D3.3 Real-time Activity Classification with user feedback

Project	SWELL
Project leader	Wessel Kraaij (TNO)
Work package	3
Deliverable number	3.3
Authors	Maya Sappelli (TNO/RUN), Suzan Verberne (RUN)
Reviewers	Wessel Kraaij (TNO), Joris Janssen (Sense OS)
Date	24-11-2014
Version	1.0
Access Rights	COMMIT/ public
Status	final

SWELL Partners:

Almende, Noldus, Novay, Philips, TNO, Radboud Universiteit Nijmegen, Roessingh Research and Development, Sense-Os, Universiteit Twente,

COMMIT/ A public- private research community

This publication was supported by the Dutch national program COMMIT.

Summary

For this software deliverable D3.3 "*Real-time activity classification with user feedback*", we adapted the software implementation of deliverable D3.2 "*Activity classification*" to make the context identification as well as the task identification available to the end-user of the software in real-time. Furthermore, both the user interface as well as the backend software are extended with capabilities to actively relearn from user annotations and feedback. This functionality is integrated in the Happy Worker App as part of M1: Time and Stress Monitor (D3.1b) and M4: Search Assistant. The Happy Worker App provides personal feedback on working behavior, based on computer logging and user input.

In this deliverable we describe the implementation of this functionality in the Happy Worker App software.

Contents

Summary 1
1. Introduction 3
1.1 Context of this deliverable 3
1.2 Demonstration of the software 3
2. Functional Description 6
2.1 Main functional requirements 6
2.2 Task Activity Classification 7
2.3 Context Classification 7
3. Technical Specification 8
3.1 Change Log for D3.3 8
3.2 Requirements for future use of the Happy Worker App 9
4. Conclusion 9
5. References 9

1. Introduction

In our project SWELL (smart reasoning for well-being at home and at work) we aim to improve well-being at work by supporting knowledge workers. We define knowledge workers as people who use and produce information as their main task, and frequently use computers. Well-working could be defined as “being and feeling in control”, with a positive impact on work efficiency and effectiveness, work pleasure, mental and physical health status (see deliverable D3.8).

In the project SWELL we aim to support knowledge workers in their daily life. Work package 3 (SWELL@Work) focuses on supporting knowledge worker during their work. The main goal is to allow the knowledge worker to ‘work in context’. For this purpose it is important to detect reliably what the user is doing such that coaching and other help can be made context-aware. In deliverable D3.2 we have worked on this by developing and implementing algorithms to detect the task and context a user is working on. In D3.7 we extended these algorithms to provide suggestions for documents to read. In this deliverable we focus on making the algorithms work in real-time, and improving the accuracy of the algorithms by providing the opportunity to the users to give feedback in real time on the detection quality. The feedback is then used to retrain the network.

In Section 2 we give a functional description of the software, followed by technical specifications and requirements in Section 3.

1.1 Context of this deliverable

The software developed for this deliverable is an improved version of the software delivered in D3.2 and D3.7. The main improvement is the addition of real-time task identification. Furthermore, the implementation has been extended to allow the user to interact and give annotations and feedback on the system. By asking the user for supervision, the task and context recognition becomes more accurate and personalized.

1.2 Demonstration of the software

Figures 1 to 4 show screenshots of the Happy Worker App. Screenshots of the settings screen and the activation graph can be found in deliverable D3.7. Figure 1 shows the main application window, in which the user is presented with the detected context/project, a tag cloud of highly activated nodes that describe the context (to some extent), the detected task and a list of suggestions. This interface is updated in real-time when the user progresses in his/her working activities. Figures 2 to 5 show the pop-up boxes that are presented to the user when he/she is asked for feedback. Finally, Figure 6 shows the Gantt chart of the detected tasks. The screenshot shows that a user could add a new task ‘distraction’ to detect how often he/she is distracted from work.

