Robert Lokaiczyk, Andreas Faatz, Benedikt Schmidt, Matthäus Martynus:
Useability versus Adaptation - Approaches to Higher Level Context Detection. In: Jörg Cassens and Anders Kofod-Petersen and Marielba Silva Zacarias and Rebekah K. Wegener: Proceedings of the Sixth International Workshop on Modeling and Reasoning in Context, p. 25-36, August 2010.
http://ceur-ws.org/Vol-618/paper3.pdf.

Useability versus Adaptation – Approaches to Higher Level Context Detection

Robert Lokaiczyk[1], Andreas Faatz[2], Benedikt Schmidt[2], and Matthäus Martynus[3]

[1] TU Darmstadt,
Rundeturmstr. 10
D-64283 Darmstadt, Germany**
[2] SAP Research,
Bleichstr. 8,
D-64285 Darmstadt, Germany
[3] SAP AG
Dietmar-Hopp-Allee 16
D-69190 Walldorf
Germany

Abstract. While the research in detection of context features - even cognitive ones like the user's goal - steps forward continuously, it is often focused on the quality of detection. But since context often finds use in context adaptive systems that proactively adapt to the user's need, research should always consider the resulting benefit of context for the user in any context-based application. This paper presents the results of a case study which a) evaluates algorithms to detect three subtypes of user goals in knowledge work and b) focuses on useability issues, that have a direct impact for a measurable improvement of the work results. Our work includes a quantitative evaluation like task completion time improvement and qualitative aspects (e.g. intrusiveness of the system). Finally, we draw conclusions from our measurements, especially on how modeling of an adaptive approach at the workplace might take place.

1 Introduction

While low level context features like location or temperature can be determined easily by appropriate sensors, higher level context is not an explicit state that could be measured by sensors. In personal information management systems or context-aware e-learning scenarios, such cognitive context features include the user's goal, which is in best case a conscious state of mind expressible by the user. Neither is it feasible to measure the user's goal with a physical apparatus, nor is it adequate to continuously ask the user for his/her goal. Actively interrupting the user during his/her work reduces the useability of a contextualized information or e-learning system. In most cases frequent pop-ups will immediately lead the user to a refusal of the overall software system, no matter how helpful the

** research conducted while the author was with SAP Research Darmstadt recommended resources or actions based on the context are. Useability overrules context-aware adaptation here.

Consequently, research should not only focus on unintrusive context detection with high precision and accuracy but also on adequate mechanisms in order to use this context information for applications in provable helpful way. Useability of context-aware applications might also suffer in many ways while reconfiguration of GUI elements or ever changing lists of recommendations might distract the user from the working task and actually lead to longer task completion times (efficiency) or even lower task completion ratios (effectiveness). As a conclusion context based systems should not only be benchmarked by the quality of their context detection mechanisms but primarily by the benefit that context information is gives the user.

In order to analyze effects of the design of context adaptive systems to useability and work results in a corporate knowledge work scenario we set up a user study with the goal to measure the interrelation between acceptance (i.e. usability) and efficiency (i.e. ability to speed up knowledge work) of a context detection environment. Therefore we sketch the connection between context and user goal in Section 2. define a taxonomy of user goals for knowledge work in Section 3 and introduce up-to-date approaches for unintrusive, probabilistic context detection in Section 4. We instantiate these approaches in a corporate user study that involves 12 typical tasks of knowledge work (see Section 5). The resulting improvement of the user's knowledge work is measured qualitatively and quantitatively in Section 5, before we draw conclusions more on a systems modeling level.

2 Connection between User Goal and Context

Context adaptive systems generate user support with respect to a user goal identified by the system. This demands an understanding of the connection between goal and context. To describe this connection, we use an extended k-system control-circuit model as presented by [1] (see Figure 1), originally used to describe system-world interaction. In our adaptation it shows the dominance of the user goal on user context mediated by action in and perception of the real world. A user might have multiple goals concurrently which have different relevancies. We consider the goal with the highest relevance as trigger for the organization of user-world interaction in a situation. Organization means that the goal leads to a planning process of the user, how to achieve a goal. The resulting plan as behavior in rehearsal guides user perception of and user action in the real world, as described in adaptive resonance theory [2]. Thus, user context is dependent on a goal and the resulting plan. A respective context term has been introduced by [3][4] and slightly modified by [5]. They distinguish the following categories of the real world for an individual:

