On Resource Acquisition in Adaptive Workplace-Embedded E-Learning Environments

Robert Lokaiczyk[1], Eicke Godehardt[12], Andreas Faatz[1] and Marek Meyer[13]
[1] SAP Research, Darmstadt, Germany
[2] Fraunhofer-IGD, Darmstadt, Germany
[3] TU Darmstadt, Darmstadt, Germany

Abstract—**This paper presents an approach to address the idea of self-directed context-aware e-learning environments that tightly integrate into the electronic workplace of a knowledge worker. While developing the paradigm of informal, in-time e-learning on the workplace, the question appears where and how to acquire suitable learning material for the current learning need of the knowledge worker (KW). In contrast to traditional e-learning systems where didactically engineered courses address the user's learning interest, we provide learning snippets that directly relate to potential short-term informational learning goals. We show ways of acquiring concepts from the work context the user potentially wants to learn about. For these concepts we determine different types of learning material from different sources that answer the current learning need about the concept.**

Index Terms—**Context Awareness, Electronic Workplace, In-time E-Learning.**

I. INTRODUCTION

The characteristics of the fast-paced knowledge work performed by today's business users strongly demand an integrated, in-time e-learning solution that directly supports the user's current learning goal. KWs often do not have the time for class-room training or an e-learning course just to fulfill a particular short-term information need. To our understanding, the KW quickly changes his roles during the work process. As soon as the KW faces a problem in the work process the worker becomes a learner, seeking for a particular information snippet on a concept related to a process step in the work process. Whereas the objective of most e-learning systems is to educate the learner afore his actual work process, we believe that the traditional 'learn first, apply later'-paradigm is not applicable to today's dynamic knowledge economy. In contrast to inflexible learning systems, which require time-consuming preparation of learning material and courses by experts in advance, we believe a learning system should be integrated directly into the knowledge workers desktop and make use of resources that have not been created explicitly as learning material. In order to better fit the KW's dynamic needs, we propose the recommendation of learning snippets that satisfy the immediate learning need of the user.

Therefore, we present a context-aware learning system that is able to react in-time and in awareness of the current (informational) learning goal of the user. Thus, the learner can rely on up-to-date, informal knowledge connected to learning concepts that relate to the current work process.

Nevertheless, Informal Learning is not thought to be a replacement for class-room training or hand-crafted e-learning courses. Certain learning situations of the user, especially those with less pre-knowledge and very unspecific and abstract learning goals, still require extensive guidance and well-prepared learning courses.

Our system is thought as an extension to traditional e-learning which addresses certain short-term learning situations of a knowledge worker. If the learning goals are directed and closed [1], then the acquisition of potentially helpful learning resources can be completely automated. We incorporate the context of the learner's computer desktop, extract relevant terms and aggregate them to learning concepts. These concepts are then disambiguated in a semantic sense and represent a bag of potential learning topics the user is interested in. Then, we use a broad range of public and corporate learning object repositories (LOR) to retrieve learning material that might solve the current learning need. Lists of learning objects that are grouped according to their didactical type are then displayed to the user, who is able to browse and select the ones he is actually interested in. The whole architecture is shown on a high-level process diagram in Figure 1.

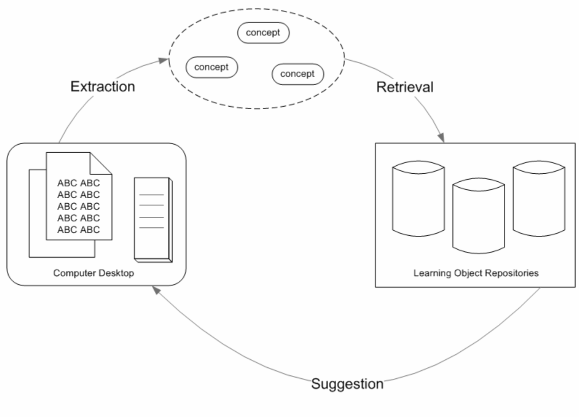

Figure 1. Overview: System architecture

The remainder of the paper is structured as follows: In Chapter II we describe a process that extracts learning concepts from the full-text representation of the user's electronic workplace. The extracted terms are then used as input in a learning resource retrieval algorithm in Chapter III. Next, the prototype to the approach is introduced in Chapter IV together with first experience reports. We end with conclusions and ideas for further extensions.