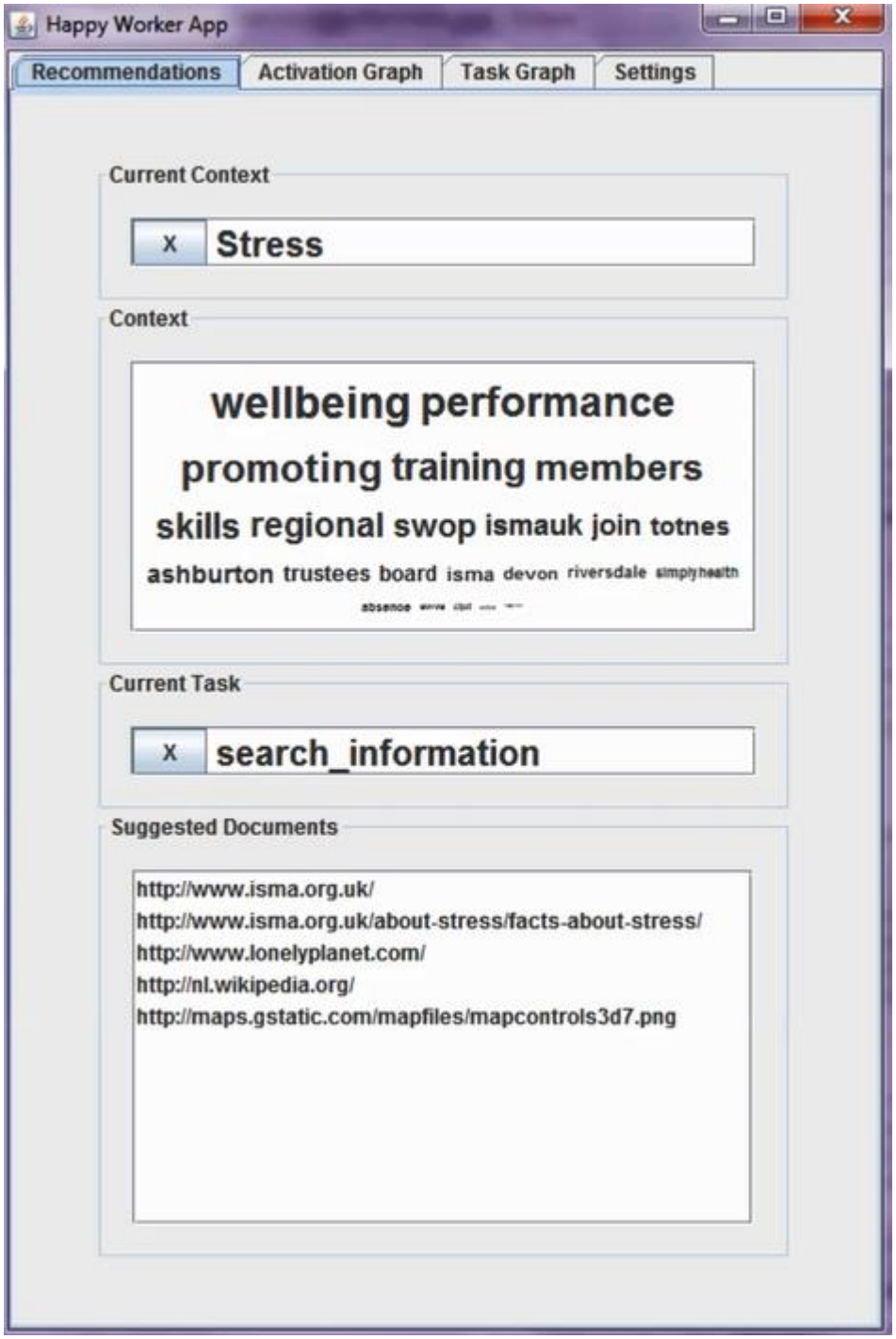

Figure 1: Main application window showing detected context, a tag cloud of active contextual elements, the detected task and a list of suggested resources.

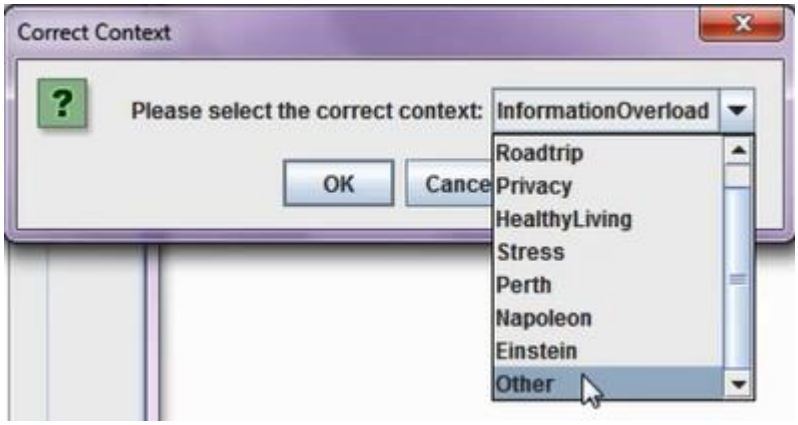

Figure 2: Pop-up dialog window to correct the detected context. A new context can be added by selecting the option 'Other'.