- Intrinsic Context: Those elements of the physical world which are consciously perceived and considered by an individual as related to a goal

[4] referred to in [4]

– Extrinsic Context: Those elements of the physical world which are consciously perceived and not considered as related to a goal
– Unperceived things: Aspects of the real world which are not consciously perceived by an individual

Context is changed by actions and consumed by perception. That can result in adaptation of the plan and can again have an effect on the goal, which closes the modeled control-circuit. A context adaptive system interacts with the control circuit. It is an element of the real world which detects user actions based on sensors and deduces user goals based on collected sensor data. In a desktop environment this means that the actions of the user, like opening applications, using application-specific functionalities, etc. are indicators which are used to identify a user goal. Once, the system identifies a user goal three different kinds of support can be given:

– support the perception of the intrinsic context (e.g. highlight specific elements of the context to support user orientation)
– support actions on elements of the intrinsic context (e.g. automation of time consuming and stereotype activities)
– extend the intrinsic context by adding elements to the conscious and goal related user perception (e.g. recommendation lists)

To identify goals and decide on reasonable support a thorough understanding of user goals and their effects on the interaction with the real world is necessary. In the following we describe the understanding of user goals we followed to realize a context adaptive system.

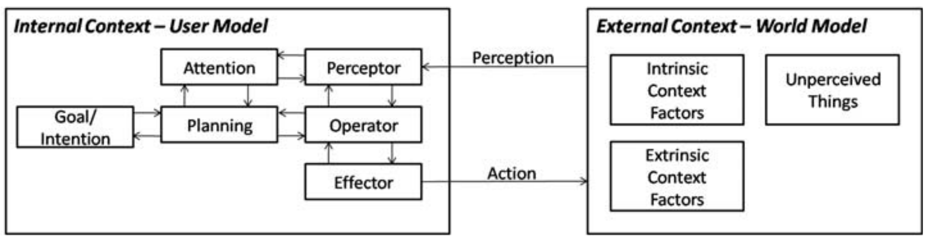

Fig. 1. Connection between User Goal and Context

3 Modeling User Goals

In accordance to Broder [6], who defined a taxonomy of user goals for web retrieval, we sub-divide the concept of user goal into three different subtypes including:

1. Informational Goal

2. Transactional Goal
3. Navigational Goal

While all subcategories are intentions the user might have during his/her work, the characteristics and the methodology to detect these goals differ.

In certain work situations the knowledge worker needs factual knowledge in order to fulfill his/her working task. An *informational goal* addresses background knowledge that is necessary to understand concepts with regard to the working task. If for example the knowledge worker needs to model a software with UML, the understanding of the concept UML class diagram is crucial to complete the task. The search for proper knowledge resources, that explain a class diagram, embodies an informational goal.

The user goes after a *transactional goal* if s/he is trying to accomplish a working task in a work process by performing certain transactions. Those transactions might involve user interaction with the system that result in a defined work result (e.g. a text document or a spreadsheet). The detection of a transactional goal consist of the analysis of the user interaction with the system in order to reason about the anticipated work task. A work task might be writing a letter, creating a balance sheet or compiling a presentation.

A *navigational goal* however is not characterized with regard to content aspects, but does target on a navigation path to a state of location in the work environment. The user's intent here is to find a particular document, directory or file, s/he already used. This involves also web resources like URLs the users has visited sometimes in the past. The resource might be of interest for him/her as a template or as an example. The navigational goal represents a description of the location of the object of interest via a path or a URL.

Since the three types of user goals cover different aspects of knowledge work, the algorithms to anticipate them vary too. The next section describes probabilistic approaches to reason for the supposed goal of the user by analyzing context features of the work environment (for a detailed enumeration of our context features see [7]). All approaches target to manage to get along with low explicit user input (like it would be the case with online learning or feedback mechanisms), in order not to spoil the useability. We strongly believe the user is not willing to accept additional effort to support a system that is meant to support him/her without an overall benefit.

