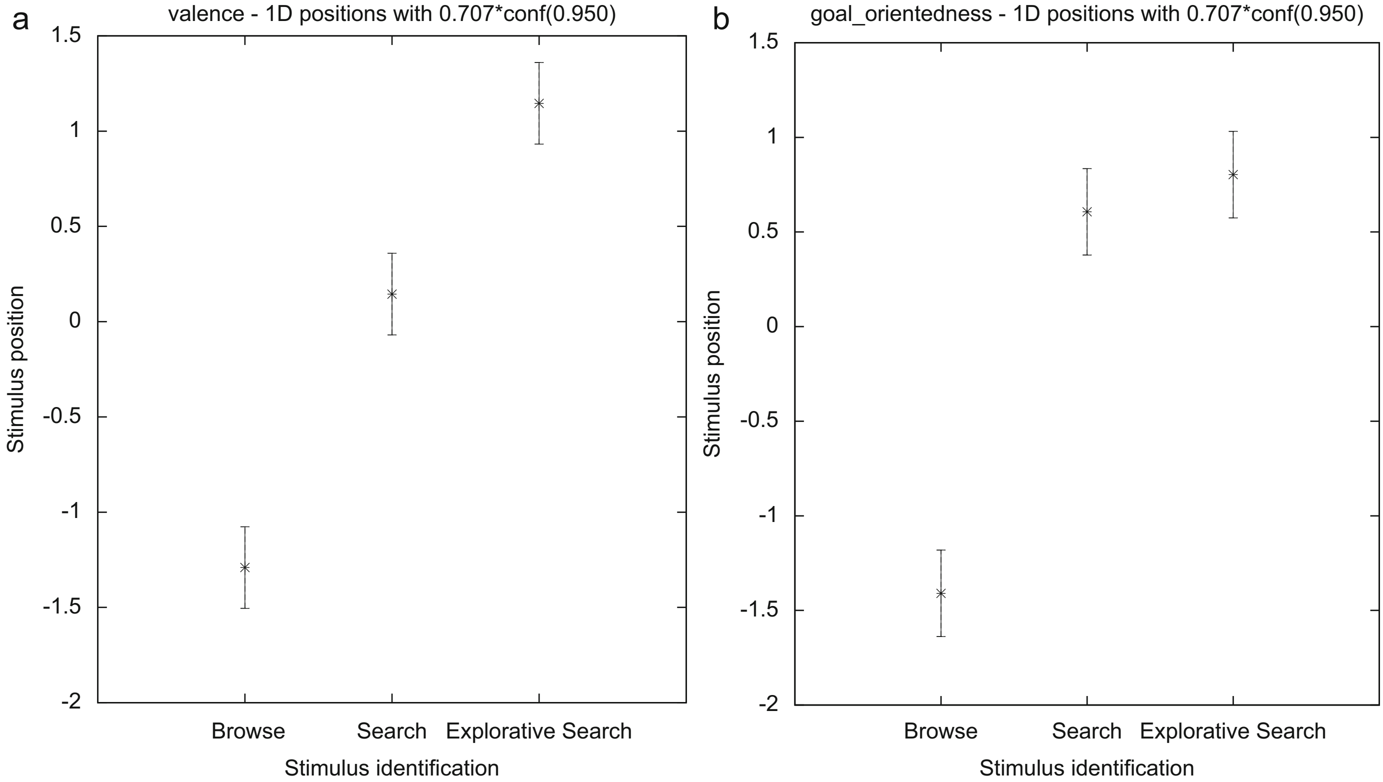


Fig. 14. Results of the event-based experience sampling. (a) Valence. (b) Goal-orientedness.

reported their gender. Since all submissions were made anonymously, for submissions from the same machine it is not possible to track if the reported groups consisted of different individuals or the same person, which is another reason the analysis here is not based on phases-of-use of a machine. For this study, the focus was on demonstrating the feasibility of- and possibilities with our framework to jointly evaluate product usage and user experience data. But for a follow-up study in which the focus is more on the data, multi-user systems and single-user systems can be evaluated differently, based on specific needs and goals. Based on all "Thumbs Down" submissions, the most dominant type of problem is "interactivity problem," 50% of which are again reported to be about the "SearchVideo" feature. First-use problems and static-design problems are the second most prominent issues reported by users. It should be noted in viewing Fig. 15(a) and (b) that users can report multiple problems in submitting a "Thumbs Down" survey, which is why these two figures cannot be directly correlated. For instance, in Fig. 15(a), it appears that during the study only one survey has been submitted from machine-13, which is a "Thumbs Down" survey. In Fig. 15(b) however, two problems, namely, a static design problem and an interactivity problem, have been reported. With more data, it is possible to do χ^2 tests to statistically explore relationships between all categorical variables of interest, such as the correlation between certain age groups and experienced problem types, or the correlation between the respective time period of occurrence of certain problem types or certain delight types, which are all automatically captured via user responses to parts (2) and (3) and hence relate to the respective concepts of the consumer appraisal ontology.