Figure 3: Pop-up dialog window to enter a new context.

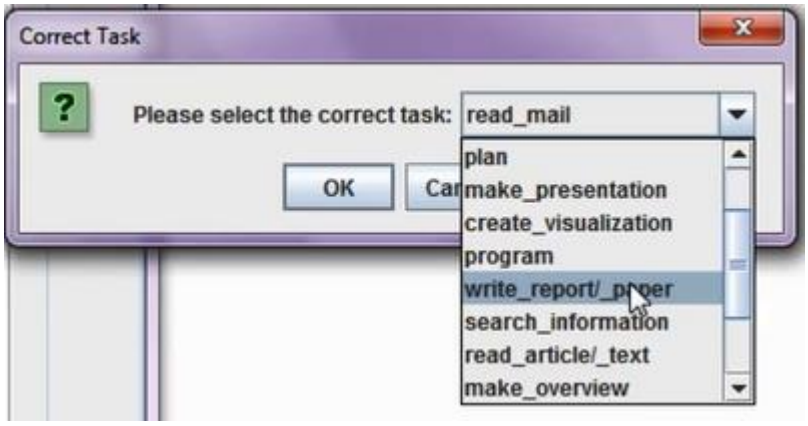

Figure 4: Pop-up dialog window to correct the detected task. A new task can be added by selecting the option 'Other' from the drop-down menu list.

Figure 5: Pop-up dialog window to correct the detected task when the confidence level of the task classifier is below 50%.