4 Approaches to User Goal Detection

4.1 Informational Goal

In order to estimate the user's need for information in a certain work situation, the system needs to have means of interpretation of the topic the user is currently dealing with. The scientific field of topic detection offers a number of options given a textual corpus that can be analyzed. In distinction to topic detection in computational linguistics, where the input consist solely of one text document or fragment, we define the combination of all textual context features of the work environment as input for the topic detection. This involves not only documents currently opened in different applications, but also the content of websites displayed in the browser, window titles and file names. Recent methods for topic detection that deliver good results include e.g. LSA[5] [8]. But since we extract topical information only for the purpose to identify relevant words or concepts the user might have a question or informational need for, we focus on simple keyword-based approaches here. The extracted keywords relevant to the user's current working task represent potential informational goals the user needs information on. Based on the list of keywords a knowledge management system or work-place embedded help system might offer resources that explain the concept or define the keyword. In a generic scenario the learning resources can be derived simply by offering the corresponding Wikipedia page to the keyword, in specific corporate scenarios a corporate knowledge repository, reflecting professional needs is recommended. The relevance of a keywords with regard to the working task can be estimated by term relevancy measures as in [9] or given by a static list of relevant terms defined by a domain expert. In a similar fashion to the APOSDLE approach [10] we applied string matching on a list of keywords characteristic for a particular task. But in contrast to the APOSDLE approach the tagged learning material was automatically drawn from Wikipedia, i.e. referring to pages with the respective tags.

4.2 Transactional Goal

Task detection as a research category, which is used for the identification of a transactional goal, has already been described in [11], [12] and [7]. We also proceed with a machine learning approach that uses user interaction with the system and the work environment itself as an indication for a particular work task, since after a short training phase which can also be outsourced, the system works autonomously. Therefore we operate on slices of the event stream captured by desktop sensors. Our context model here is a holistic one with regard to the number of features that can be captured on the computer desktop itself, not in the physical environment around it. We apply a hybrid voting approach between the decision tree ID3 [13], Naïve Bayes [14], Euclidean distance [15], Irep [16] and SMO(128) [17] algorithms in this experiment. This outperformed the single algorithms named on our data set.

4.3 Navigational Goal

The identification of the navigational goal is the task of identifying the next document to be opened by the user. If we anticipate the document right and provide a short link the user saves time for navigation and search. We formulate this problem scientifically as sequential prediction and leverage from recent research in clickstream analysis (see [9]). Without any background models we create a navigational graph in-time that consists of documents accessed as nodes

[5] Latent Semantic Analysis

and transitions between two documents as vertices in the graph. A sample graph from our user study is shown in Figure 2. The detection of the matching document is then based on a sequence of documents in the navigation path, which was recently used by the user. On this graph we can use partitioning algorithms and propose all navigation objects in the actual partition that have not been accessed in the actual session. Phase 1 of our user study showed the most effective results for navigational goal detection, which are based on the paradigm of spreading activation [18] on such a navigational graph.

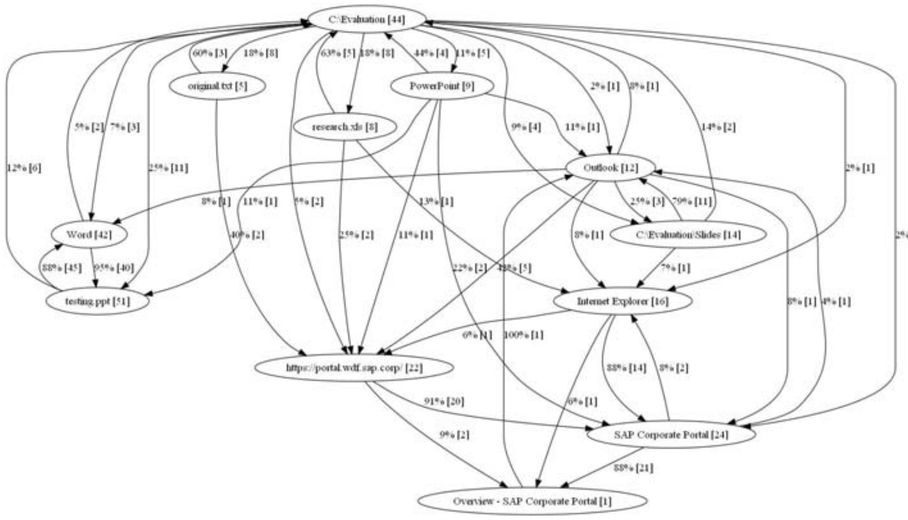

Fig. 2. Navigational graph enabling algorithms without task or domain model

5 Design of the User Study

The following paragraph describes the general setup of our in-house context detection study in the corporate environment of a large software company with dominant characteristics of knowledge work in daily business. Our general research question addresses not only the accuracy of context detection itself but also the interrelation between acceptance (i.e. usability) and efficiency (i.e. ability to speed up knowledge work) of the described context detection environment. The experiment consisted of two phases:

In phase 1 we collected training and evaluation data. We used these data to improve our system and find the best machine learning algorithms and parameters for the recommendation algorithms. In phase 2 we tested the improved system.