The textual feedback descriptions of users (i.e., part (1) of "Thumbs Up" or "Thumbs Down" surveys) were manually checked to validate and verify users' accuracy and consistency in encoding their feedback using the embedded concepts of ontologies by selecting the appropriate choices provided in parts (2)–(4) of the surveys. Furthermore, we could extract the particular user action context in which each user-initiated survey was submitted. For example, consider Fig. 16, which depicts a process model mined with the Fuzzy Miner (Günther and Aalst, 2007). This process model shows the aggregated actions that participants performed five minutes before submitting a "Thumbs Down" survey indicating that they "Disliked the Static Design". As stated earlier, the user response on a positive or negative survey can, if needed, be further evaluated at the desired and relevant abstraction level (by aggregating or disaggregating certain actions) in the context of the user activities performed around the time of the submission (cf. Fig. 13) to achieve a more reliable understanding of the situatedness of certain interactions.

One can see that directly before reporting a "Disliked Static Design" feedback (cf. highlighted node "Disliked-StaticDesign"), users were all typing search terms (i.e., "TypeLetter"). This indicates that the way the edit functionality is integrated within the application may be the reason the participants were dissatisfied. In fact, from

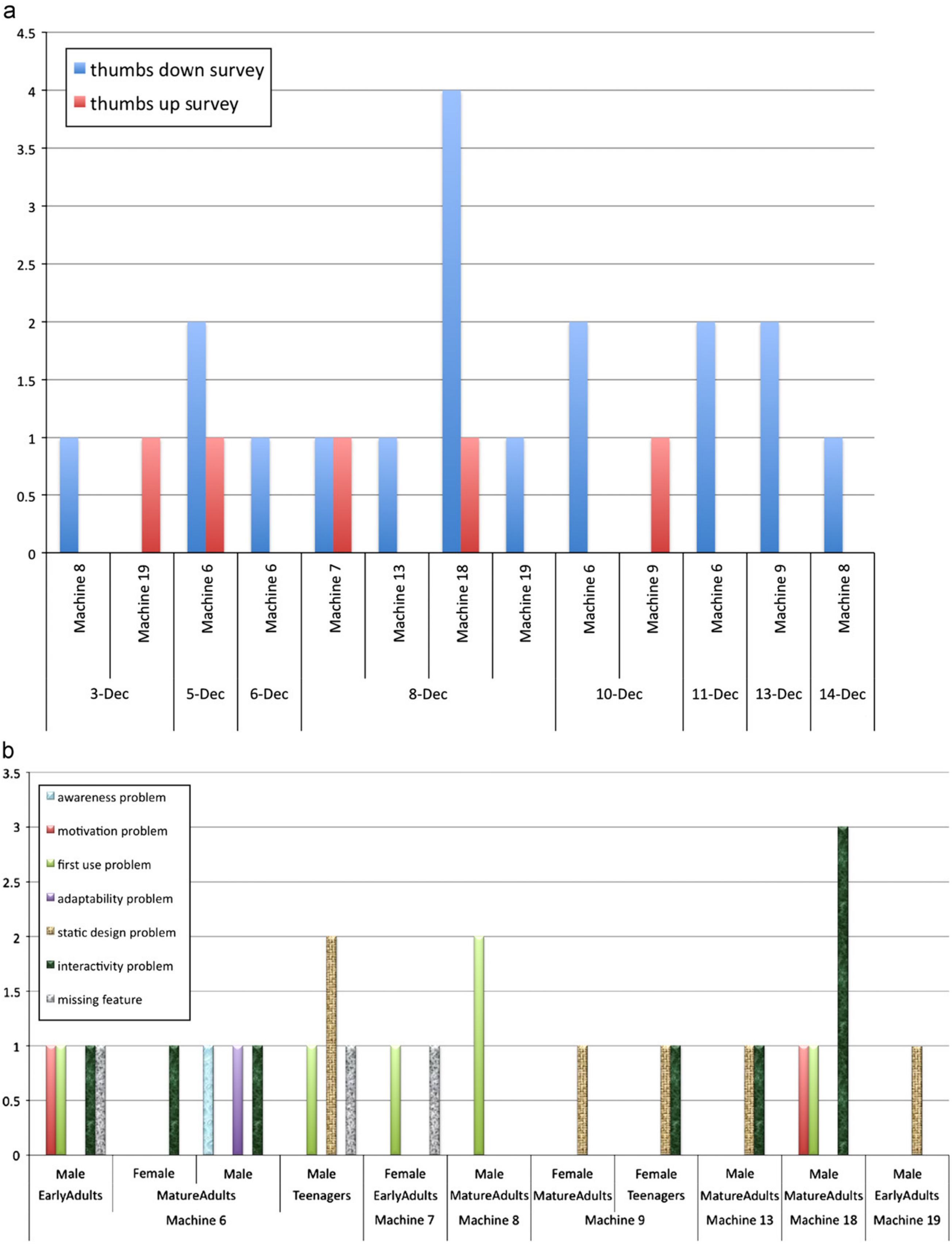


Fig. 15. Results of the user-initiated feedback. (a) Numbers of “Thumbs Up” and “Thumbs Down” surveys submitted by users over time. (b) Distribution of the types of problems identified by users of each machine through the “Thumbs Down” survey.