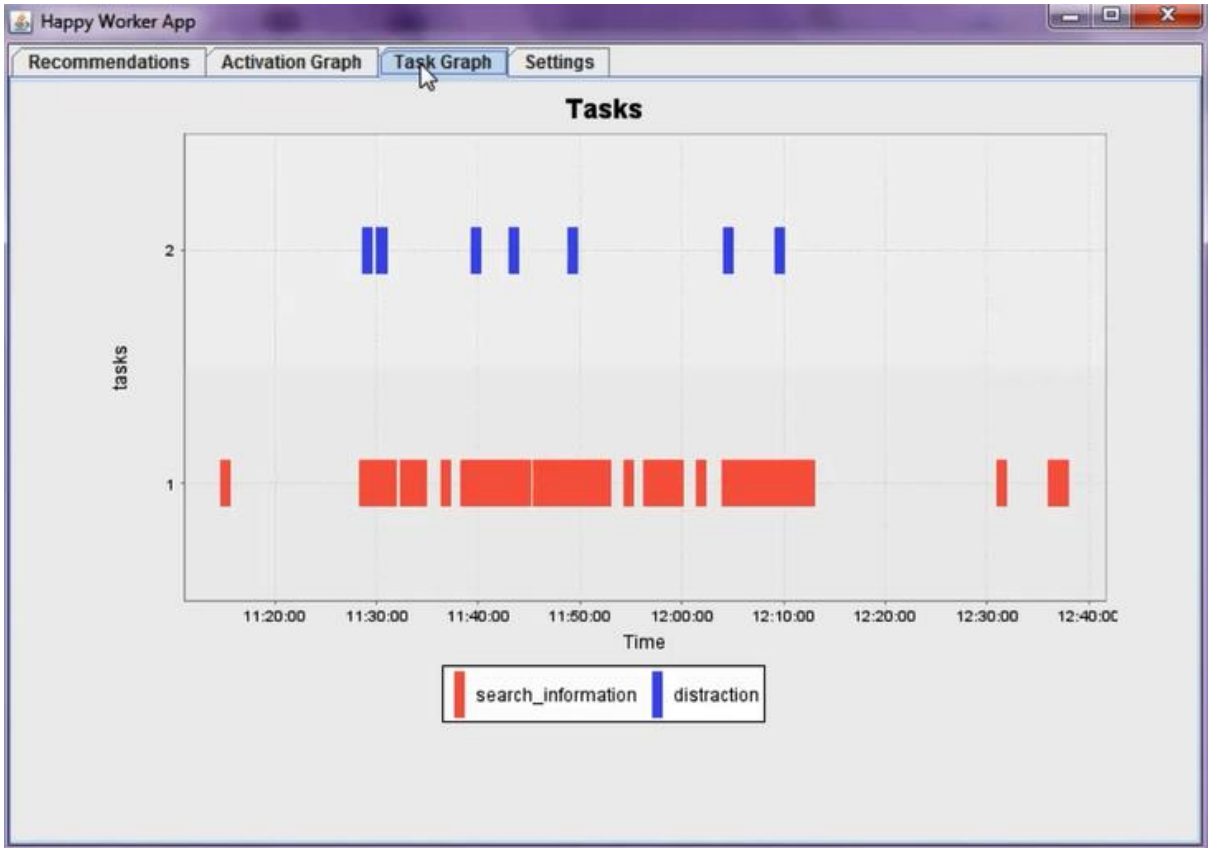

Figure 6: Graph overview of the detected tasks.

2. Functional Description

2.1 Main functional requirements

The main functional requirements for the Personal feedback tool *Happy Worker App* in the SWELL@work system are:

- F1: The software unobtrusively collects objective behavior data (computer logging).

- F2: For feedback, the data is visualized in an intuitive, easily interpretable and appealing way. Our idea is that feedback that is easily interpretable and engaging is most effective for gaining insights.

An additional function requirement for real-time recommendation of documents and webpages is:

- F3: The software provides real time access to documents and webpages relevant to the current activities. Our idea is that it is better to let the user decide to access the recommendations rather than to push recommendations.

These functionalities have been implemented in a Java version of the *Happy Worker App* that focuses on task classification, context classification and context-aware recommendations.

2.2 Task Activity Classification

- The basis for the implementation of task activity recognition is the implementation from D3.2. The activity classification is implemented as a supervised learning task. We use the Updatable Naïve Bayes model from the WEKA package for this purpose.
- The initial training set for the algorithm comes from Koldijk(2011). The initial set of labels that are used are: *read mail, write mail, organize/archive data, plan, make presentation, create visualization, program, write report/paper, search information, read article/text, make overview, analyze data, away from keyboard, none*.
- The implementation is adapted such that feature extraction, task identification and the updating of the visualization of task identification over time takes place in real-time.
- The implementation is adapted such that dynamic event blocks can be used, rather than fixed windows of 5 minutes.
- The user is allowed to create a new task and provide training examples for the new task.
- The user is asked to confirm or correct a detected task when the classifier's confidence is lower that 50%.

2.3 Context Classification

- The basis for the implementation of context classification is the implementation from D3.2 and D3.7. The context classification is implemented as a feed forward network based approach with the focus on entity, topic, time and location information. The network is depicted in Figure 7.
- The user is allowed to create a new context. The algorithm makes automatic connections between the newly created context and the contextual elements in the event that is observed.
- The user can correct a context detection. At this point the weights in the network are updated using back propagation to enhance the activity of the correct label and decrease the activity of the incorrectly recognized label.