Our report will sketch these phases, discuss the measurements and draw conclusions for the introduction of context detection to an organization - in general, along task models and along domain ontologies characterizing the organization.

Phase 1

We defined 12 tasks, which have been performed by 20 participants from the SAP Research CEC Darmstadt (mainly postdoc-level researchers and PhD candidates) during the first phase without any contextualized support.

The tasks are condensed versions of typical work a researcher has to carry out in the context of the industrial research projects, transfer projects and program activities at SAP Research in Darmstadt (comments in italics):

1. create a presentation on Generics in Java *(i.e. preparing technical slides)*
2. leave request *(interacting with a typical SAP tool)*
3. update the SRN *(the SAP Research Knowledge Representation Tool)* page of your project
4. distribute presentation slides *(find the right people and their full names to send slides to)*
5. visualization of quantitative research results *(MS Excel-style)*
6. translation of executive summary *(a typical task as SAP is a bilingual company)*
7. code development: Hello World class in Java *(very simplified programming task)*
8. create a handout *(for a presentation)*
9. create a UML-diagram *(very simplified)*
10. budget calculation *(no tool pre-nominated)*
11. software update *(non-automatic software update of one tool)*
12. inventory update *(modeled as an interruption of another task)*

To avoid correlations between the tasks the participants got the tasks in a random order. During the data collection phase these tasks were conducted in a restricted time and with some hints on supporting material (e.g. the presentation slides). We collected the data in a database and labeled it according to the task in which it was collected.

Phase 2

In this phase we let 15 participants from the SAP Research CEC Darmstadt (a true subset of the phase 1 participants) perform the tasks a second time in random order. This time they were supported by our context detection system (see Figure 3). This happened weeks after the first phase to blur the participants memory on how the tasks are performed - a quite realistic condition with support from our system. In comparison to recent approaches (e.g. APOSDLE P3) we used a very simplified user interface (see the screenshot below) only showing documents stemming from task detection, topic detection and clickstream analysis to collect information about how our system helps them to fulfill the task faster and easier. By design, the users in phase 2 had as easy access to the tools for the tasks (left side, for instance MS Excel) as in phase 1 to principally enable work without context detected support.

Fig. 3. Screenshot of the simplified UI - the right hand sidebar showing three potential recommended list of documents from topic detection (with a topic as informational goal), purely unsupervised mechanisms based on the users clickstream (a document or URL as a navigational goal) and task detection (with a task as transactional goal).

5.1 Evaluation and Discussion

We used a connection of the users actions to a system clock to measure the amount of time needed per task in phase 1 and phase 2. Figure 4 shows a detailed time analysis in the form of and the relative amount of time gained (green) or lost (red) with/without context adaptive support.

The average user became 30% faster on an average task. Intuitively, this number is more than the speed-up expected without tool support by a pure learning effect with scrambled, once interrupted short tasks resembling a heavy workload. But the proof via a control group that tool support is the reason for speed-up is still missing.

The two most effectively tasks speeded up were routine tasks (Update SRN and Software Update) with relatively low involvement of personal creativity. However, despite the fact that these were routine tasks, their repetitivety is low enough to still offer enormous room for automated support as in our study, e.g. by just presenting the right entry point to the system to be updated. Tasks like Visualization Results and the ones on Prototype Development or Translation