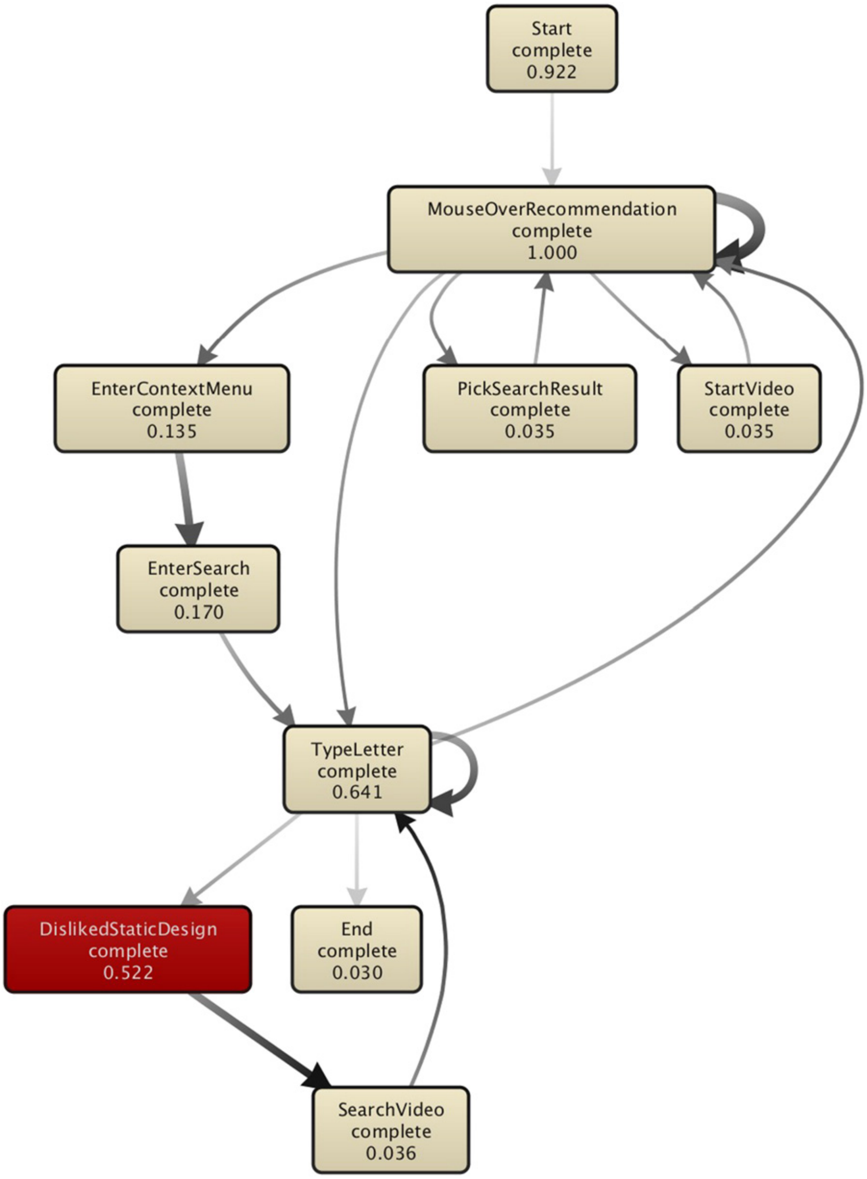


Fig. 16. Mined model showing actions of users five minutes before they indicated that they dislike the static design (i.e., static design problem in Fig. 15(b)). Note that users have reported their dissatisfaction after typing in queries (i.e., after using the pointer to type in queries). This indicates that this feature may be a good candidate for improvements.

the textual feedback description of the issues reported within the corresponding "Thumbs Down" surveys, we were able to trace that it was indeed a problem for some users that the edit functionality was disliked due to its design-induced limitations. For example, one of these users has reported "… *Very useless searching program* !!!!!!!!!!!!! *Why can't you just type what you want to search*??" and another one "…*pointer works bad on* 4 1/2 *meter distance*". These examples nicely illustrate that information about the survey initiation context can enhance the evaluation of subjective user feedback.

7. Discussion

Our framework has been presented in the context of an interactive TV application case with the main intention to demonstrate its automated systematic provision of an initial roadmap for facilitating further evaluation of the recommendation functionality. Therefore, it is not our ambition here to flesh out the various ways in which the results from this case can be further analyzed to reveal answers to particular questions. Rather, we demonstrate the feasibility and applicability of our approach, and indicate the possibility to do more specialized analyses *iteratively* based on the achieved results. For example, based on the data presented in Section 6.3.1, browsing of recommendations is the least preferred interaction mode. In principle, it is possible to probe this information deeper to find out the causes via text mining users' textual feedback submitted within "Thumbs Up" and "Thumbs Down" surveys for certain keywords, or via introducing more targeted event-based pop-up questionnaires triggered by certain usage activity patterns, or via interviewing the users. Such narrower-focused iterations can then reveal if the specific causes are related to the user interface, or the experience of using the novel interaction device, etc. However, doing this kind of exhaustive iterations over the initial results from the presented case study is peripheral to our focus here.