II. LEARNING CONCEPT ELICITATION

In order to understand what the user's actual learning domain is and what concepts s/he wants to learn about, we need to elicit a description of the learning situation from the computer desktop. We determine concepts of the current work by observing the user's desktop. Technically we extract all terms in opened documents of the computer desktop as well as terms from window titles and path names of files accessed. We determine a frequency-ranked list, filter typical stopwords and determine the most relevant terms by a difference analysis with regard to a reference corpus. These terms become concepts that describe the worker's current task and working domain when a word sense disambiguation is applied. This process is explained in detail within the next subsections.

A. Extraction

For determining the user’s learning goal, we observe the user’s desktop context. Therefore we use a general-purpose monitoring daemon, which is already introduced in [2] and [3]. This software is able to unobtrusively observe the behavior of the user, his interactions with the system and the resulting system state. In particular it is possible to extract the textual description of application window titles as well as document and window content of a significant number of applications.

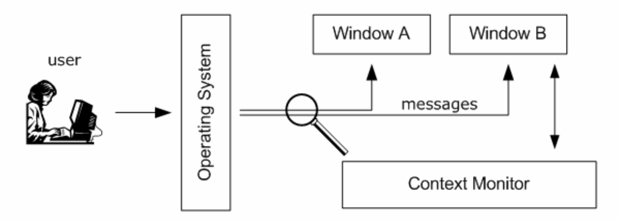

Figure 2. Desktop Context Monitor

The context information is gathered by software hooks and Windows API functions, which operate on system level. Figure 2 illustrates this process. The Windows operation system translates many system events to messages. For example, an application is notified of a keystroke by receiving a system message. These messages can be intercepted by the context monitor by setting a hook in the message queue of the system via a Windows kernel function. This way, it forwards all messages to the target application after unobtrusively logging the message. For this paper and the corresponding prototype we focus on all textual features on the computer desktop. That includes window titles, names of files currently opened, document content as well as website content and URLs of pages currently visited.

The underlying assumption here is that by filtering and aggregating this context data, we can extract concepts from the textual representation of the desktop which are on the one hand related to the current work process and on the other hand potentially subjects of the user’s learning goal.

B. Filtering

The unprocessed textual representation of the computer desktop itself is not an appropriate input for an algorithm retrieving learning resources. Such a representation still contains stop-words and unrelated terms not connected to the current work situation. We apply typical methods from the natural language processing area to filter the textual description and determine the most relevant terms describing the current working situation. In a first step, we take the textual representation and build a frequency-ordered word list. Terms with very high frequency often do not add significant meaning to the text like e.g. “the”, “and”, “a”, or “of”. They are called stop-words. We filter the top 100 stop-words[1] for each language our knowledge workers are handling with. The remaining list is then ranked according to the tdidf[2] measure. This measure ranks items with high frequency in the considered text fragment and low frequency across different documents of the reference corpus as semantically highly-relevant to the overall topic of the text. We select the 10 top-most terms by relevancy for further processing.

C. Disambiguation

Still, the extracted terms from the desktop context are highly ambiguous. This bears the problem that many terms map to different semantic concepts the user possibly wants to learn about. If the user for example wants to learn about the term *SAP*, it is crucial to disambiguate between the meanings:

1. The world’s largest enterprise for developing e-business software or
2. A historical socialist workers party in Germany.

But although the answer seems obvious to a human, who interprets the context of the term, developing algorithms to replicate this ability is actually a difficult task.

In the past several context-based methods for word sense disambiguation have been introduced in the related literature (see References [5], [6], and [7]). Most context definitions there are based on a formal characterization of the surrounding context of a word or a linguistic concept. Whereas context means either ‘neighbors of the examined term’, ‘the whole sentence’ or even ‘a larger paragraph’. Most algorithms use the similarity of contexts as decisive indicator for a particular for word sense. In Equation 1 we refer to context as the set of extracted terms after the filtering process.

In most related works, word sense disambiguation is considered as a classification problem, where a term t is classified as a concept c from the set of possible concepts (c_1, ..., c_n). In accordance to related literature we define a classification function f: t → c as

$$f(t) = \operatorname{argmax}_{c_i} (\text{context}(t) \cap \text{context}(c_i)). \qquad (1)$$

As possible concepts c_i we use the listed concepts from the Wikipedia disambiguation page for term t. Then we calculate the term intersection sets of the Wikipedia page for c_i and the extracted and filtered terms of the desktop context. We decide for the Wikipedia concept with the largest intersection set.

[1] The stop-word lists are extracted from the Wikipedia reference corpus of the correspondent language.

[2] term frequency, inverse document frequency [4]

III. Learning Resource Acquisition

Now that we not only have textual description of potential learning topics we are in need of learning resource repositories in order to select and display resources of high relevance with regard to the elicited concepts.