	Generics	Update SRN	Distribute Slides	Visualization Results	Leave Request	Translation	Prototype Development	Inventory Update	Handout	UML	Budget Calculation	Software Update	Complete
User 1	-21,05%	-70,31%	36,42%	59,05%	-61,63%	33,33%	-36,76%	-0,85%	-31,76%	-39,92%	-32,75%	-69,38%	-29,96%
User 2	13,59%	-60,63%	-8,89%	158,20%	24,38%	32,52%	28,13%	-4,19%	105,93%	22,20%	33,93%	-39,89%	14,97%
User 3	-55,59%	-62,33%	17,65%	-20,14%	32,45%	-1,74%	-25,12%	-24,14%	-58,91%	5,31%	-10,97%	-7,33%	-18,91%
User 4	-50,00%	-51,56%	-53,40%	41,22%	-28,30%	-31,00%	-18,57%	-51,75%	-26,24%	-49,53%	-35,41%	-51,56%	-33,69%
User 5	-53,55%	-55,81%	-35,03%	-35,27%	-39,01%	-54,22%	-58,88%	-56,33%	-31,30%	-50,72%	66,67%	-55,17%	-42,78%
User 6	-51,19%	-74,68%	-11,97%	-59,86%	-29,05%	-23,87%	-24,51%	-22,97%	-17,65%	-33,05%	-2,57%	-26,83%	-32,13%
User 7	-50,48%	-93,52%	-63,70%	-10,33%	45,71%	-25,65%	4,22%	-57,58%	-53,72%	-36,73%	23,60%	11,22%	-31,92%
User 8	5,83%	-84,77%	-29,53%	-78,69%	-47,00%	-10,39%	-55,81%	2,16%	-10,42%	-29,54%	-35,14%	-49,55%	-41,62%
User 9	-66,88%	-33,11%	-6,02%	-9,50%	4,33%	-0,46%	39,08%	-6,15%	-79,15%	-31,79%	-4,03%	-53,47%	-25,14%
User 10	-30,60%	-55,26%	-62,41%	-34,72%	-10,90%	-47,62%	12,39%	-46,19%	3,83%	-59,27%	-35,78%	-54,82%	-32,17%
User 11	-5,29%	-81,25%	-16,99%	-40,67%	-34,84%	-42,79%	33,97%	-5,11%	-38,39%	32,76%	-42,66%	-45,00%	-29,77%
User 12	-55,85%	-87,85%	-51,92%	-49,76%	-57,36%	-29,05%	-65,16%	-59,71%	-75,80%	-54,53%	-59,42%	-76,53%	-58,82%
User 13	-67,00%	-82,46%	-71,63%	-3,86%	43,42%	-9,74%	13,22%	-41,52%	-14,36%	-38,44%	-62,46%	2,58%	-31,98%
User 14	-26,97%	-47,92%	-55,05%	-3,61%	24,22%	29,72%	-36,50%	-2,82%	-10,26%	-30,43%	-19,18%	-62,17%	-20,00%
User 15	-37,77%	-45,13%	-7,14%	-29,46%	-37,33%	-62,14%	-54,94%	-60,16%	-30,70%	-6,83%	-22,44%	-60,66%	-36,32%
Average	-37,51%	-65,79%	-27,97%	-7,83%	-11,39%	-16,21%	-16,35%	-29,15%	-24,60%	-26,70%	-15,91%	-42,50%	-30,02%

Fig. 4. Speed measurements, relative speed-up (green) or slow-down (red) of users

involve more creativity, but still show the supportability by context detection in our lab setting. There was one user (User 2), who was very much distracted by the recommendation system - at least in a positive sense, as s/he liked the system very much and started playing with it. This attitude seems to be an outlier in comparison to the other fourteen users. Another outlier, the leave request with relatively many users taking more time to perform with context detection support, lacks explanation so far. None of the 15 users found particular bad suggestions by the system regarding this task - and even 43% of the users found particularly good recommendations in that task.

Figure 5 shows the overall satisfaction with the combination of tools as determined by a questionnaire.

A surprising result shown at the top of Figure 5 is the overall opinion about the intrusiveness of the system. 57% of the users did not feel intrusiveness, a reason to follow to the presentation of documents and not tasks and topics, such as in APOSDLE P3 [19] and similar systems as the Microsoft Office Assistant. As the design of the UI came nearly by chance and just piggy-bagged of the scientific approach to make the three different flavors of context detection (navigational, transactional, informational) comparable for the lab study, we see a very interesting anchor for further UI design around the smoothly morphing list of document hits updated by the context detection regularly. Further research in that direction drawing a distinction from the current metaphors “pop-up of suggestions” or “click to get a context-based suggestion” seems to be very valuable. Although the system was not felt to be very intrusive (or at least by a minority of the users perceived as such), a strong majority (93%) of the users were aware of the fact, that different kinds of context detection were designed to support them. The recommendation of documents from the spreading activation algorithm turned out to be most valuable from the users perspective. We also