Also, the amount of subjective data gathered via surveys was limited, making detailed statistical analyses less meaningful or even feasible at all. With more data, our ontological approach to semantically analyzing consumer appraisal provides unforeseeable practical findings, as partially reported in Koca et al. (2008), and an invaluable resource of longitudinal data for monitoring actual product adoption in the field. The potential strength of monitoring longitudinal data on consumer appraisal via our ontological approach is demonstrated in another case study with the Apple iPhone^{TM}, where user feedback reported during the first four weeks of use after purchase has been identified with the concepts of our consumer appraisal ontology (Koca et al., 2009). While such results are not apparent from our analyses here, based on the iPhone case, we are convinced that with more data acquired via surveys, it is possible to achieve results that explicitly demonstrate the effect of time in passing through the phases-of-use of the tested product or prototype.

The proposed framework provides an alternative to traditional field data collection practices such as behavioral observation and contextual inquiry (Holtzblatt and Jones, 1993). Such approaches are of critical importance to the evaluation of recommender systems as the psychological impact of recommender systems is studied in their natural environments. However, traditional field studies are limited in at least three main respects.

First, the effort in conducting such field studies is substantially higher as compared to lab-based studies. The amount of participants being studied is thus minimal and one cannot generalize over the whole population about potential impacts of a recommender system. The insights are primarily qualitative in nature. In contrast, our proposed approach enables the observation of large amounts of participants since observation and inquiry is automated.

Secondly, the sampling strategy of traditional field studies is inherently low. Rare interactions might thus be missed as observations are limited to small fractions of one's interaction with the system. Our proposed framework captures the whole spectrum of one's interaction with the system. This may provide valuable insights into unexpected interactions which may further lead to the definition of additional surveys that can be triggered the next time a user performs the exact or a similar interaction sequence. Moreover, this semi-automated field study approach may be combined with contextual interviews. Observational data may be used in creating scenarios that can simulate an exact response to the system and thus, rare interactions may be studied extensively.