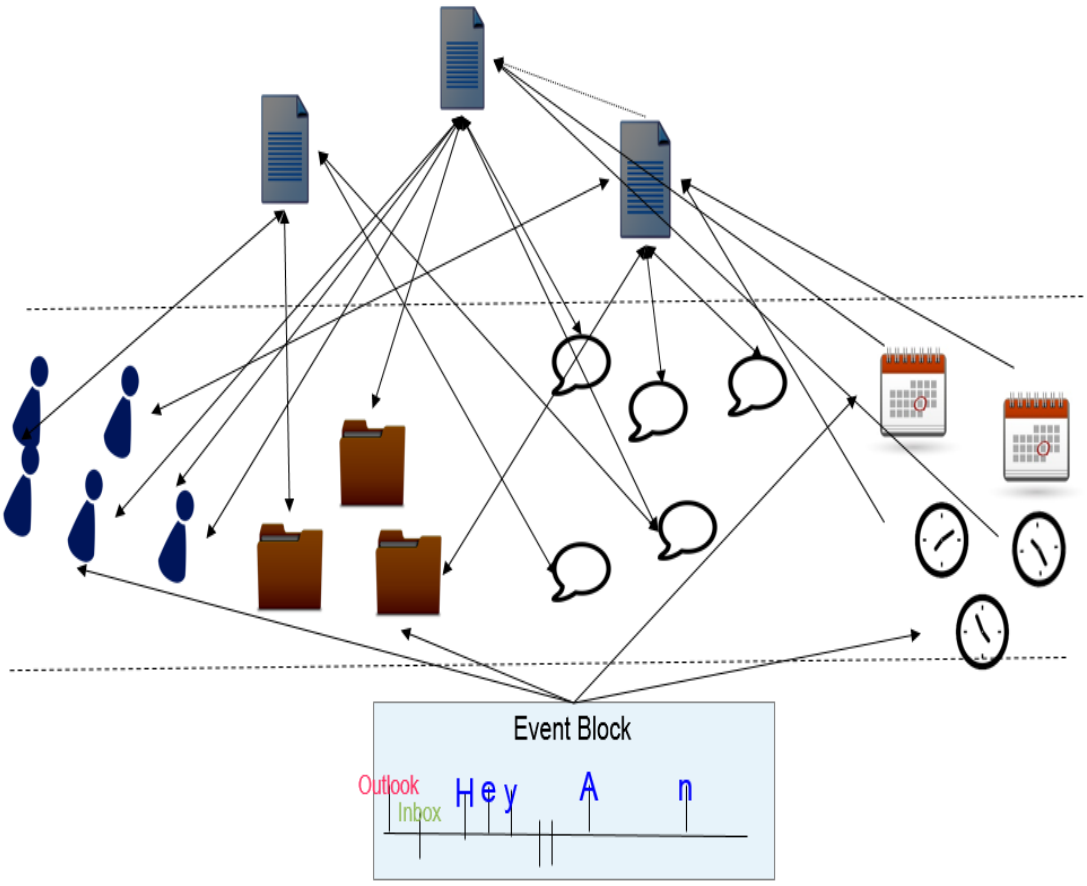

Figure 7: Schematic overview of network underlying the context detection software.

3. Technical Specification

The purpose of this section is to document the (required) changes in the software, mainly as a reference for future developers

3.1 Change Log for D3.3

- General improvements to the network method by Maya Sappelli are made (bug fixes).
- Task Detection is made real-time and based on dynamic event blocks (previously 5 minute windows).
- Interface to ask the user for supervision on the classification tasks.
- Functionality to train the connection weights in the network based on user feedback.
- Functionality to update the trained Naïve Bayes classifier for task detection.
- Interface update to show the task detection over time.
- Interface update to show additional information on the detected context in the form of a tag cloud.

3.2 Requirements for future use of the Happy Worker App

- Implement functionality to optimize the condition of the network (e.g. remove superfluous nodes and connections)
- Unit Tests
- Upload detected contexts and tasks to Common Sense by Sense Observation Systems

4. Conclusion

We improved the previous Java software implementation of classification of tasks and contexts (projects/topics) of D3.2 to make the classification real time. Additionally, we implemented functionality to ask the user for feedback on the classification quality. The user interface is updated to a small window with 4 tabs. One tab shows the recognized active project and task, together with a tag cloud of the active context and a list of 5 document suggestions. The second tab and third tab show the activation charts of tasks and projects to provide the user with an overview of his day, and the last tab allows the user to adapt the projects he wants to include in the classification as well as some other settings. We demonstrate the software with a simulation of the data that was collected in D3.5.

5. References

- McClelland, J. L., & Rumelhart, D. E. (1981). An interactive activation model of context effects in letter perception: I. An account of basic findings. *Psychological review*, *88*(5), 375. Koldijk, S. (2011). Look what you've done! Task recognition based on PC activities. Masters' thesis. Radboud University, Nijmegen, The Netherlands.
- Koldijk, S., Sappelli, M., Neerincx, M., & Kraaij, W. (2013). Unobtrusive monitoring of knowledge workers for stress self-regulation. In: Proceedings of UMAP 2013 (Rome, Italy, 10-14 July 2013).
- SWELL D3.1b Goal recognition and activity classification – updated algorithm design https://ecity.tno.nl/sites/COMMIT_SWELL/Documenten/20130624%20SWELL%20D3.1b_functional%20design.pdf
- SWELL D3.2 Activity Classification https://ecity.tno.nl/sites/COMMIT_SWELL/Documenten/20131218%20SWELL%20D3.2_Activity%20classification_FINAL.pdf
- SWELL D3.5 Dataset of activity and interaction streams https://ecity.tno.nl/sites/COMMIT_SWELL/Documenten/SWELL%20D3%205_final.pdf
- SWELL D3.7 Context and Task Sensitive information filtering and sharing agent https://ecity.tno.nl/sites/COMMIT_SWELL/Documenten/D3.7-final.pdf