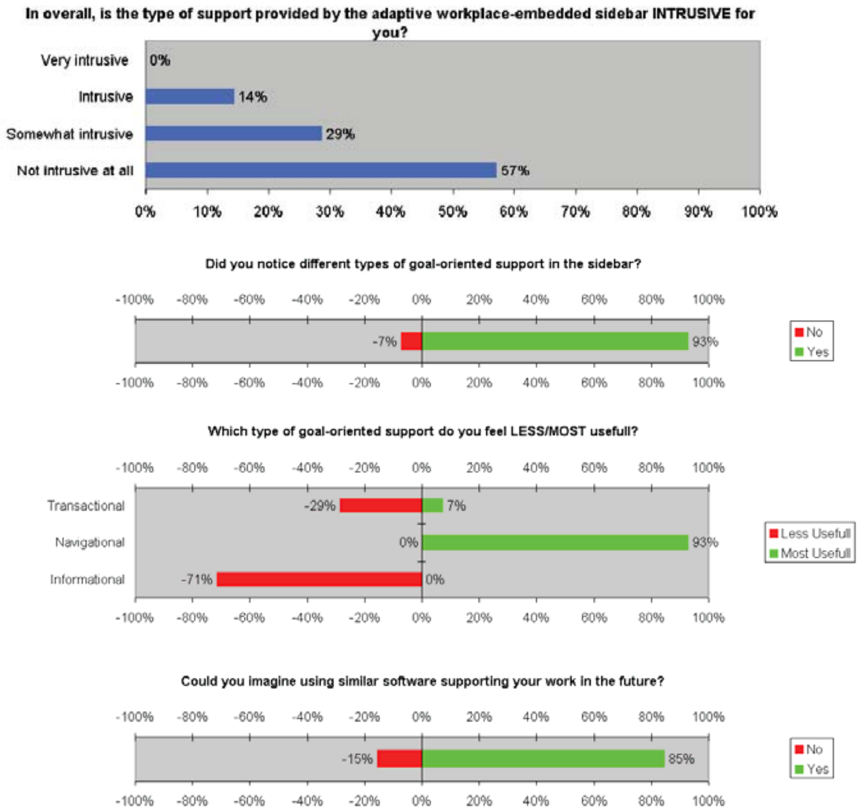

Fig. 5. Results of the questionnaires

related this question to the click behavior, which showed that these documents from the navigational mechanism were also selected most often - in contrast to a mediate selection (and a medium judgement in the questionnaires) of documents from the task detection and almost no response to the topic detection. This is a clear distinction from the results in the summative evaluation of APOSDLE, where the topic detection was favored above the task detection. The outcome depends on the quality of the models and annotation. In this experiment, the linking to Wikipedia was perceived as too general even for "non-office" tasks, where a source of encyclopedic knowledge might be expected as useful. The other way around, navigational mechanisms do not depend on models, supported the users in our study - but have the conceptual drawback of only referring to documents, which were already at least touched by the users (i.e. triggering the question, how new knowledge can be transported to the individual user, a question of true knowledge transfer in the organization).

Finally, the system was perceived as helpful, which we conclude from the combination of the two facts that it caused speed-up and that most of the users claimed, that they would use it again. The questions which remain open are less on the algorithmic side, but more on the organizational side and UI side:

- How to effectively and efficiently connect models to the context detection and how to complement this with the more personal navigational mechanisms?
- Which UI-metaphor is best suited for presenting the continuous flow of context-dependent suggestions?
- Does the possible extension of defining and adopting holistic enterprise models compensate the heavy modeling effort?

6 Summary and Outlook

We identified the best context detection algorithms in a specific empirically accessible workplace environment. Our study included task detection, topic detection and model-free, purely statistically determined navigational context detection. Contextualized support seems to be helpful along with the right metaphor for the UI but - when going beyond the helpful reconstruction of personal document streams and piles - strongly dependent on the investment in the modeling of tasks and topics (i.e. in transactional and informational goals). Thus the main follow-up of our study is, that we are now in the position to suggest a way of context detection intertwined with modeling, which slightly deviates from the strategy taken in e.g. APOSDLE integrated modeling [20]. The key is human activity monitoring: We suggest to apply context detection already in the very early phases of task and topic modeling by navigational mechanisms individually and to complement this by task models and ontologies, which in some sense (e.g. by pooling search terms passed to search engines) are seeded decentrally in the organization. The purpose is to focus modeling on situations, where we can automatically determine, that navigational context detection passes back values (e.g. documents), which are from the personal history, but do not reflect the user's goal. Such a gap analysis should determine the shape of the domain and task model. This differs from a modeling strategy of interviewing experts and opens the door to a more decentralized and continuous way of relating detection mechanisms and modeling. However, the knowledge engineer and the domain expert are still be needed to work on the seed we mentioned as a result of human activity monitoring and the later phases of validation. In our upcoming work, we will focus on the category of transactional goals. We want to use user action data of multiple users to identify reoccuring subcategories of the transactional goal. For these subcategories we want to provide identification mechanisms and realize specific, proactive user support. This will help to get a better understanding of the difference and similarity of transactional goals of multiple users.