Thirdly, the proposed approach enables the study of long-term effects of recommender systems. Longitudinal studies are increasingly laborious as the time of study increases and thus, longitudinal studies of recommender systems are only rarely seen in practice. The proposed framework scales down the complexity of longitudinal studies as observation is automated and the strategy for experience sampling, i.e., when and what to measure, may be modified as research questions narrow down over time.

Experience sampling has been for long considered as the gold standard in momentary psychological assessment (Kahneman et al., 2004). Traditional approaches to experience sampling are limited to random sampling strategies. This paper proposed a novel framework for event-triggered experience sampling through usage observation. Similar endeavors can be found in the field of ubiquitous and pervasive computing (Intille et al., 2003; Froehlich et al., 2007; Khan et al., 2008) where sampling is triggered from the physical sensors, e.g., identification of location through WiFi networks (Khan et al., 2008). Such approaches, however, due to the different context of use, typically do not assess time-dependencies between actions that are typical in observing product usage. The proposed framework is the first, to our knowledge, to tackle this issue and provide experience sampling for product usage monitoring.

In our approach we heavily rely on the possibility to iteratively refine and change the way data are collected and interpreted at run time. However, it is a current technical limitation that the layer of hooks (the instrumented parts of the observed application) is static. That is, while routing, processing, and semantics can be flexibly configured and influenced, the set of data sources in the prototype application remains stable after release. Nevertheless, it can be expected that this limitation can be overcome in the future with reflective architectures or model-driven development flows (Funk et al., 2009b).

Finally, note that our framework is not limited to the evaluation of recommender systems. It can be applied for the usage and experience monitoring of other types of products and deployed applications as well. In fact, the evaluation of objective information (user actions) and subjective information (user perceptions) is a necessity to test the effectiveness of any user interface design:

- On the one hand, user interface design often entails the formulation of *usage scenarios* that should be supported by the product or application. Using process mining techniques, it is possible to compare actual usage behavior with these envisioned usage scenarios.
- On the other hand, *non-functional goals* that should be achieved are formulated during the user interface design. For example, the product should be exciting for the user, or perceived as easy to use. Conclusions about such non-functional aspects can only be obtained by feedback from the user.

For example, in Hofstra (2009) a comparison of ideal usage scenarios and the actual user behavior during a usability test of a television was performed. However, the data collection approach in Hofstra (2009) requires manual annotation steps and is thus time-consuming and error-prone. The framework presented in this paper automates the data collection and, therefore, allows for much easier evaluations of usage scenarios. In addition, the achievement of initial interface design goals can be verified by collecting feedback from users about their perceptions. By a combined analysis of objective and subjective information as discussed in this paper, not only discrepancies between anticipated and actual usage behavior can be revealed, but it can also help to detect correlations between usage and the achievement of design goals.

8. Conclusion

Much work in the evaluation of recommender systems employs objective measures of recommendation effectiveness to assess their user acceptance (Herlocker et al., 2004). In this paper we argued that users'' satisfaction with a recommender system, which has been repeatedly cited as the overall measure of quality of recommender systems (Herlocker et al., 2004; Konstan and Riedl, 1999), will be contextually situated, influenced not only by the effectiveness of the recommendation, but also by the exact goals that users formulate while using the system.

In this paper we proposed an evaluation framework that aims at capturing such contextual judgments in the field. This framework is motivated by findings in psychological research that highlights that retrospective assessments of individuals' affective experiences are filled with biases, as emotions cannot be stored in memory but only reconstructed from episodic and semantic information that is accessible. When individuals fail in recalling episodic information from memory, general beliefs about how they should respond in certain occasions are used in reconstructing the felt emotion, thus leading to systematic biases (Robinson and Clore, 2002). The framework employs a modification of the Experience Sampling Method (Hektner et al., 2007), which is considered as the gold standard in momentary assessment of emotional experiences (Kahneman et al., 2004), where researchers may define interaction sequences that trigger surveys.

The major strengths of this event-triggered experience sampling approach are the freedom to instrument the data collection process at any moment after the test products have been placed in the field, and the ability to collect both objective and subjective data linked by semantic meta-data. This enables the experimenter to combine an understanding of how the product is used in the field with an understanding of how users feel when carrying over specific interaction sequences (which traditionally was only possible through experiments in controlled environments). In addition the presented approach may result in an iterative evaluation procedure where the experimenter deepens the research questions during the course of the study by analyzing and redefining data collection repeatedly. It may further be combined with retrospective interviews that are grounded on users' behaviors and opinions during the field study.