Since the envisioned general purpose learning systems should dynamically adapt to the learning domain we do not want to define relevant resources for each concept manually. This would restrict the usage to the manually engineered concepts and domains. Preferably we would display learning snippets dynamically and the user itself is able to select and use them for self-directed learning.

We differentiate the following types of learning resources which we want to retrieve automatically:

I) *Definition* - general description of a concept in order to distinguish between different concepts

II) *Example* - act of specialization from a general class as clarification for the user

III) *Essay* - a detailed textual description with focus on the concept

IV) *Question and Answer* - description of a problem and a solution addressing the problem

V) *Instruction / User Manual* - directive procedure of actions with regard to performing certain activities with a concept

We would like to display actual instances of learning resources for each type in order to let the learner decide by herself which learning material type he/she prefers and so optimize the learning accomplishment. The following paragraphs describe which repositories can be used to retrieve learning materials for each type of learning goal.

For acquiring definitions of concepts you can easily use the world's largest online lexicon, Wikipedia, which provides short definitions for concepts within the article's abstract section. This learning material type is mostly recommended for advanced users since *learning by definition* often requires earlier knowledge. The Wikipedia data is available for download in a large number of languages and therefore can be integrated into a workplace-embedded prototype without any complications.

Many learners benefit from *learning by example* since this requires less intellectual ability of abstraction. In order to display examples, which could be lists or pictures of the relevant learning concept, we need annotated learning material. Here, we make use of the semantic annotations of the DBpedia[3] project. DBpedia is a collection of semantic relations between objects which are extracted from the Wikipedia metadata. In this paper we make use of the SKOS [8] relation "broader" to retrieve corresponding examples for a concept. Additionally we use links to images included in the foaf:depiction attribute as examples for the determined learning concepts.

Learning material for the resource type *Essay* is determined by using the set of learning concepts as input to a corporate information retrieval system. It delivers documents like presentations, white papers or articles relevant to the given search query. The expectation to find direct answers to a particular learning goal is weakest here, since these resources are most likely not created for a particular learning purpose. But incorporating internal document repositories covers learning topics which might not be available in the public web due to legal or vantage reasons. This yields access to very particular domain-specific knowledge that is not included in open content repositories[4567].

In the recent past a number of community-based *Q&A* services have been started. Websites like WikiAnswers or yeeda.com allow users to ask questions which can be replied by other users. Thus, many possible questions from a broad range of topics already have a reference answer that can be used to offer it to the user as a solution to his learning goal. Similar repositories can be derived from internal or external bulletin boards and user support forums. Also for the learning material type *Instruction* we can rely on public resource repositories. WikiHow.com offers how-to manuals under the Creative Commons license that can be used as a step by step instruction to particular learning goal.

IV. Implementation

The described approach is implemented in a workplace-integrated prototype. A screenshot of the user-visible part is shown in Figure 3. The browser window is continuously updated according to the acquired learning concept collected from the user desktop.

We already plan a number of extensions to improve the quality of the system. The uncertain precision and relevance of retrieved learning resources can be further improved by implicit relevance feedback collected from the user interaction. Technically, the appropriateness of a recommended resource can be evaluated by its click-rate and its averaged display time. Thus our desktop sensor environment also provides an infrastructure for evaluations

In Reference [9], Meyer et al. sketch a promising machine learning approach for determining the didactic function of a previously untagged learning resource. This way, content repositories can be used whose documents are not initially thought to be a learning resource. This could further extend our approach of using learning material with previously known didactical function.