References

1. Stachowiak, H.: Allgemeine Modelltheorie. Springer-Verlag (1973)

2. Grossberg, S.: The link between brain learning, attention, and consciousness. Causality, Meaningful Complexity and Embodied Cognition (1999) 3–45
3. Hewitt, K.: Context effects in memory: A review. Unpublished manuscript. Cambridge University, Psychological Laboratory. (1977)
4. Tiberghien, G.: Il CONTEXT AND COGNITION: INTRODUCTION. Revue bimestrielle publiée avec le concours des Universités de Provenœ et Aix-Marseille II **6**(2) (1986)
5. Öztürk, P., Aamodt, A.: A context model for knowledge-intensive case-based reasoning. International Journal of Human Computer Studies **48** (1998) 331–356
6. Broder, A.: A taxonomy of web search. SIGIR Forum **36**(2) (2002) 3–10
7. Lokaiczyk, R., Faatz, A., Beckhaus, A., Görtz, M.: Enhancing just-in-time e-learning through machine learning on desktop context sensors. In Kokinov, B.N., Richardson, D.C., Roth-Berghofer, T., Vieu, L., eds.: CONTEXT. Volume 4635 of Lecture Notes in Computer Science., Springer (August 2007) 330–341
8. Landauer, T.K., Dumais, S.T.: A solution to Plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge. Psychological Review **104** (1997) 211–240
9. Lokaiczyk, R.: Verfahren kontextbasierter Nutzerzielanalyse. PhD thesis, Universität Leipzig (2009)
10. APOSDLE consortium: Deliverable d6.3 - use cases & application requirements 2 (second prototype). Technical report (2008)
11. Shen, J., Li, L., Dietterich, T.G.: Real-time detection of task switches of desktop users. In Veloso, M.M., ed.: IJCAI. (2007) 2868–2873
12. Granitzer, M., Kröll, M., Seifert, C., Rath, A.S., Weber, N., Dietzel, O., Lindstaedt, S.N.: Analysis of machine learning techniques for context extraction. In Pichappan, P., Abraham, A., eds.: ICDIM, IEEE (2008) 233–240
13. Quinlan, J.R.: Induction of Decision Trees. Machine Learning **1**(1) (1986) 81–106
14. Dumais, S.T., Platt, J.C., Hecherman, D., Sahami, M.: Inductive Learning Algorithms and Representations for Text Categorization. In Gardarin, G., French, J.C., Pissinou, N., Makki, K., Bouganim, L., eds.: Proceedings of the 1998 ACM CIKM International Conference on Information and Knowledge Management, Bethesda, Maryland, USA, November 3-7, 1998, ACM (1998) 148–155
15. Agrawal, R., Faloutsos, C., Swami, A.N.: Efficient similarity search in sequence databases. In: FODO '93: Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms, London, UK, Springer-Verlag (1993) 69–84
16. Fürnkranz, J., Widmer, G.: Incremental Reduced Error Pruning. In: Proceedings the Eleventh International Conference on Machine Learning, New Brunswick, NJ (1994) 70–77
17. Platt, J.: Fast Training of Support Vector Machines using Sequential Minimal Optimization. MIT Press (1998)
18. Anderson, J.R.: A spreading activation theory of memory. In Collins, A., Smith, E.E., eds.: Readings in Cognitive Science: A Perspective from Psychology and Artificial Intelligence. Kaufmann, San Mateo, CA (1988) 137–154
19. Lindstaedt, S.N., Ley, T., Mayer, H.: Integrating Working and Learning with APOSDLE. In: Proceedings of the 11th Business Meeting of Forum Neue Medien, 10-11 November 2005, Vienna, Verlag Forum Neue Medien (2005)
20. Rospocher, M., Ghidini, C., Serafini, L., Kump, B., Pammer, V., Lindstaedt, S., Faatz, A., Ley, T.: Collaborative enterprise integrated modelling. In: Proceedings of the 5th Workshop on Semantic Web Applications and Perspectives (SWAP2008), Rome, Italy, December. (2008) 15–17