In a field study we aimed at testing this evaluation framework. The framework was implemented in an interactive TV set-top box prototype device that incorporates content-based and knowledge-based recommendation of video content. Eight devices were given to different families and were used for a period of 10 days. The device allowed for alternative ways of navigating through video content: *browsing* or *searching*. Our interest was to elicit the experienced emotion at the exact moment they perform these two alternate interactions. Contrary to our initial hypothesis, users were more satisfied when employing searching behaviors than when browsing for video content. While this study was limited in some respects, our aim in this paper was to test the deployment of the system in the field. Future work will further test the framework in more extensive studies both in terms of time and number of participants.

The quality of recommendation has been so far assessed through mostly objective measures, whereas related research shows that satisfaction is the overall measure. In this paper, we have provided a framework that supports the situated assessment of recommendation quality in field studies and we showed that the quality of recommendation will be affected by the mode of interaction, e.g., its goal-orientedness (e.g., searching versus browsing). In the future, evaluation practices should highlight more the subjective and situated nature of the perceived quality of recommender systems. Our framework is a first system that aims at capturing such situated judgments in the field. We showed that, using the framework and the collected objective and subjective data, the user experience of recommender systems can be evaluated in real-life usage scenarios.

Acknowledgments

This work is being carried out as part of the *Managing Soft Reliability in Strongly Innovative Product Creation Processes* project (see www.softreliability.org for further information on the project), sponsored by the Dutch Ministry of Economic Affairs under the IOP-IPCR program. Some of the authors are also supported by the European project SUPER (www.ip-super.org). Furthermore, the authors would like to thank Philips for the generous donation of prototype machines which were used in the experiment.

References

Ali, K., van Stam, W., 2004. Tivo: making show recommendations using a distributed collaborative filtering architecture. In: KDD '04: Proceedings of the Tenth ACM SIGKDD International Conference on

Knowledge Discovery and Data Mining. ACM, New York, NY, USA, pp. 394–401.

Alves de Medeiros, A., Pedrinaci, C., van der Aalst, W., Domingue, J., Song, M., Rozinat, A., Norton, B., Cabral, L., 2007. An outlook on semantic business process mining and monitoring. In: Meersman, R., Tari, Z., Herrero, P. (Eds.), OTM Workshops (2). Lecture Notes in Computer Science, vol. 4806. Springer, Berlin, pp. 1244–1255.

Alves de Medeiros, A.K., van der Aalst, W.M.P., Pedrinaci, C., 2008. Semantic process mining tools: core building blocks. In: Proceedings of the 16th European Conference on Information Systems (ECIS).

Ardissono, L., Maybury, M., 2004. Preface: special issue on user modeling and personalization for television. User Modeling and User-Adapted Interaction 14 (1), 1–3.

Ardissono, L., Kobsa, A., Maybury, M., 2004. Personalized Digital Television: Targeting Programs to Individual Viewers, Human-Computer Interaction Series, vol. 6. Kluwer Academic Publishers, Norwell, MA, USA.

Bolger, N., Davis, A., Rafaeli, E., 2003. Diary methods: capturing life as it is lived. Annual Reviews in Psychology 54 (1), 579–616.

Bradley, M., Lang, P., 1994. Measuring emotion: the self-assessment Manikin and the semantic differential. Journal of Behavior Therapy and Experimental Psychiatry 25 (1), 49–59.

Bruijn, J., Lausen, H., Polleres, A., Fensel, D., 2006. The web service modeling language WSML: an overview. In: Sure, Y., Domingue, J. (Eds.), ESWC. Lecture Notes in Computer Science, vol. 4011. Springer, Berlin, pp. 590–604.

Casati, F., Shan, M., 2002. Semantic analysis of business process executions. In: Eighth International Conference on Extending Database Technology (EDBT '02). Springer, London, UK, pp. 287–296.

Chin, J., Diehl, V., Norman, K., 1988. Development of an instrument measuring user satisfaction of the human-computer interface. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, New York, NY, USA, pp. 213–218.

Csikszentmihalyi, M., Larson, R., 1992. Validity and reliability of the experience sampling method. In: The Experience of Psychopathology: Investigating Mental Disorders in their Natural Settings, pp. 43–57.