3 http://www.dbpedia.com

4 http://www.lecturefox.com

5 http://www.slidestar.net

6 http://ariadne.cs.kuleuven.be

7 http://www.osotis.com

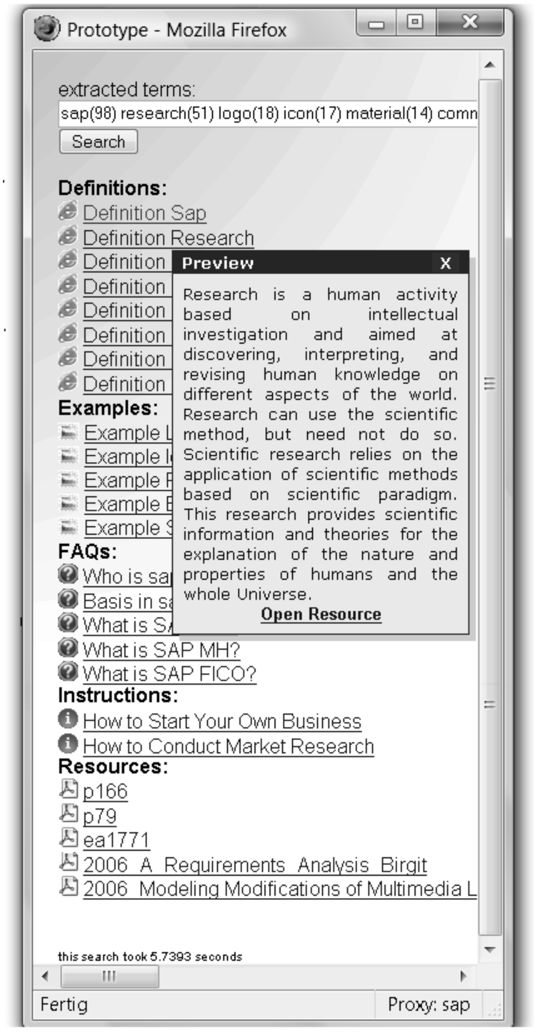

Figure 3. Context-aware prototype embedding different types of learning material to support the user's informational goal

V. Summary

In order to support the idea of a self-directed context-aware e-learning system, that seamlessly integrates into the knowledge worker's workplace, we offer a number of options to automatically acquire learning material that is relevant to the user's need in the work process. Relevance according to the current work process is achieved by observing the user's desktop context. We portray an automated approach for learning concept elicitation by means of text analysis with natural language processing methods. The extracted learning concepts are then used as input for a meshed-up retrieval of learning resources from different public and corporate learning object repositories. Different types of learning material are then offered to the user on the desktop, who is able to select the ones which actually answer his learning need. Despite this approach of in-time e-learning on the workplace is limited to short-term informational learning goals, it contrasts traditional e-learning systems where didactically engineered courses have to be created in a time-consuming and costly manner in advance.

Acknowledgment

We thank Manuel Goertz for lively discussions and valuable feedback on an early version of the idea presented in this paper.

References

[1] D. E. Rose and D. Levinson, "Understanding user goals in web search", *WWW '04: Proceedings of the 13th international conference on World Wide Web*, ACM Press, 2004, 13-19.

[2] R. Lokaiczyk, "Unsupervised Acquisition of Desktop Application Taxonomies", *Proceedings of the IEEE International Conference on Advanced Learning Technologies*, ICALT 2008, Santander, Spain, 2008, in press.

[3] R. Lokaiczyk, A. Faatz, A. Beckhaus, and M. Goertz, "Enhancing Just-in-Time E-Learning through Machine Learning on Desktop Context Sensors", *Proceedings of CONTEXT '07,* 2007, 330-341.

[4] C. Manning, "Foundations of Statistical Natural Language Processing", *MIT Press,* 1999.

[5] P. Velardi and A. Cucchiarelli, "Dependency of context-based word sense disambiguation from representation and domain complexity", in 'NAACL-ANLP 2000 Workshop on Syntactic and semantic complexity in natural language processing systems', Association for Computational Linguistics, Morristown, NJ, USA, 2000, pp. 28-34.

[6] M. Anthony and N. Biggs, "Computational Learning Theory", Cambridge University Press, 1997.

[7] M.J. Kearns and U.V. Vazirani, "An Introduction to Computational Learning Theory", MIT Press, 1994.

[8] A. Miles and D. Brickley (Ed.), "SKOS Core Guide.", W3C, 2. November 2005 (W3C Working Draft 2).

[9] M. Meyer, A. Hannappel, C. Rensing, and R. Steinmetz, "Automatic Classification of Didactic Functions of e-Learning Resources", ACM *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia, ACM Press,* 2007, 513-516.

Authors

Robert Lokaiczyk is with SAP Research, Bleichstr. 8, 64283 Darmstadt, Germany (e-mail: robert.lokaiczyk@sap.com).

Eicke Godehardt is with Fraunhofer-Institute for Computer Graphics, Fraunhoferstr. 5, 64283 Darmstadt, Germany and SAP Research, Bleichstr. 8, 64283 Darmstadt, Germany (e-mail: eicke.godehardt@sap.com).

Dr. Andreas Faatz is with SAP Research, Bleichstr. 8, 64283 Darmstadt, Germany (e-mail: andreas.faatz@sap.com).

Marek Meyer is with KOM Multimedia Communications Lab, Technische Universität Darmstadt, Merckstr. 25, 64283 Darmstadt, Germany and with SAP Research, Bleichstr. 8, 64283 Darmstadt, Germany (e-mail: marek.meyer@sap.com)

Manuscript received 15 April 2008.
Published as submitted by the author(s).