Froehlich, J., Chen, M., Consolvo, S., Harrison, B., Landay, J., 2007. MyExperience: a system for in situ tracing and capturing of user feedback on mobile phones. In: Proceedings of the 5th International Conference on Mobile Systems, Applications and Services. ACM Press, New York, NY, USA, pp. 57–70.

Funk, M., van der Putten, P.H.A., Corporaal, H., 2008a. Specification for user modeling with self-observing systems. In: Proceedings of the First International Conference on Advances in Computer-Human Interaction, pp. 243–248.

Funk, M., van der Putten, P.H.A., Corporaal, H., 2008b. Model interpretation for executable observation specifications. In: Proceedings of the 20th International Conference on Software Engineering and Knowledge Engineering, Knowledge Systems Institute, pp. 785–790.

Funk, M., Rozinat, A., Alves de Medeiros, A., van der Putten, P., Corporaal, H., van der Aalst, W., 2009a. Improving product usage monitoring and analysis with semantic concepts. In: ISTA '09: Proceedings of the 2009 International Conference on Information Systems Technology and its Applications, pp. 190–201.

Funk, M., Hoyer, P., Link, S., 2009b. Model-driven instrumentation of graphical user interfaces. In: Proceedings of the 2nd International Conference on Advances in Computer-Human Interaction, pp. 19–25.

Gruber, T., 1993. A translation approach to portable ontology specifications. Knowledge Acquisition 5 (2), 199–220.

Günther, C., Aalst, W., 2007. Fuzzy mining: adaptive process simplification based on multi-perspective metrics. In: Alonso, G., Dadam, P., Rosemann, M. (Eds.), International Conference on Business Process Management (BPM 2007), vol. 4714, pp. 328–343.

Günther, C.W., van der Aalst, W.M.P., 2006. A generic import framework for process event logs. In: Eder, J., Dustdar, S. (Eds.), Business Process Management Workshops, vol. 4103, pp. 81–92.

Hartson, H., Castillo, J., 1998. Remote evaluation for post-deployment usability improvement. In: Proceedings of the Working Conference on Advanced Visual Interfaces, pp. 22–29.

Hassenzahl, M., Ullrich, D., 2007. To do or not to do: differences in user experience and retrospective judgments depending on the presence or absence of instrumental goals. Interacting with Computers 19 (4), 429–437.

Hektner, J.M., Schmidt, J.A., Csikszentmihalyi, M., 2007. Experience Sampling Method: Measuring the Quality of Everyday Life. Sage Publications Inc., Beverley Hills, CA.

Hepp, M., Leymann, F., Domingue, J., Wahler, A., Fensel, D., 2005. Semantic business process management: a vision towards using semantic web services for business process management. In: IEEE International Conference on e-Business Engineering (ICEBE 2005), pp. 535–540.

Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T., 2004. Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems 22 (1), 5–53.

Hilbert, D.M., Redmiles, D.F., 1998. An approach to large-scale collection of application usage data over the internet. In: Proceedings of the 20th International Conference on Software Engineering, Kyoto, Japan, pp. 136–145, ISBN:0-8186-8368-6.

Hofstra, P.P.H.J., 2009. Analysing the effect of consumer knowledge on product usability using process mining techniques. Master's Thesis, Eindhoven University of Technology, Department of Industrial Design, Eindhoven, The Netherlands.

Holtzblatt, K., Jones, S., 1993. Contextual inquiry: a participatory technique for system design. In: Participatory Design: Principles and Practice, pp. 180–193.

Intille, S., Rondoni, J., Kukla, C., Ancona, I., Bao, L., 2003. A context-aware experience sampling tool. In: Conference on Human Factors in Computing Systems. ACM, New York, NY, USA, pp. 972–973.

Kabitzsch, K., Vasyutynskyy, V., 2004. Architecture and data model for monitoring of distributed automation systems. In: 1st IFAC Symposium on Telematics Applications in Automation and Robotics, Helsinki, pp. 19–24.

Kahneman, D., Krueger, A., Schkade, D., Schwarz, N., Stone, A., 2004. A survey method for characterizing daily life experience: the day reconstruction method.

Karapanos, E., Hassenzahl, M., Martens, J.-B., 2008. User experience over time. In: CHI '08: CHI '08 Extended Abstracts on Human Factors in Computing Systems. ACM, New York, NY, USA, pp. 3561–3566.

Khan, V., Markopoulos, P., Eggen, B., IJsselsteijn, W., de Ruyter, B., 2008. Reconexp: a way to reduce the data loss of the experiencing sampling method. In: Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services. ACM, New York, NY, USA, pp. 471–476.

Koca, A., Brombacher, A.C., 2008a. Extracting "broken expectations" from call center records: Why and how. In: Czerwinski, M., Lund, A.M., Tan, D.S. (Eds.), CHI '08: CHI '08 Extended Abstracts on Human Factors in Computing Systems. ACM Press, New York, NY, USA, pp. 2985–2990.

Koca, A., Brombacher, A.C., 2008b. User-centered analysis of feedback operations for quality improvement in new product development. In: van der Vaart, T., van Donk, D.P. (Eds.), Proceedings of the 15th International EurOMA Conference (EurOMA 2008): Tradition and Innovation, EIASM.

Koca, A., Schouwenaar, A.J., Brombacher, A.C., 2007. Field-feedback in innovative product development: a comparison of two industrial approaches. In: Fernandes, A., Teixeira, A., Jorge, R.N. (Eds.), Proceedings of the 14th International Product Development Management Conference, EIASM, pp. 657–668.

Koca, A., Funk, M., Karapanos, E., Rozinat, A., van der Gaarden, N., 2008. Grasping product pragmatics: a case with internet on TV. In: Darnell, M., Masthoff, J., Panabaker, S., Sullivan, M., Lugmayr, A. (Eds.), Proceedings of uxTV 2008 International Conference on Designing Interactive User Experiences for TV and Video, ACM

International Conference Proceeding Series. ACM Press, New York, NY, USA, pp. 193–202.

Koca, A., Karapanos, E., Brombacher, A., 2009. "Broken expectations" from a global business perspective. In: CHI '09: Proceedings of the 27th International Conference on Human Factors in Computing Systems. ACM, New York, NY, USA, pp. 4267–4272.

Konstan, J., Riedl, J., 1999. Research resources for recommender systems. In: CHI '99 Workshop Interacting with Recommender Systems.

Larson, R., Csikszentmihalyi, M., 1983. The experience sampling method. New Directions for Methodology of Social & Behavioral Science 15, 41–56.

Leong, T., Howard, S., Vetere, F., 2008. Choice: abdicating or exercising? In: CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems. ACM, New York, NY, USA, pp. 715–724.

McNee, S.M., Riedl, J., Konstan, J.A., 2006. Being accurate is not enough: how accuracy metrics have hurt recommender systems. In: CHI '06: CHI '06 Extended Abstracts on Human Factors in Computing Systems. ACM, New York, NY, USA, pp. 1097–1101 http://doi.acm.org/10.1145/1125451.1125659.

O'Riain, S., Spyns, P., 2006. Enhancing the business analysis function with semantics. In: Meersman, R., Tari, z. (Eds.), OTM Conferences (1). Lecture Notes in Computer Science, vol. 4275, Springer, Berlin, pp. 818–835.

Robinson, M., Clore, G., 2002. Belief and feeling: evidence for an accessibility model of emotional self-report. Psychological Bulletin 128 (6), 934–960.

Russell, J., 2003. Core affect and the psychological construction of emotion. Psychological Review-New York 110 (1), 145–172.

Shifroni, E., Shanon, B., 1992. Interactive user modeling: an integrative explicit-implicit approach. User Modeling and User-Adapted Interaction 2 (4), 331–365.

van der Aalst, W., Reijers, H., Weijters, A., van Dongen, B., Alves de Medeiros, A., Song, M., Verbeek, H., 2007a. Business process mining: an industrial application. Information Systems 32 (5), 713–732.

van der Aalst, W.M.P., van Dongen, B.F., Günther, C.W., Mans, R.S., Alves de Medeiros, A.K., Rozinat, A., Rubin, V., Song, M., Verbeek, H.M.W., Weijters, A.J.M.M., 2007b. ProM 4.0: comprehensive support for real process analysis. In: Kleijn, J., Yakovlev, A. (Eds.), Application and Theory of Petri Nets and Other Models of Concurrency (ICATPN 2007), Lecture Notes in Computer Science, vol. 4546, Springer, Berlin, pp. 484–494.

W3C: Web Ontology Language (OWL) 〈http://www.w3.org/2004/OWL/〉.

WSMT: Web Service Modeling Toolkit 〈http://sourceforge.net/projects/wsmt〉.